청구의 범위

청구항 1

다수의 연속매체 화일이 기록된 기록매체를 구비하여 통신망을 경유하여 가입자별로 요구된 연속매체 화일들을 동시에 제공하기 위한 연속매체 공급장치를 가진 연속매체 서비스 시스템에 있어서,

상기 연속매체 공급장치로부터의 다수의 연속매체 화일들에 데이터 스트림을 일시적으로 저장하기 위한 연속매체 버퍼와,

상기 연속매체 버퍼의 저장영역을 다수의 블록들로 구분하여 블록별로 제어하는 버퍼 제어수단과,

상기 연속매체 공급장치로부터 가입자별로 부여된 채널의 수에 따라 상기 연속매체 버퍼의 블록들을 할당하고 그 할당된 연속매체 버퍼의 블록들에 다수의 연속매체 화일에 대한 데이타 스트림들이 저장되도록 하고, 상기 연속매체 버퍼의 블록별로 저장되어진 데이터 스트림이 통신망 쪽으로 전송될 때가지의 소요시간에 대한 테이블을 마련하고, 그 테이블에 의하여 데이터 스트림의 전송시점을 결정하는 스트림 스케쥴링 수단과,

상기 연속매체 버퍼에 저장된 상기 데이타 스트림들을 통신망 쪽으로 전송하기 위한 통신망 구동 수단과,

상기 통신망 구동수단과 상기 연속매체 공급장치와의 호신호를 중계하고 상기 스트림 스케쥴링 수단을 제어하는 커널 적응형 중계수단을 구비한 것을 특징으로 하는 연속매체 통신 중계 장치.

청구항 2

제 1 항에 있어서,

상기 스트림 스케쥴링 수단, 상기 버퍼 제어수단 및 상기 통신망 구동수단의 사이에 접속되고, 상기 커널 적응형 중계수단에 제어 하에 스트림 스케쥴링 수단으로부터 데이타 스트림들을 상기 버퍼 제어수단을 경유하여 상기 연속매체 버퍼쪽으로 전송함과 아울러 상기 버퍼 제어수단으로부터의 데이타 스트림들을 상기 통신망 구동수단 쪽으로 전송하는 전송 프로토콜 중계수단을 추가로 구비한 것을 특징으로 하는 연속매체 통신 중계 장치.

청구항 3

제 2 항에 있어서,

상기 통신망 구동수단이 통신망의 다른 방식으로 구동하는 다수의 통신망 중계기를 구비하고,

상기 전송 프로토콜 중계수단은 상기 버퍼 제어수단으로부터 상기 다수의 통신망 중계기들 쪽으로 전송될 데이타 스트림들을 상기 통신망 중계기들이 요구하는 형태로 재 포맷화하는 것을 특징으로 하는 연속매체 통신 중계장치.

명세서

[발명의 명칭]

연속매체 통신 중계 장치

[도면의 간단한 설명]

제 1 도는 본 발명의 실시 예에 따른 연속매체 통신 중계 장치가 적용된 연속매체 서비스 시스템의 블록도.

제 2 도는 제1도에 도시된 연속매체 버퍼의 일 예의 맵핑상태를 도시하는 도면.

제 3 도는 제1도에 도시된 스트림 스케줄러에 의해 작성된 타임 스케줄의 일 예를 도시하는 도면.

제 4 도는 연속매체 화일에 대한 다수의 데이터 스트림이 통신망을 경유하여 실시간적으로 전송되는 것을 설명하는 도면.

* 도면의 주요부분에 대한 부호의 설명

10 : 사용자 모듈 12 : 사용자 인터페이스

14 : 기록매체 20 : 통신망

30 : 커널 모듈 32 : 시스템 콜 인터페이스

34 : 스트림 스케줄러 36 : 전송 프로토콜 처리기

38 : 버퍼 제어기 40 : 연속매체 버퍼

42 : 커널 적응형 중계기 44 : O/S

CID_1 내지 CID_i : 제1 내지 제i 통신망 중계기

ST_1 내지 ST_n : 가입자 단말기

[발명의 상세한 설명]

본 발명은 가입자들의 요구에 부응하는 연속매체 화일들을 통신망을 통하여 가입자 단말기들 쪽으로 전송하는 연속매체 서비스 시스템용 연속매체 서버에 관한 것으로, 특히 다수의 가입자들이 요구한 연속매체 화일들을 연속적이고 실시간적으로 다수의 가입자 단말기들 쪽으로 전송할 수 있는 연속매체 통신 중계장치에 관한 것이다.

통상의 연속매체는 세그먼트의 집합체로 구성된 오디오 및 비디오 프로그램 정보를 의미한다. 이 세그먼트는 독립적으로 디코딩이 가능한 적어도 하나 이상의 데이터 그룹, 예를 들면 MPEG 비디오 데이터의 화상 그룹(GOP; Group of Pictures)으로 구성된다. 여기서, MPEG 비디오 데이터는 동화상 전문가 그룹(Motion Picture Expert Group)에 의해 개발된 압축 알고리즘에 따라 압축된 비디오 데이터를 말하다. 그리고 연속매체 화일은 세그먼트 단위로 데이터 스트림화 되어 전송된다.

그리고 연속매체 서비스 시스템은 기록매체(예를 들면, 디스크)에 저장된 연속매체 화일들을 통신망을 경유하여 다수의 가입자들에게 데이터 스트림의 형태로 공급된다. 이를 위하여, 연속매체 서비스 시스템은 통신망에 접속되어 기록매체로부터의 연속매체 화일들을 통신망을 경유하여 다수의 가입자 단말기들쪽으로 전송하기 위한 하나의 센터(Center)로서 연속매체 서버를 구비한다. 이 연속매체 서버는 연속매체 화일들을 실시간적으로 가입자들에게 공급하기 위하여 가입자별로 통신 채널을 할당함과 아울러 기록매체에 저장된 연속매체 화일들을 데이터 스트림의 형태로 변환하고 가입자별로 할당된 통신 채널들에 연속매체 화일들에 대한 데이터 스트림들을 각각 실어 전송한다. 이를 위하여, 연속매체 서버는 유닉스(UNIX)를 기반으로 하는 오퍼레이팅 시스템(Operating System; 이하 "O/S"라 함)을 사용한다.

그러나, 유닉스를 기반으로 하는 O/S는 동화상과 같은 대용량의 연속매체 화일의 전송을 고려하여 개발되지 않았기 때문에 구조적으로 다수의 연속매체 화일들을 동시에 다수의 가입자들에 전송하기에 부적합하다. 또한, 유닉스를 기반으로 하는 O/S는 연속매체 화일을 구성하는 세그먼트를 기록매체로부터 버퍼로 전송하기에 여러 번의 억세스 과정을 필요로 함으로, 연속매체 화일에 대한 데이타 스트림을 연속적이고 실시간적으로 가입자 단말기 쪽으로 전송하기 곤란하다. 이로 인하여, 연속매체 서비스 시스템은 고품질의 연속매체 서비스를 가입자들에게 제공할 수 없었다.

따라서, 본 발명의 목적은 다수의 가입자들이 요구한 연속매체 화일들을 연속적이고 실시간적으로 다수의 가입자 단말기들 쪽으로 전송할 수 있는 연속매체 통신 중계장치를 제공함에 있다.

본 발명의 다른 목적은 다수의 가입자들이 요구한 연속매체 화일들에 대한 데이타 스트림들을 실시간적으로 가입자들에게 전송될 수 있도록 하는 연속매체 스케쥴링방법을 제공함에 있다.

상기 목적을 달성하기 위하여, 본 발명에 따른 연속매체 서비스 시스템의 연속매체 통신 중계장치는 연속매체 공급장치로부터의 다수의 연속매체 화일들에 대한 데이타 스트림들을 일시적으로 저장하기 위한 연속매체 버퍼와, 연속매체 버퍼의 저장영역을 다수의 블록들로 구분하여 블록별로 제어하는 버퍼 제어 수단과, 연속매체 공급장치로부터 가입자별로 부여된 채널의 수에 따라 연속매체 버퍼의 블록들을 할당하고 그 할당된 연속매체 버퍼의 블록들에 다수의 연속매체 화일에 대한 데이타 스트림들이 저장되도록 하는 스트림 스케쥴링 수단과, 연속매체 버퍼에 저장된 데이타 스트림들을 통신망 쪽으로 전송하기 위한 통신망 구동 수단과, 통신망 구동수단과 연속매체 공급장치와의 호신호를 중계하고 스트림 스케쥴링 수단을 제어하는 커넬 적응형 중계수단을 구비한다.

본 발명에 따른 연속매체 스케쥴링 방법은 연속매체 공급장치로부터의 연속매체 화일들에 대한 데이타 스트림들을 저장하는 연속매체 버퍼의 저장영역을 다수의 스트림 스케쥴링 블록으로 구분하는 단계와, 가입자들에게 부여된 전송채널의 수에 따라스트림 스케쥴링 블록들을 전송채널별로 균등하게 분배하는 단계와, 전송채널들에 실려 전송될 연속매체 화일들에 대한 데이타 스트림들을 할당된 스트림 스케쥴링 블록들에 저장하고 그들이 통신망 쪽으로 전송될 때까지의 소요시간에 대한 스케쥴 테이블을 마련하는 단계와, 스케쥴 테이블에 따라 연속매체 버퍼에 저장된 데이타 스트림들을 전송채널의 수 만큼씩 통신망 쪽으로 전송되도록 하는 단계를 포함한다.

상기 목적을 외에 본 발명의 다른 목적 및 이점들은 첨부 도면을 참조한 다음의 바람직한 실시 예에 대한 설명을 통하여 명백하게 드러나게 될 것이다.

이하, 본 발명의 실시 예를 첨부한 제1도를 참조하여 상세히 설명하기로 한다.

제 1 도는 본 발명의 실시 예에 따른 연속매체를 통신 중계 장치가 적용된 연속매체 서비스 시스템의 블록도이다. 제1도에 있어서, 연속매체 서비스 시스템은 사용자 모듈(10)과 통신망(20) 사이에 접속된 커넬 모듈(30)과, 통신망(20)에 공통적으로 접속된 다수의 가입자 단말기들(ST_1 내지 ST_n)을 구비한다. 사용자 모듈(10)은 통신망(20)과 커넬 모듈(30)을 경유하여 다수의 가입자 단말기들(ST_1 내지 ST_n)에 응답하여 가입자들이 요구한 연속매체 화일들을 제공한다. 이를 위하여, 사용자 모듈(10)

은 커넬 모듈(30)과 직렬 접속된 사용자 인터페이스(12)와 기록매체(14)를 구비한다. 사용자 인터페이스(12)는 다수의 가입자들(ST_1 내지 ST_n)로부터 통신망(20)과 커넬 모듈(30)을 경유하여 입력되는 호에 응답하여 가입자들(ST_1 내지 ST_n)이 원하는 연속매체 화일에 대한 정보를 접수한다. 그리고 사용자 인터페이스(12)는 가입자들이 요구한 연속매체 화일들을 기록매체(14)로부터 판독하고 그 판독된 연속매체 화일들을 커넬 모듈(30)과 통신망(20)을 경유하여 가입자 단말기들(ST_1 내지 ST_n)쪽으로 전송한다. 한편, 기록매체(14)는 적어도 하나 이상의 연속매체 화일들이 저장된 다수의 비디오 카세트 테이프 및 광디스크 등을 구비한다.

커넬 모듈(30)은 다수의 가입자 단말기들(ST_1 내지 ST_n)로부터 통신망(20)을 경유한 호(Call) 신호를 사용자 모듈(10)쪽으로 전송함과 아울러 사용자 모듈(10)로부터의 연속매체 화일들을 다수의 가입자 단말기들(ST_1 내지 ST_n)쪽으로 동시에 연속적이고 실시간적으로 전송한다. 이를 위하여, 커넬 모듈(30)은 사용자 모듈(10)의 사용자 인터페이스(12)에 직렬 접속된 시스템 콜 중계기(32), 스트림 스케쥴러(34), 전송 프로토콜 처리기(36), 버퍼 제어기(38) 및 연속매체 버퍼(40)를 구비한다.

시스템 콜 인터페이스(32)는 사용자 인터페이스(12)와 커넬 적응형 중계기(42)와의 호 처리 신호 및 제어변수들을 양방향으로 전송함과 아울러 사용자 인터페이스(12)를 경유한 기록매체(12)로부터의 연속매체화일을 스트림 스케쥴러(32)쪽으로 전송한다. 여기서, 사용자 인터페이스(12)는 통신망 O/S 변수들과 호 신호들을 시스템 콜 인터페이스(32)를 경유하여 커넬 적응형 중계기(42)쪽으로 전송하고 아울러 가입자 단말기들(ST_1 내지 ST_n)에 의해 발생된 호 신호들을 시스템 콜 인터페이스(32)를 경유하여 커넬 적응형 중계기(42)로부터 입력한다.

그리고 스트림 스케쥴러(34)는 시스템 콜 인터페이스(32)와 통신하여 사용자 인터페이스(12)로부터의 연속매체 화일들을 입력함과 아울러 커넬 적응형 중계기(42)로부터 연속매체 화일들이 실리게 될 채널정보를 입력한다. 그리고 스트림 스케쥴러(34)는 버퍼 제어기(38)에 의해 구분된 연속매체 버퍼(40)내의 스트림 버퍼링 블록들을 채널들에 실시간(Real Time)적으로 할당함으로써 다수의 연속매체 화일들에 대한 스트림을 스케쥴링 한다. 이 스케쥴링 과정을 상세히 하면, 스트림 스케쥴러(34)는 현재 동시 처리해야 할 채널 수에 따라 각 채널별로 스트림 스케쥴링 블록들(ssb)을 제2도와 같이 맵핑시킨다. 그리고 스트림 스케쥴러(34)는 연속매체 화일들이 스트림 스케쥴링 블록(ssb)에 저장된 시점으로부터 통신망(20)쪽으로 전송될 때까지의 소요시간을 나타내는 타임 테이블을 제3도와 같이 생성시킨다. 또한, 스트림 스케쥴러(34)는 이 타임 테이블에 근거하여 연속매체 화일의 실시간 전송을 보장하는 임계값에 가장 근접하는 스트림 스케쥴링 블록에 저장된 데이타를 처리하도록 전송 프로토콜 처리기(36)를 제어한다. 이 결과, 스트림 스케쥴러(34)는 제4도와 같이 N개의 연속매체 화일들에 대한 N개의 데이타 스트림이 동시에 전송되도록 보장한다.

또한, 전송 프로토콜 처리기(36)는 커넬 적응형 중계기(42)로부터의 스트림 스케쥴링 블록들에 대한 어드레스 정보 및 라이트 제어신호와 함께 스트림 스케쥴러(34)로부터의 연속매체 화일들에 대한 데이타 블록을 버퍼 제어기(38)쪽으로 전송하여 연속매체 화일들에 대한 데이타 블록들이 연속매체 버퍼(40)내의 해당 스트림 스케쥴링 블록에 저장되도록 한다. 그리고 전송 프로토콜 처리기(36)는 커넬 적응형 중계기(42)로부터의 스트림 스케쥴링 블록들에 대한 어드레스 정보와 리드 제어 신호를 버퍼 제어기(38)쪽으로 인가함으로써 통신망(20)쪽으로 전송될 연속매체 화일들에 대한 데이타 스트림들을 버퍼 제어기(38)로부터 입력한다. 또한, 전송 프로토콜 처리기(36)는 커넬 적응형 중계기(42)의 제어 하에 데이타 스트림들이 전송될 통신망(20)에 해당하는 다양한 형태로 각각 포맷화 한다. 이 포맷화된 데이타 스트림들은 전송 프로토콜 처리기(36)에 공통적으로 접속된 제1 내지 제i 통신망 중계기(CID_1 내지 CID_i)중 해당하는 통신망 중계기들(CID)을 경유하여 통신망(20)쪽으로 각각 전송된다. 더 나아가, 전송 프로토콜 처리기(36)는 커넬 적응형 중계기(42)와 제1 내지 제i 통신망 중계기(CID_1 내지 CID_i)간의 호 신호를 양방향으로 전송한다. 이 결과, 전송 프로토콜 처리기(36)는 종래의 TCP/IP 및 XTP 등 뿐만 아니라 ATM 모드에서의 가입자와 서버간의 버퍼 플로우 콘트롤(Buffer Flow Control)을 위한 가변 비트 전송(Variable Bit Rate) 기능을 제공한다. 그리고 제1 내지 제i

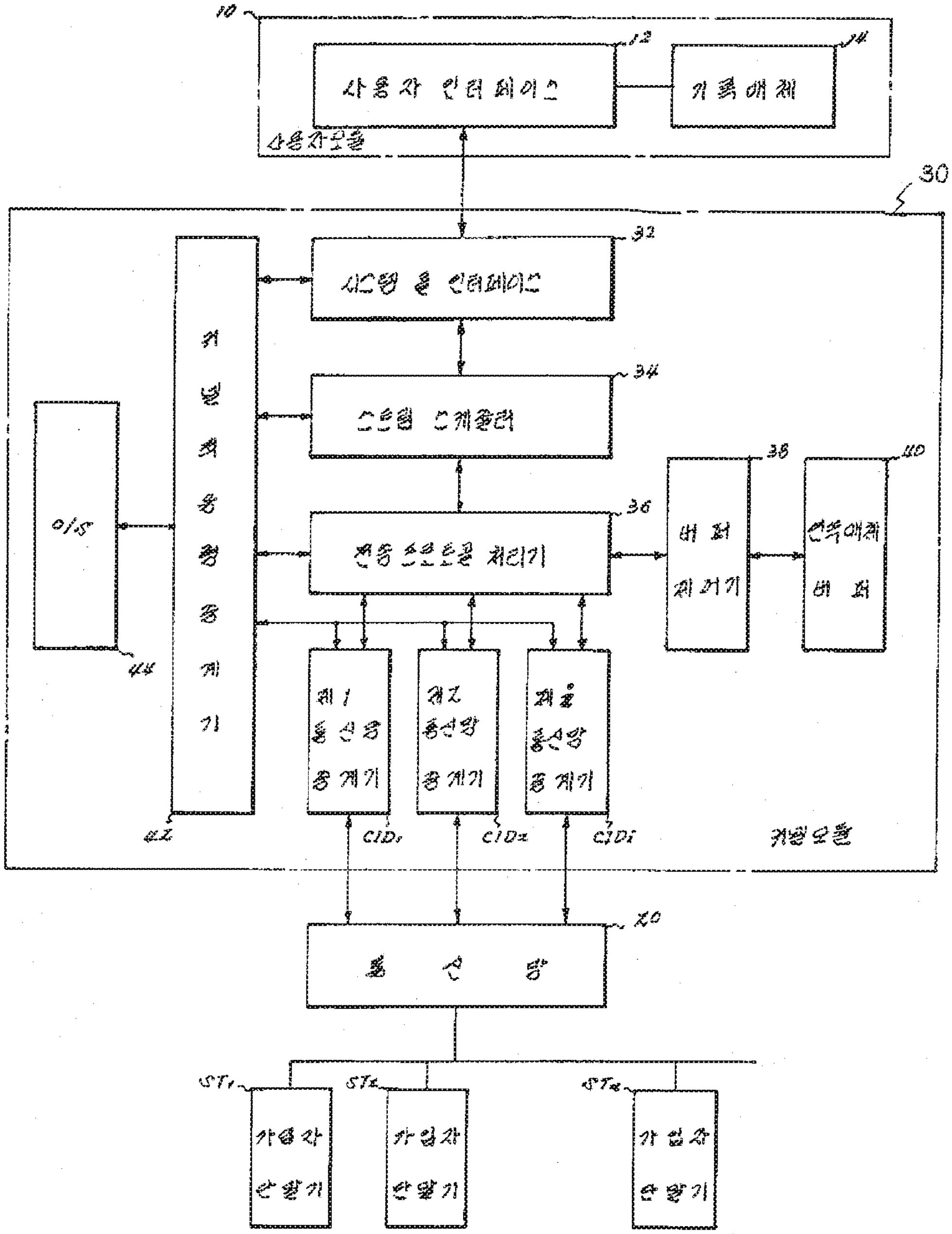
통신망 중계기(CID_1 내지 CID_i)는 각각 커넬 적응형 중계기(42)의 제어 하에 통신망(20)과의 전송 루우프를 개폐한다.

한편, 연속매체 버퍼(40)는 통신망(20)쪽으로 전송될 다수의 연속매체 화일들에 대한 데이타 스트림을 그 전송 채널별로 저장하다. 그리고 버퍼 제어기(38)는 전송 프로토콜 처리기(36)와 통신하여, 스트림 스케쥴러(34)에 의해 맵핑된 형태로 다수의 연속매체 화일들에 대한 데이타 스트림들을 연속매체 버퍼(38)내의 스트림 스케쥴링 블록들(ssb)에 분산 저장함과 아울러 이들 스트림 스케쥴링 블록들(ssb)에 저장된 데이타 스트림들을 전송 프로토콜 처리기(36)에 공급한다.

마지막으로, 커넬 적응형 중계기(42)는 전송 프로토콜 처리기(36)로부터의 호신호를 시스템 콜 인터페이스(32)를 경유하여 사용자 인터페이스(12)쪽으로 전송함과 아울러 사용자 인터페이스(12)로부터 시스템 콜 인터페이스(32)를 경유한 호 신호를 전송 프로토콜 처리기(36)쪽으로 전송한다. 그리고 커넬 적응형 중계기(42)는 O/S(44)중에서 사용자 인터페이스(12)로부터 시스템 콜 인터페이스(32)를 경유한 통신망 제어변수에 해당하는 O/S에 따라 전송 프로토콜 처리기(36)의 통신 모드를 제어한다. 그리고 커넬 적응형 중계기(42)는 사용자 인터페이스(12)로부터의 연속매체 화일들이 실릴 채널 정보를 스트림 스케쥴러(34)쪽으로 중계함과 아울러 스트림 스케쥴러(34)로부터의 어드레스 정보와 리드/라이트 제어신호를 전송 프로토콜 처리기(36)쪽으로 중계한다. 이러한 중계 동작을 수행함으로써 커넬 적응형 중계기(42)는 커넬 모듈(30)이 사용자 모듈(10)에 종속되지 않고 다양한 유닉스 O/S를 독립적으로 지원케 한다.

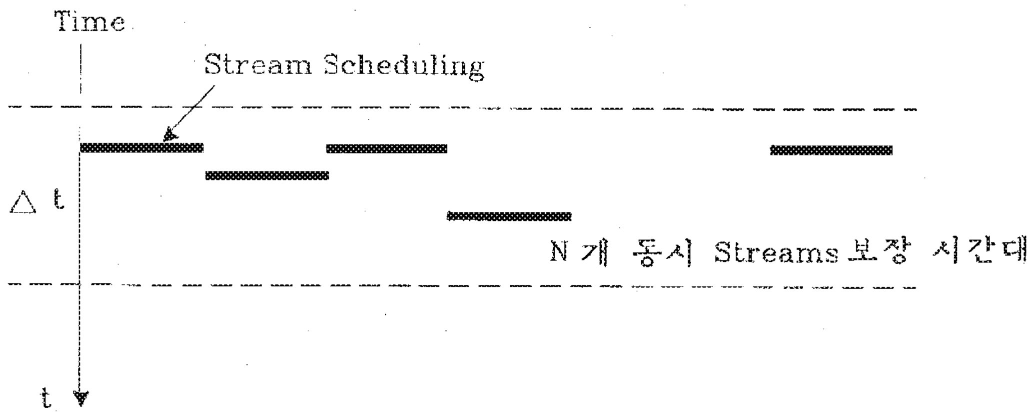
상술한 바와 같이, 본 발명에 따른 연속매체 서비스 시스템의 연속매체 통신 중계 장치는 스트림 스케쥴러를 이용하여 연속매체 버퍼의 저장영역들을 가입자 채널별로 할달하고 할당된 연속매체 버퍼의 저장영역에 연속매체 화일들이 각각 분산시킴으로서 연속매체 화일들을 동시에 연속적으로 가입자들에게 전송할 수 있다. 이에 따라, 본 발명에 따른 연속매체 서비스 시스템의 연속매체 통신 중계 장치는 연속매체 화일들이 실시간적으로 전송되도록 보장할 수 있다. 그리고 본 발명에 따른 연속매체 서비스 시스템의 연속매체 통신 중계 장치는 전송 프로토콜 처리기를 이용하여 다양한 통신 프로토콜을 지원함으로써 통신망의 종류에 무관하게 연속매체화일을 가입자들에게 제공할 수 있는 있다.

이상과 같이, 본 발명은 제1도에 도시된 실시 예로서 설명되었으나 당업자라면 본 발명의 기술사상을 일탈하지 아니하는 범위에서 다양한 변경 및 수정이 가능함을 알 수 있을 것이다. 따라서, 본 발명의 기술적 범위는 명세서의 상세한 설명을 기재된 내용으로 한정되는 것이 아니라 특허 청구의 범위에 의하여 정하여져야만 한다.



ssb11	ssb21	ssb31	ssb41		ssbn1
ssb12	ssb22	ssb32	ssb42		ssbn2
ssb1n	ssb2n	ssb3n	ssb4n		ssbnn

Linked SSB Delta Time Table					
CHi	CHi+1	CHi+2	CHi+3		CHi+N



Espacenet

Bibliographic data: KR100253230 (B1) — 2000-04-15

UNIX NETWORK SUB-SYSTEM FOR CONTINUOUS STREAM TRANSFER

Inventor(s):	LIM HYUN-CHANG [KR] ± (LIM, HYUN-CHANG)
Applicant(s):	LG ELECTRONICS INC [KR] ± (LG ELECTRONICS INC)
Classification:	**- international:** ***H04L12/28;*** (IPC1-7): H04L12/28 **- cooperative:**
Application number:	KR19970072518 19971223
Priority number(s):	KR19970072518 19971223

Abstract of KR100253230 (B1)

PURPOSE: A unix network sub-system for continuous medium stream transfer is provided to minimize a radiation movement of a data by clients by directly transmitting a continuous medium stream data received from a file system through a buffer through a network when a stream control command is received from a server manager. CONSTITUTION: A plurality of operating modules(COM) transmit a continuous medium stream data received from a buffer through a pre-set connection, and is dynamically generated or terminated. A management module(CAM) dynamically manages the operating module(COM) for every stream according to a stream control command received from a server manager. The management module(CAM) includes a manager agent for interpreting the received stream control command and performing a command, an operation module manager for generating, controlling and managing the operation module(COM), a connection manager for setting up and releasing a connection, a QoS repeater for converting a QoS(Quality of Service) of a user received from the buffer to a QoS of a network and maintaining a connection by means of the connection manager, an address repeater for converting a session ID received from the buffer to an address of a client and outputting it, an admission checker for checking the amount of a network resource for a new session request by the buffer and checking whether the request can be admitted; and a system tuner for changing a tuning parameter value of a continuous medium network sub-system. The operation module(COM) includes a command receiver for receiving a management command from the management module(CAM) and processing it, a discarder for discarding a frame remaining in the buffer after releasing the connection, an error controller for re-transmitting a data lost during transmission for accurate transmission, a frame stamper for informing a transmission state of a frame being transmitted, a frame divider for dividing the received frame suitably to each client, and a flow controller for controlling a transfer rate.

(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2000년04월15일
(11) 등록번호 10-0253230
(24) 등록일자 2000년01월22일

(51) 국제특허분류코드
H04L 12/28 (2006.01)

(21) 출원번호 10-1997-0072518

(22) 출원일자 1997년12월23일
심사청구일자 2000년01월22일

(65) 공개번호 특1999-0052960

(43) 공개일자 1999년07월15일

(72) 발명자
임현창
서울특별시 서초구 양재동 329-7, 대한민국

(73) 특허권자
구자홍
엘지전자주식회사
서울특별시 영등포구 여의도동 20번지, 대한민국

(74) 대리인
박장원

(77) 심사관
이상용

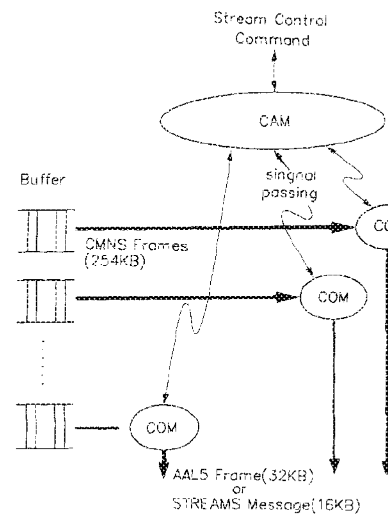
전체 청구항 수 : 총 3 항

(54) 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조

(57) 요 약

본 발명은 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조에 관한 것으로, 종래에는 일반 데이터를 전송하는 비디오 서버의 네트웍 서브시스템을 이용하여 연속매체 스트림 데이터를 전송할 경우 시스템을 수정없이 그대로 사용하므로 연속매체 데이터의 특성에 맞지않아 그 만큼 효율이 떨어지는 문제점과 실시간으로 전송할 수 없는 분제점이 있다. 따라서 본 발명은 이미 설정된 커넥션을 통해 버퍼로 부터 전달된 연속매체 스트림 데이터를 전송하고, 동적으로 생성 또는 소멸하는 복수개의 동작 모듈(COM)과, 서버 관리자로 부터 스트림 컨트롤 커맨드에 따라 한 스트림마다 하나씩 상기 동작 모듈을 동적으로 관리하는 관리 모듈(CAM)로 구성하여, 각 클라이언트별로 데이터의 복사 이동을 최소화하도록 하고, 동적으로 생성되는 동작 모듈을 이용하여 연속매체의 스트림만 전송할 수 있도록 하여 연속매체를 실시간으로 전송할 수 있도록 한 것이다.

대표도 - 도 1



청구의 범위

청구항 1

이미 설정된 커넥션을 통해 버퍼로 부터 전달된 연속매체 스트림 데이터를 전송하고, 동적으로 생성 또는 소멸하는 복수개의 동작 모듈(COM)과, 서버 관리자로 부터 스트림 컨트롤 커맨드에 따라 한 스트림마다 하나씩 상기 동작 모듈을 동적으로 관리하는 관리 모듈로 구성된 것을 특징으로 하는 연속매체 스트림 전송을 위한 유닉스 네트워크 서브시스템 구조.

청구항 2

제1항에 있어서, 동작 모듈은 관리 모듈로 부터 관리 커맨드를 받아 처리하는 커맨드 수신기와, 커넥션 해제 후에 버퍼에 남아있는 프레임을 버리는 프레임 디스카더와, 전송도중 잃어버린 데이터에 대해 재전송하여 정확히 전달되도록 하는 에러 컨트롤러와, 전송되는 프레임의 전송상태를 알려주는 프레임 스탬퍼와, 전달받은 프레임을 각 클라이언트에 맞게 분할하여 주는 프레임 분할기와, 전송되는 전송율을 조절하는 플로우 컨트롤러로 구성된 것을 특징으로 하는 연속매체 스트림 전송을 위한 유닉스 네트워크 서브시스템 구조.

청구항 3

제1항에 있어서, 관리 모듈은 전달받은 스트림 컨트롤 커맨드를 해석하여 그 커맨드를 수행하는 매니저 에이전트와, 동작 모듈(COM)을 생성, 제어 및 관리하는 동작모듈용 매니저와, 커넥션을 설정하고 해제하는 역할을 수행하는 커넥션 매니저와, 버퍼로 부터 전달받은 유저의 QoS(Quality Of Service)를 네트워크의 QoS로 변환시켜 상기 커넥션 매니저가 이용하여 커넥션을 유지하도록 한 QoS중계기와, 버퍼로 부터 전달받은 세션(session) 아이디(ID)를 어드레스로 변환시켜 출력하는 어드레스 중계기와, 버퍼로 부터의 새로운 세션 요구에 대해 네트워크 자원의 양을 알아보고 그 요구를 수용할 수 있는지의 여부를 점검하는 어드미션 체커와, 연속매체 네트워크 서브시스템의 튜닝 파라미터의 값을 변경하는 시스템 튜너로 구성된 것을 특징으로 하는 연속매체 스트림 전송을 위한 유닉스 네트워크 서브시스템 구조.

명세서

발명의 명칭

연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조
도면의 간단한 설명

도 1은 본 발명의 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조도.

도 2는 도 1에서, COM과 CAM의 상세도.

도 3은 도 1에서, COM상태 천이도.

*** 도면의 주요부분에 대한 부호의 설명 ***

11 : 매니지 에이전트 12 : 시스템 튜너

13 : 어드미션 체커 14 : 동작모듈 매니지

15 : QoS 중계기 16 : 어드레스 중계기

17 : 커넥션 매니지 21 : 커맨드 수신기

22 : 프레임 디스카더 23 : 에러 컨트롤러

24 : 타임 스탬퍼 25 : 프레임 분할기

26 : 플로우 컨트롤러

발명의 상세한 설명

본 발명은 연속매체 스트림 데이터의 특성에 맞는 유닉스 네트웍 서브시스템 구조에 관한 것으로, 특히 서버 관리자로 부터 스트림 컨트롤 커맨드 입력시 버퍼를 통해 파일 시스템(File system)으로 부터 전달받은 연속매체 스트림 데이터를 망을 통해 바로 전송하도록 함으로써 각 클라이언트별로 데이터의 복사 이동을 최소화하고, 실시간 전송이 가능하도록 한 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조에 관한 것이다.

종래에는 비디오 서버의 네트웍 서브시스템은 문자 데이터, 통계자료 데이터와 같은 일반적인 데이터를 전송하는데 사용하였다.

그리고, 연속매체 스트림에 대하여도 상기 비디오 서버의 네트웍 서브시스템을 이용하여 전송하였다.

그러나, 상기에서와 같은 종래기술에서 일반 데이터를 전송하는 비디오 서버의 네트웍 서브시스템을 이용하여 연속매체 스트림 데이터를 전송할 경우 시스템을 수정없이 그대로 사용하므로 연속매체 데이터의 특성에 맞지않아 그 만큼 효율이 떨어지는 문제점과 실시간으로 전송할 수 없는 문제점이 있다.

따라서 상기에서와 같은 종래의 문제점을 해결하기 위한 본 발명의 목적은 서버 관리자로 부터 스트림 컨트롤 커맨드 입력시 버퍼를 통해 파일 시스템(File system)으로 부터 전달받은 연속매체 스트림 데이터를 망을 통해 바로 전송하도록 함으로써 각 클라이언트별로 데이터의 복사 이동을 최소화하도록 한 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조를 제공함에 있다.

본 발명의 다른 목적은 각 클라이언트별로 스트림 데이터를 실시간으로 전송할 수 있도록 한 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조를 제공함에 있다.

본 발명의 또 다른 목적은 유닉스 커널(UNIX kernel)에 구현하여 시스템 클럭, 타이머, 인터럽트 핸들러에 대한 접근이 용이하도록 한 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조를 제공함에 있다.

상기 목적을 달성하기 위한 본 발명은 이미 설정된 커넥션을 통해 버퍼로 부터 전달된 연속매체 스트림 데이터를 전송하고, 동적으로 생성 또는 소멸하는 복수개의 동작 모듈과, 서버 관리자로 부터 스트림 컨트롤 커맨드에 따라 한 스트림마다 하나씩 상기 동작 모듈을 동적으로 관리하는 관리 모듈로 구성한다.

이하, 첨부한 도면에 의거하여 상세히 살펴보면 다음과 같다.

도 1은 본 발명의 연속매체 스트림 전송을 위한 유닉스 네트웍 서브시스템 구조도로서, 이에 도시한 바와같이, 이미 설정된 커넥션을 통해 버퍼로 부터 전달된 연속매체 스트림 데이터를 전송하고, 동적으로 생성 또는 소멸하는 복수개의 동작 모듈(COM)과, 서버 관리자로 부터 스트림 컨트롤 커맨드에 따라 한 스트림마다 하나씩 상기 동작 모듈을 동적으로 관리하는 관리 모듈(CAM)로 구성한다.

상기 관리 모듈(CAM)은, 도 2에 도시한 바와같이, 전달받은 스트림 컨트롤 커맨드를 해석하여 그 커맨드를 수행하는 매니저 에이전트(11)와, 동작 모듈(COM)을 생성, 제어 및 관리하는 동작모듈 매니저(14)와, 커넥션을 설정하고 해제하는 역할을 수행하는 커넥션 매니저(17)와, 버퍼로 부터 전달받은 유저의 QoS(Quality Of Service)를 네트웍의 QoS로 변환시켜 상기 커넥션 매니저가 이용하여 커넥션을 유지하도록 한 QoS중계기(15)와, 버퍼로 부터 전달받은 세션(session) 아이디(ID)를 클라이언트의 어드레스로 변환시켜 출력하는 어드레스 중계기(16)와, 버퍼로 부터의 새로운 세션 요구에 대해 네트웍 자원의 양을 알아보고 그 요구를 수용할 수 있는지의 여부를 점검하는 어드미션 제거(13)와, 연속매체 네트웍 서브시스템의 튜닝 파라미터의 값을 변경하는 시스템 튜너(12)로 구성한다.

상기 동작 모듈(COM)은, 도 2에 도시한 바와같이, 관리 모듈로 부터 관리 커맨드를 받아 처리하는 커맨드 수신기(21)와, 커넥션 해제 후에 버퍼에 남아있는 프레임을 버리는 프레임 디스카더(22)와, 전송도중 잃어버린 데이터에 대해 재전송하여 정확히 전달되도록 하는 에러 컨트롤러(23)와, 전송되는 프레임의 전송상태를 알려주는 프레임 스탬퍼(24)와, 전달받은 프레임을 각 클라이언트에 맞게 분할하여 주는 프레임 분할기(25)와, 전송되는 전송율을 조절하는 플로우 컨트롤러(26)로 구성한다.

이와같이 구성된 본 발명의 동작 및 작용 효과에 대하여 상세히 설명하면 다음과 같다.

연속매체 스트림을 전송하는 유닉스 네트웍 서브시스템은 비디오 서버 관리자로 부터 스트림 컨트롤 커맨드를 전달받으면 파일 시스템(File system)으로 부터 버퍼가 연속매체 스트림 데이터를 읽어오고, 그 버퍼가 읽어들인 연속매체 스트림 데이터를 망을 통해 전송한다.

이와같은 동작을 수행하는 유닉스 네트웍 서브시스템은 도 1에서와 같이 하나의 관리 모듈(CAM)과 여러개의 동작 모듈(COM)로 구성된다.

상기 관리 모듈(CAM)과 동작 모듈(COM)은 오퍼레이팅 시스템(OS)의 하부 구조인 커널(Kernel)내의 트레이드(Thread)로 구현한다.

비디오 서버 관리자로 부터 스트림 컨트롤 커맨드가 입력되면, 관리 모듈(CAM)의 매니저 에이전트(11)가 그 스트림 컨트롤 커맨드를 해석한 후 관리모듈 매니저(14)와 커넥션 매니저(17)를 호출하여 커맨드를 수행하도록 한다.

예를 들면, 전송해야 할 스트림 데이터가 있으면 관리모듈 매니저(14)에게 그 스트림에 대한 동작모듈(COM)에게 이미 열려진 커넥션을 통하여 데이터 전송을 담당하도록 한다.

그러면 상기 관리모듈 매니저(14)는 전송해야 할 스트림 데이터가 있으면 동작모듈(COM)을 생성하고, 그 생성된 동작모듈(COM)을 제어하거나 관리하고, 커넥션 매니저(17)는 상기 동작 모듈(COM)이 클라이언트로 스트림 데이터를 전송할 수 있도록 커넥션을 설정해주고, 전송이 끝나면 커넥션을 해제한다.

이때 QoS 중계기(15)는 서버 관리자의 버퍼(Buffer)에서 전달받은 QoS(Quality Of Service) 파라미터를 네트웍 QoS로 변환시키면, 이를 상기 커넥션 매니저(17)가 이용하여 커넥션을 유지하도록 한다.

상기에서 QoS 파라미터를 네트웍 QoS로 변환시키는 이유는 QoS 파라미터가 유저 QoS이기 때문이다.

그리고, 어드레스 중계기(16)는 서버 관리자의 버퍼로 부터 전달받은 세션 ID를 클라이언트의 어드레스로 변환시키고, 어드미션 체커(13)는 상기 버퍼로 부터의 새로운 세션 요구에 대해 네트웍 자원의 양을 알아보고, 그 요구를 수용할 수 있는지의 여부를 점검하고, 시스템 튜너(12)는 유닉스 네트웍 서브시스템의 튜닝 파라미터의 값을 변경하는 역할을 한다.

이상에서와 같이 관리 모듈(CAM)을 구성하는 각 부에서 동작을 수행하여 동작 모듈(COM)이 생성되면, 상기 동작 모듈(COM)은 관리 모듈(CAM)의 제어에 의해 파일 시스템으로 부터 전송되는 스트림 데이터를 각 클라이언트로 전송하여 주는데, 이에 대하여 살펴보면 다음과 같다.

관리 모듈(CAM)의 동작모듈 매니저(14)가 관리 명령을 전송하면, 도 2에서 동작 모듈(COM)을 구성하는 커맨드 수신기(21)에서 커맨드를 받아 처리한다.

가령 커맨드 수신기(21)로 전달된 커맨드가 스트림 데이터를 전송하라는 것이면, 이를 프레임 분할기(25)로 알려준다.

그러면 상기 프레임 분할기(25)는 버퍼(Buffer)를 통해 파일 시스템으로 부터 전송된 데이터는 256KB이므로 이를 64KB의 프레임으로 분할하여 TCP/IP 네트워크(30)로 전송한다.

그러면 TCP/IP 네트웍(30)을 통해 각 클라이언트로 전송된다.

이때 타임 스탬퍼(24)는 상기에서 분할된 프레임이 각 클라이언트로 전달되는 시점의 시간을 유지하여 해당 스트림이 종료되었을 때 유지하고 있던 시간 정보를 화면에 출력하거나 특정 파일에 출력한다.

그리고 상기에서와 같은 전송 도중에 잃어버린 데이터가 있으면 에러 컨트롤러(23)에서 제어하여 재전송이 이루어지도록 하여 정확한 스트림 데이터가 전송되도록 하고, 플로우 컨트롤러(26)는 전송되는 스트림 데이터의 전송율을 조절한다.

이와같은 방법으로 생성된 동작 모듈(COM)에서 스트림 데이터를 전송한 후 커넥션이 해제된 후 버퍼(Buffer)에 남아 있는 불필요한 프레임을 버리는 동작을 프레임 디스카더(22)에서 실행한다.

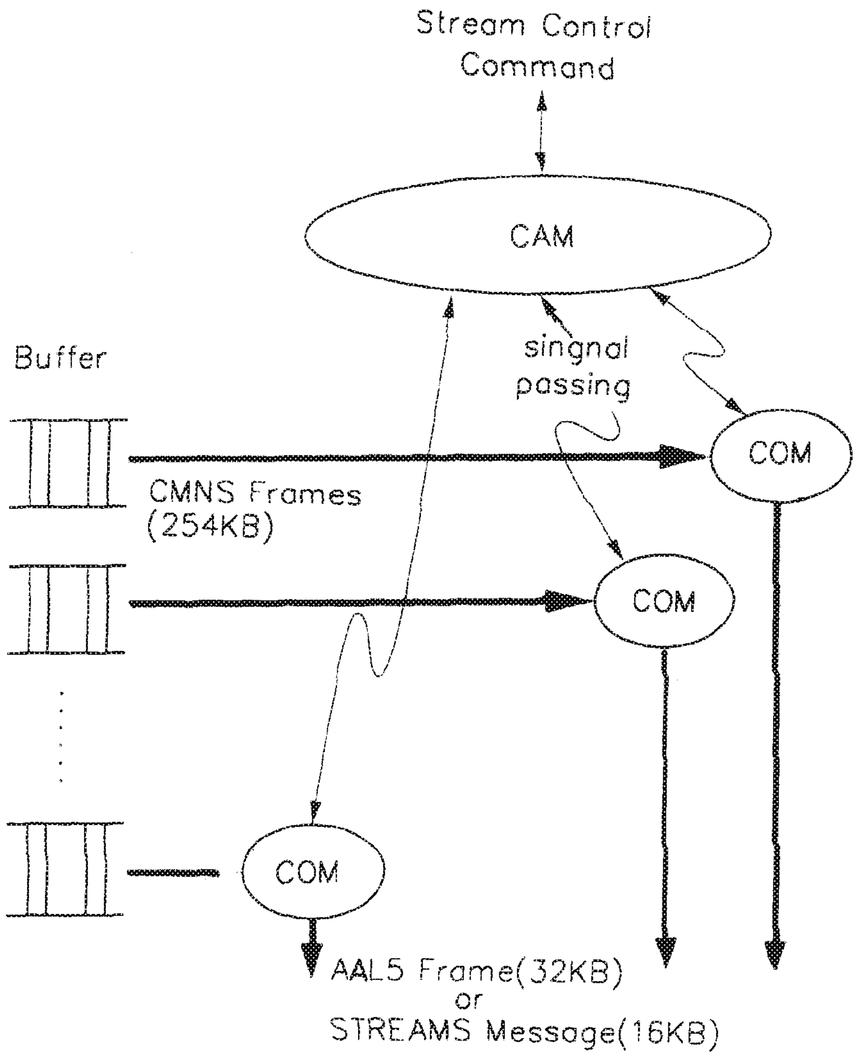
상기에서와 같은 동작에 대하여 도 3에 도시한 천이도를 이용하여 관리 모듈(CAM)이 동작 모듈(COM)을 관리하는 과정에 대하여 살펴보면, 동작 모듈이 생성되지 않은 상태인 COM_NONE에서 관리 모듈(CAM)이 스트림 오픈 커맨드를 전달받으면 오퍼레이팅 시스템(OS)에 LWP_CREATE(생성)와 LWP_SUSPEND(중지)신호를 보낸다.

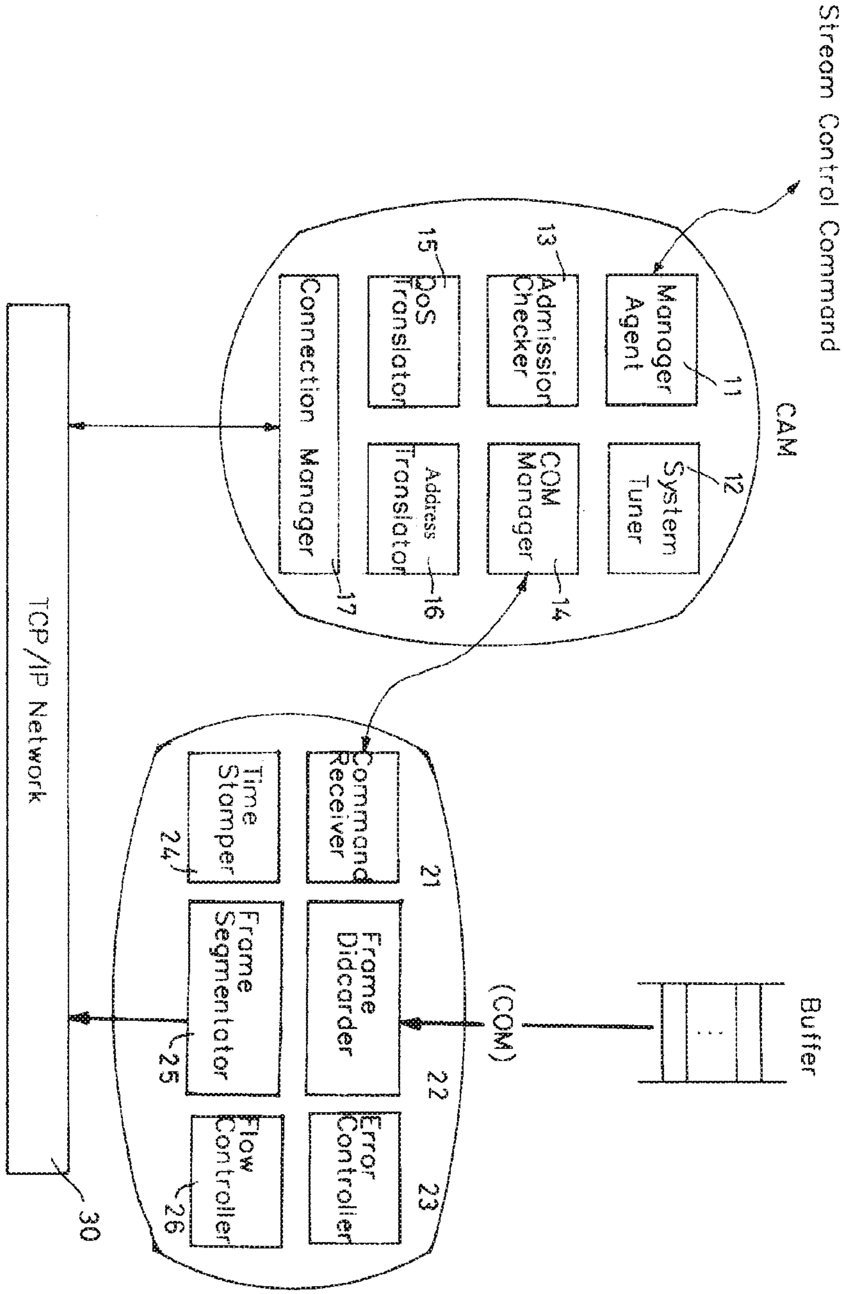
그러면 오퍼레이팅 시스템(OS)에 의해 동작 모듈(COM)이 생성되고(COM_CREATE)된 후 동작을 정지상태를 유지한다.(COM_SUSPEND)

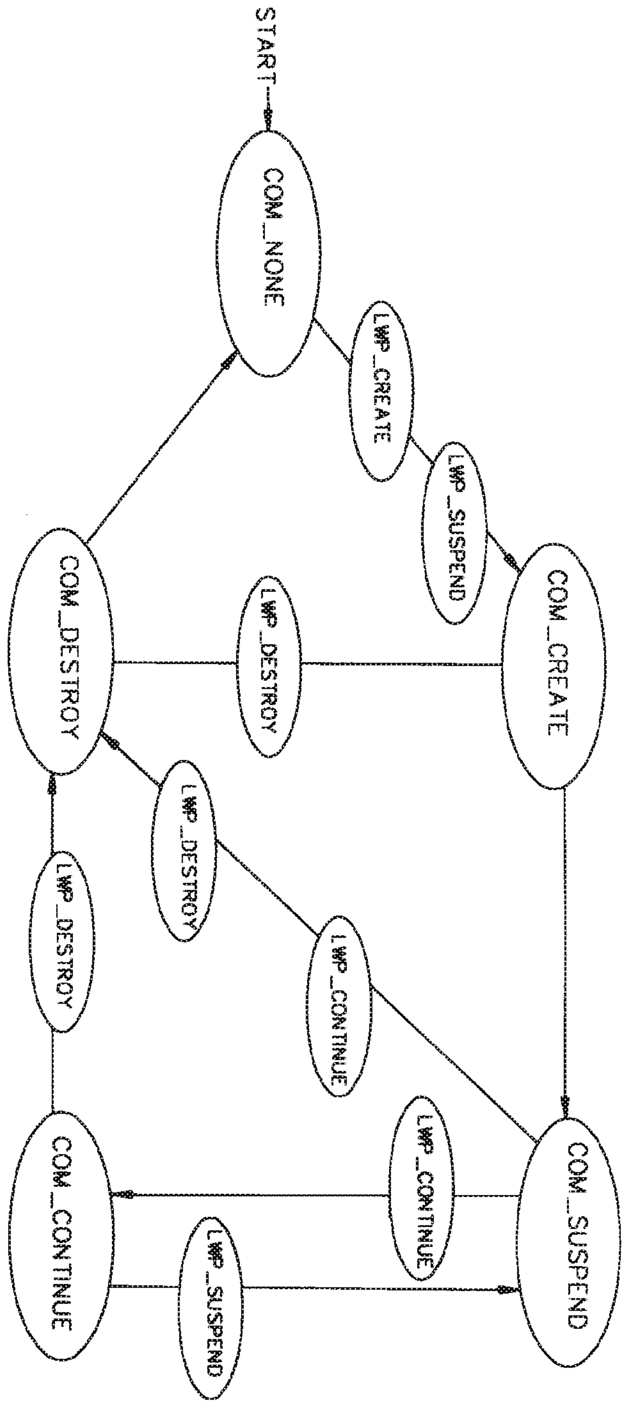
상기 생성 후 중지상태에 있는 동작 모듈로 LWP_CONTINUE와 같은 플레이신호를 보내면, 동작 모듈이 동작하고, LWP_DESTROY와 같은 끝났으면 해제하라는 신호를 보내면 스트림을 전송하는 동작을 수행한 후 해제하여 소멸된다.

이와같은 방법으로 동작 모듈(COM)이 동적으로 생성되어 동작하므로 세션의 관리 측면에서 효율적이고, 연속매체 스트림의 전송만 담당하므로 연속매체를 실시간으로 전송할 수 있다.

따라서, 본 발명은 서버 관리자로 부터 스트림 컨트롤 커맨드 입력시 버퍼를 통해 파일 시스템(File system)으로 부터 전달받은 연속매체 스트림 데이터를 망을 통해 바로 전송하도록 함으로써 각 클라이언트별로 데이터의 복사 이동을 최소화하도록 하고, 동적으로 생성되는 동작 모듈을 이용하여 연속매체의 스트림만 전송할 수 있도록 하여 연속매체를 실시간으로 전송할 수 있도록 하고, 또한 유닉스 커널에 구현됨으로써 시스템 클럭, 타이머, 인터럽트 핸들러에 대한 접근이 용이하도록 한 효과가 있다.


Stream Control
Command
CAM
Buffer
singnal
passing
COM
CMNS Frames
(254KB)
COM
COM
AAL5 Frame(32KB)
or
STREAMS Message(16KB)


Stream Control Command
CAM
Manager Agent
11
System Tuner
12
Admission Checker
13
COM Manager
14
QoS Translator
15
Address Translator
16
Connection Manager
17
Buffer
(COM)
Command Receiver
21
Frame Didcarder
22
Error Controller
23
Time Stamper
24
Frame Segmentator
25
Flow Controller
26
TCP/IP Network
30


START
COM_NONE
LWP_CREATE
LWP_SUSPEND
COM_CREATE
COM_DESTROY
LWP_DESTROY
LWP_DESTROY
LWP_DESTROY
LWP_CONTINUE
COM_SUSPEND
LWP_CONTINUE
COM_CONTINUE
LWP_SUSPEND

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04B	A2	(11) International Publication Number: **WO 97/12447** (43) International Publication Date: 3 April 1997 (03.04.97)

(21) International Application Number: PCT/IL96/00104

(22) International Filing Date: 8 September 1996 (08.09.96)

(30) Priority Data:
115263 12 September 1995 (12.09.95) IL

(71) Applicant (*for all designated States except US*): VOCALTEC LTD. [IL/IL]; 1 Maskit Street, 46733 Herzelia (IL).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): COHEN, Alon [IL/IL]; 6 Yakinton Street, 75426 Rishon Lezion (IL).

(74) Agent: A. TALLY EITAN - ZEEV PEARL & CO.; Lumir House, 22 Maskit Street, 46733 Herzlia (IL).

(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: SYSTEM AND METHOD FOR DISTRIBUTING MULTI-MEDIA PRESENTATIONS IN A COMPUTER NETWORK

(57) Abstract

A method and a system for distributing a multi-media presentation in a computer network is provided. The method includes the steps of feeding at least one site of said computer network with a stream of data of the presentation and forming in each site a plurality of data files, each data file including a segment of said multi-media presentation. The method further includes distributing the data files to at least one display and displaying the distributed data files.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

SYSTEM & METHOD FOR DISTRIBUTING MULTI-MEDIA PRESENTATIONS IN A COMPUTER NETWORK

FIELD OF THE INVENTION

The present invention relates to a method and system for distributing video, audio graphic and textual information on a computer network.

BACKGROUND OF THE INVENTION

Computer networks, in particular the INTERNET network, allow users operating user operated displays (UOD) connected to the network, to receive and display multi-media presentations. Usually, multimedia presentations are stored as files which are downloaded from sites of the network in which they are stored and to which they were fed from any source.

In the prior art, each multi-media file stored in a network site contains all the data of a presentation, i.e. if the multi-media presentation is a video clip of a song, the file stored in the network site includes the video, audio, graphic and textual data of the particular video clip and therefore, when retrieved by a network user, a relatively long delay exists before the user can display the presentation on his computer since the volume of data that needs to be transferred from the network site to the user is very large.

For example, if the multi-media presentation is a 660 Kbyte file, it will be transferred via a 14.4K modem in about 12 minutes. An example of such 660 Kbyte file is a 1 minute audio file at 11 Ksamp/sec.

SUMMARY OF THE INVENTION

An object of the present invention is to provide an improved method and system for distributing multi-media presentations to users of a computer network.

Another object of the present invention is to provide substantially in real time multi-media presentations to at least one user operating a UOD connected to the network.

A further object of the present invention is to enable a user operating a UOD to selectively display portions of or the entire multi-media presentation. The user operating the UOD may display the presentation with a desired delay, to re-display portions of the presentation or to display portions of the presentation out of order.

It will be appreciated that the term multi-media refers herein to video, audio, graphic, textual, music instrument device interface (midi) or any other digital data, each taken alone or in any combination therebetween. The term display refers herein to the display of any data included in the multi media presentation.

According to an aspect of the present invention, the system enabling the selective display includes a feeding unit which feeds the multimedia presentation, a forming and distribution unit which feeds the multimedia presentation as data files including segments of the presentation and a UOD, all connected in any suitable computer network. Preferably, the data files include consecutive segments of the presentation.

According to a preferred embodiment of the present invention, the computer network is the INTERNET network.

According to a further aspect of the present invention, for the INTERNET network, the forming and distribution unit reside in World Wide Web (WWW) sites. The forming unit and the distribution unit may be either in the same WWW site or in a different WWW site.

According to yet another aspect of the present invention, the forming unit may be part of the feeding unit wherein the distribution unit is a HyperText Transfer Protocol (HTTP) server in a WWW site.

According to yet another aspect of the present invention, additional distribution units located in additional sites of the WWW may retrieve the data files including the segments of the multi-media presentation from the distribution unit and distribute them to additional users employing UODs communicating with these additional sites.

One advantage of the present invention is that the communication protocol employed for connecting the UOD, the HTTP WWW server and the multi-media presentations feeding computer is a communication protocol presently used in the INTERNET, preferably a reliable TCP/IP communication protocol. TCP/IP is compatible with HTTP WWW and browsing applications used for browsing the WWW.

According to a second preferred embodiment of the present invention, the communication protocol is any suitable communication protocol, such as a non-reliable UDP communication protocol.

There is thus provided, according to a preferred embodiment of the present invention, a method for distributing a multi-media presentation in a computer network which includes the following steps:

A. feeding at least one site of the computer network with a stream of data of the presentation;

B. forming, in each site a plurality of data files, each data file including a segment of the multi-media presentation;

C. distributing the plurality of data files to at least one display; and

D. displaying the distributed data files.

In accordance with a preferred embodiment of the present invention, the feeding includes providing at least one parameter characterizing the segment and the forming includes forming consecutive data files, each including the segment in accordance with the at least one parameter.

Further, according to a preferred embodiment of the present invention, the step of forming includes forming the data files as a sequence of files. Alternatively, the step of forming includes forming the data files in a cyclic fashion.

Additionally, according to a preferred embodiment of the present invention, the steps of distributing and displaying include the step of activating an interactive display application operating to receive the distributed data files and to display them in a user selected sequence. Preferably, the user selected sequence is selected from the group which includes displaying a most current file, displaying a formerly formed file, displaying a formerly distributed file, re-displaying displayed files and displaying the files out of order. The most current file refers herein to the file most recently formed by the forming unit.

Further, according to a preferred embodiment of the present invention, the method may include the step of selecting a time lag between the steps of feeding and forming and the steps of distributing and displaying, the time lag determined whether the display is substantially a real time display or a delayed display.

Still further, according to yet another preferred embodiment of the present invention, the step of distributing may also include the step of distributing the data files to additional distribution units operating to distribute the data files to additional displays.

According to a preferred embodiment of the present invention, the computer network is the INTERNET network.

There is also provided, in accordance with a preferred embodiment of the present invention, a method for displaying substantially in a real time a multi-media presentation which includes the steps of:

A. forming data files representing segments of the multi-media presentation; and

B. displaying the data files in a user selected sequence, the user selected sequence includes at least one of the group which includes displaying a most current file, displaying a formerly distributed file, re-displaying displayed files and displaying the files out of order.

There is further provided, in accordance with a preferred embodiment of the present invention a system for distributing a multi-media presentation in a computer network which includes:

A. a feeding unit for feeding at least one site of the computer network with a stream of data of the presentation;

B. a forming manager for forming, in each site a plurality of data files, each file including a segment of the multi-media presentation;

C. a distribution unit for distributing the plurality of data files to at least one UOD; and

D. a UOD for displaying the distributed data files.

According to a preferred embodiment of the present invention, the computer network is the INTERNET network and the distribution units are one or more HTTP servers in WWW sites.

There is further provided, in accordance with a preferred embodiment of the present invention, a method for distributing a multi-media presentation in a computer network which includes the following steps:

A. feeding at least one site of the computer network with a stream of data of the presentation;

B. forming, in each site a data file including the multi-media presentation;

C. repeating the steps of feeding and forming, thereby rewriting portions of the data of the presentation in a cyclic fashion;

D. distributing, substantially simultaneously with the steps of feeding and forming, the current data file to at least one display; and

E. displaying the distributed data file,

wherein a selectable time lag exists between the steps of distributing and displaying.

Finally, according to a preferred embodiment of the present invention, there are provided systems which includes units operating to carry out the steps of the methods of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be understood and appreciated more fully from the following detailed description taken in conjunction with the appended drawings in which:

Fig. 1 is a schematic block diagrams of a multi-media presentation distributing system, constructed according to a preferred embodiment of the present invention;

Fig. 2 is a schematic block diagram illustration of a preferred method for feeding and forming the files of consecutive segments of the multi-media presentation;

Fig. 3 is a schematic illustration of a plurality of files including consecutive segments of the multi-media presentation formed in a cyclic mode;

Fig. 4 is a schematic illustration of a plurality of files including consecutive segments of the multi-media presentation formed in a sequential mode;

Fig. 5 is a schematic block diagram illustration of a preferred method for distributing and displaying the files of segments of the multi-media presentation;

Fig. 6 is a schematic block diagram illustration of an alternative preferred method for feeding and forming a multi-media presentation; and

Fig. 7 is a schematic block diagram illustration illustrating the steps of distributing and displaying of the method of Fig. 6.

DETAILED DESCRIPTION OF THE PRESENT INVENTION

Reference is now made to Fig. 1 which illustrates a multi-media presentation distribution system, constructed according to a preferred embodiment of the present invention.

The system of Fig. 1, generally referenced 10, comprises a multi-media presentation feeding unit 12 for feeding a stream of data which forms the multi-media presentation to be distributed through the computer network via a forming and distribution unit 14 to a plurality of user operated display (UOD) units of which one, referenced 16, is shown herein.

The unit 12 may comprise any suitable means for generating a multi-media presentation. Preferably, it comprises a computer equipped with suitable software and hardware for receiving the multi-media presentation from any suitable source. For example, if the multi-media presentation is an audio presentation, the computer forming the unit 12 will include an audio card capable of receiving audio signals, such as Sound Blaster, manufactured and sold by Creative Lab Technologies of the USA. If the multi-media presentation is a video presentation, the computer forming the unit 12 will include a video card capable of receiving video signals, such as Video Blaster, manufactured and sold by Creative Lab Technologies of the USA.

According to a preferred embodiment of the present invention, the unit 12 receives a real time broadcast of a multi-media presentation, for example an audio presentation from a radio station, and feeds its content to the forming and distribution unit 14.

The forming and distribution unit 14 is preferably a site in a computer network operating to receive the stream of multi-media data from the unit 12 and to form therefrom a plurality of files forming together the multi-media presentation.

According to a preferred embodiment of the present invention, the system is operating to provide substantially in real time the multi-media presentation to the users and therefore the feeding and forming are substantially simultaneous, i.e. the unit 14 receives the stream of data according to parameters defined by the feeding unit 12 and substantially simultaneously forms the files including the segments of the multi-media presentation.

The forming and distribution unit 14 preferably comprises a file forming manager 18 which receives the data of the multi-media presentation and forms data files including segments of the presentation therefrom. Preferably, the data files are consecutive data files representing consecutive segments of the multi-media presentation.

In the illustrated embodiment, the feeding unit 12 and the forming unit 18 are located in different sites of the network. Alternatively, the forming unit 18 may form part of the feeding unit 12 and not of the forming and distribution unit 14. Also, the feeding unit 12 and the forming and distribution unit 14 may reside together on a single WWW site.

The preferred network of the present invention is the INTERNET network and therefore the distribution unit is preferably an HTTP WWW server 20 which receives the files including segments of the multi-media presentation and is operative to distribute them to any number of UODs of users of the network as described in detail with reference to Fig. 5 hereinbelow.

Additionally, the distribution unit 20 is operative to distribute the data files including segments of the multi-media presentation to additional distribution units (not shown), i.e. HTTP servers in other WWW sites, whereby the multi-media presentation may be distributed substantially simultaneously to even a larger number of users.

Alternatively, each additional distribution unit may receive the data files including segments of the multi-media presentation from a corresponding retrieving unit (not shown) which operates to retrieve the data files from the unit 14 and to transfer it to the corresponding additional distribution unit.

Referring now also to Fig. 2, a preferred method for feeding and forming the files representing consecutive segments of the multi-media presentation is described. The method starts with the step 22 of activating the computer comprising the feeding unit 12.

Next, as shown in block 24, the feeding unit 12 provides the parameters defining the characteristics of the segments of stream of data of the multi-media presentation that are transferred to the file forming manager 18. The stream of data representing each of the files including the segments of the multi-media presentation is then transferred to the file forming manager 18 as indicated by step 26 and the file forming manager forms the files of the data segments of the multi-media presentation (step 28).

The characteristics provided in step 24 may be any suitable characteristics of the segment data files. Non limiting examples include the size of the data file and the cycle size in case the data is provided in a cyclic fashion as described hereinbelow with reference to Fig. 3.

It will be appreciated that the method of Fig. 2 is particularly advantageous in an environment in which it is desired to display the multi-media presentation substantially in real time. While in the prior art the stream of data representing the entire multi-media presentation is transferred to a single file in the HTTP server, thereby causing a relatively long delay between feeding and subsequent non-continuous displaying, the segmented forming of the present invention enables to display the multi-media presentation substantially simultaneously with its feeding and forming.

Furthermore, according to one preferred embodiment of the present invention, as shown in Fig. 3, the files are formed in the HTTP server 20 in a cyclic fashion. In the first cycle, files 3A - 3N, i.e. the files allocated to save segments of the multi-media presentation in the first cycle, are formed. The files of the next cycle, i.e. files 3N + 1, 3N + 2 etc. will replace files 3A, 3B etc., respectively, and the files of a further cycle represented by file 3(2N)+1 will replace file 3N+1 etc.

Forming in a cyclic fashion enables not only to display the multi-media presentation substantially simultaneously but also continuously as new files presenting segments of the multi-media presentation replace older files for any desired period of time.

It will be appreciated that forming the files in the forming unit 18 in a cyclic fashion is particularly advantageous for relatively long real time display of multi-media presentations such as the display of a radio station broadcast to users connected to the network.

According to an alternative embodiment of the present invention, as shown in Fig. 4, the plurality of files may be formed in a sequential mode, i.e. as a string of data files 4A - 4N representing the content of the entire multi-media presentation.

Reference is now made to Fig. 5 which is a schematic block diagram illustration of a preferred method for distributing and displaying the files of segments of the multi-media presentation.

The method starts at block 52 where a user views a WWW page on its UOD's display 16. According to a preferred embodiment of the present invention the communication between the UOD 16, the HTTP server 20, and the feeding unit 12 is via a TCP/IP communication protocol, thereby allowing the user to employ existing browsing application, such as the National Center for Supercomputing Applications (NCSA), NCSA Mosaic browser and the NetScape browser, commercially available from Netscape Communication Corp. of California, U.S.A.

The WWW page includes an indication to a connection file which is selected by the user (step 54). Upon selection, the UOD receives the connection file (step 56) which includes reference to an interactive display application capable of receiving and displaying the files of the segments of the multi-media presentation.

In the preferred embodiment, the user activates the interactive display application (step 58) via the browsing application.

The connection file is updated in the WWW and therefore includes the status of all the files currently in the distribution unit 20. Therefore, once the interactive display application is activated, it also receives the current status of the files including the segments of the multi-media presentation.

The interactive display application now has all the pertinent information about the files currently available in the HTTP server and in 60 the user receives the files to be displayed as shown in 62. If the presentation is a real time presentation, the information regarding the status of the files may include the file having the most current segment of the multi-media presentation, the size of cycle and the user options to delay and/or to re-display some of the files. Preferably, the steps 60 and 62 are substantially simultaneous.

It will be appreciated that the steps 60 and 62 may continue in a cyclic fashion as indicated by the cycle 64 substantially simultaneously with the cyclic forming of new files described hereinabove.

It will be appreciated that the time lag between the feeding and forming described with reference to Fig. 2 and the steps of distributing and displaying described with reference to Fig. 5 may be selected by the user. If a substantially real time display is of interest a minimal lag time will be selected and if a delayed display is of interest, the lag therebetween will be longer.

A particular advantage of the present invention is that the user operating the UOD may interact with the displayed presentation to select which part thereof will be displayed. In case of a substantially real time presentation, the user is not limited to displaying the content of the most current part of the presentation. According to the present invention, the user interacts with the interactive display application to select whether to display files which were distributed at an earlier time or the current file, or may "jump" and display only selected files, thereby displaying segments of the multi-media presentation out of order.

Reference is now made to Figs. 6 and 7 which illustrate an alternative embodiment for feeding, forming, distributing and displaying a multi-media presentation employing the system 10 (Fig. 1). In the embodiment of Fig. 6, the feeding unit 12 operates to feed a stream of data (step 126) to the forming unit 18 which forms a single data file therefrom (step 128). The steps 126 and 128 are continuous as indicated by the cycle schematically referenced 127.

It will be appreciated that the cycle 127 effectively forms a data file to which data of the multi-media presentation is written in a cyclic fashion, data which may be displayed substantially simultaneously with the cycle 127 as described with reference to Fig. 7 hereinbelow, provided that there is a sufficient time lag between the step 128 and the display of the same segment by the UOD. This time lag may also be varied selectably by the user.

As shown in Fig. 7, the user views a WWW page on its UOD's 16 (step 152). Then, the user activates an interactive display application capable of receiving and displaying the data file currently formed in step 128 (Fig. 6) employing its UOD browsing application (step 154).

The interactive display application now has all the pertinent information about the file currently available in the HTTP server 20 and in 160 the user receives the file to be displayed as shown in 162. Preferably, the steps 160 and 162 are substantially simultaneous.

It will be appreciated that the steps 160 and 162 may continue in a cyclic fashion as indicated by the cycle 164 substantially simultaneously with the cyclic forming of the data file described in step 127.

It will be appreciated that while the present invention is not limited by what has been described hereinabove and that numerous modifications, all of which fall within the scope of the present invention, exist. For example, while the present invention has been described with reference to the INTERNET network, the method and system of the present invention is not limited thereto and may be employed in any suitable computer network.

Another example is that the data fed by the feeding unit can be compressed and subsequently decompressed by the UOD. Similarly, the data files including segments of the multi-media presentation or the single file may be compressed in the HTTP server and subsequently decompressed in the UOD. These compression and decompression steps may be performed by any suitable compression and decompression algorithm known in the art.

It will be appreciated by persons skilled in the art that the present invention is not limited to what has been particularly shown and described hereinabove. Rather, the scope of the present invention is defined only by the claims that follow:

CLAIMS

1. A method for distributing a multi-media presentation in a computer network comprising:
   a. feeding at least one site of said computer network with a stream of data of said presentation;
   b. forming, in each site a plurality of data files, each data file including a segment of said multi-media presentation;
   c. distributing said plurality of data files to at least one display; and
   d. displaying said distributed data files.

2. The method of claim 1 wherein said feeding comprises providing at least one parameter characterizing said segment and said forming comprises forming consecutive data files, each including said segment in accordance with said at least one parameter.

3. The method of claim 1 wherein said forming comprises forming said data files as a sequence of files.

4. The method of claim 1 wherein said forming comprises forming said data files in a cyclic fashion.

5. The method of claim 1 wherein said distributing and displaying comprises activating an interactive display application operating to receive said distributed data files and to display them in a user selected sequence.

6. The method of claim 5 wherein said user selected sequence is selected from the group consisting of displaying a most current file, displaying a formerly formed file, displaying a formerly distributed file, re-displaying displayed files and displaying the files out of order.

7. The method of claim 1 comprising selecting a time lag between said steps of feeding and forming and said steps of distributing and displaying, said time lad determines whether said displaying is substantially a real time display or a delayed display.

8. A method according to any of the previous claims and wherein said distributing further comprises distributing said data files to additional distribution units operating to distribute said data files to additional displays.

9. A method according any of the previous claims and wherein said computer network is the INTERNET network.

10. A method for displaying substantially in a real time a multi-media presentation comprising:
    a. forming data files representing segments of said multi-media presentation; and
    b. displaying said data files in a user selected sequence, said user selected sequence comprising at least one of the group consisting of displaying a most current file, displaying a formerly formed file, displaying a formerly distributed file, re-displaying displayed files and displaying the files out of order.

11. A system for distributing a multi-media presentation in a computer network comprising:
    a. a feeding unit for feeding at least one site of said computer network with a stream of data of said presentation;
    b. a forming manager for forming, in each site a plurality of data files, each file including a segment of said multi-media presentation;
    c. a distribution unit for distributing said plurality of data files to at least one UOD; and
    d. a UOD for displaying said distributed data files.

12. The system of claim 11 wherein said feeding unit comprises means for providing at least one parameter of characterizing said segment and said forming unit comprises means for forming consecutive data files, each including said segment, in accordance with said at least one parameter.

13. The system of claim 11 wherein said forming unit operates to form said data files as a sequence of files.

14. The system of claim 11 wherein said forming unit operates to form said data files in a cyclic fashion.

15. The system of claim 11 wherein said distribution unit and said UOD comprise means for activating an interactive display application operating to receive said distributed data files and to display them in a user selected sequence.

16. The system of claim 15 wherein said user selected sequence is selected from the group consisting of displaying a most current file, displaying a formerly formed file, displaying a formerly distributed file, re-displaying displayed files and displaying the files out of order.

17. The system of claim 11 comprising means for selecting a time lag between said steps of feeding and forming and said steps of distributing and displaying, said time lag determined whether said displaying is substantially a real time display or a delayed display.

18. A system according to any of claims 11 - 17 and wherein said network includes additional distribution units operating to receive said data files and to distribute them to additional UODs.

19. A system according to any of claims 11 - 18 and wherein said computer network is the INTERNET network.

20. A system according to claim 19 wherein said distribution unit is an HTTP server.

21. A method for distributing a multi-media presentation in a computer network comprising:

a. feeding at least one site of said computer network with a stream of data of said presentation;

b. forming, in each site a data file including said multi-media presentation;

c. repeating said steps of feeding and forming, thereby rewriting portions of the data of said presentation in a cyclic fashion;

d. distributing, substantially simultaneously with said steps of feeding and forming, the current data file to at least one display; and

e. displaying said distributed data file, wherein a selectable time lag exists between said forming and distributing.

22. A system for distributing a multi-media presentation in a computer network comprising:

a. a feeding unit for feeding at least one site of said computer network with a stream of data of said presentation;

b. a forming unit for forming, in each site a data file including said multi-media presentation;

c. means for repeating the operation of said feeding unit and forming unit, thereby rewriting portions of the data of said presentation in a cyclic fashion;

d. means for distributing, substantially simultaneously with the operation of said feeding unit and forming unit, the current data file to at least one display; and

e. a UOD for displaying said distributed data file, wherein a selectable time lag exist between said forming and distributing.

23. A method according to any of claims 1 - 10 and 21 substantially as described hereinabove.

24. A method according to any of claims 1 - 10 and 21 substantially as illustrated in any of the drawings.

25. A system according to any of claims 11 - 20 and 22 substantially as described hereinabove.

26. A system according to any of claims 11 - 20 and 22 substantially as illustrated in any of the drawings.

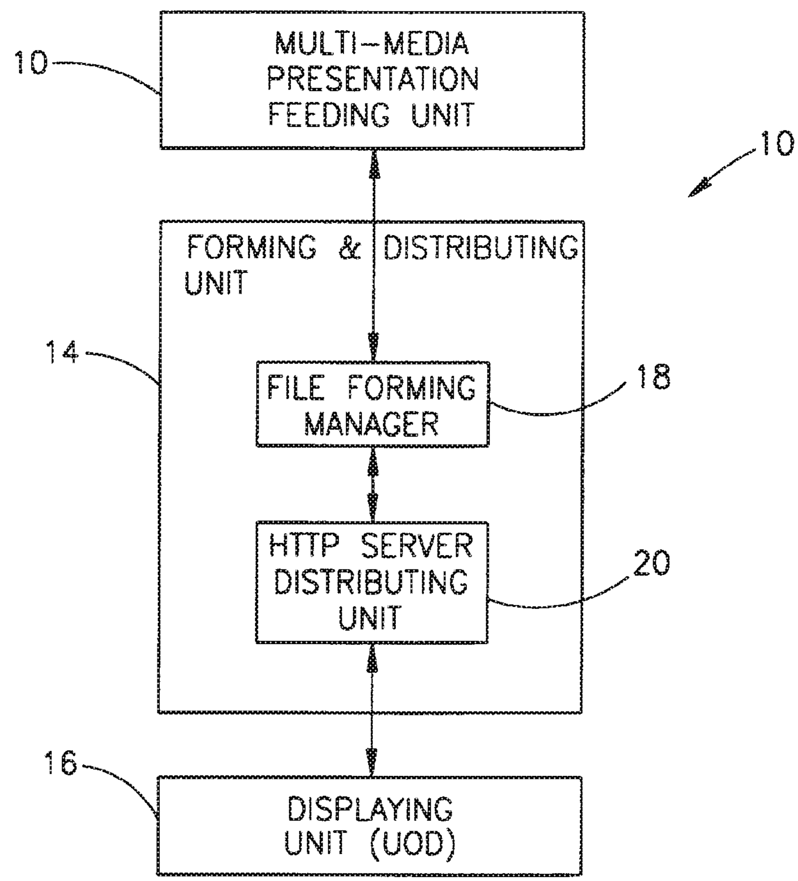

MULTI-MEDIA PRESENTATION FEEDING UNIT
10
10
FORMING & DISTRIBUTING UNIT
14
FILE FORMING MANAGER
18
HTTP SERVER DISTRIBUTING UNIT
20
16
DISPLAYING UNIT (UOD)

FIG.1

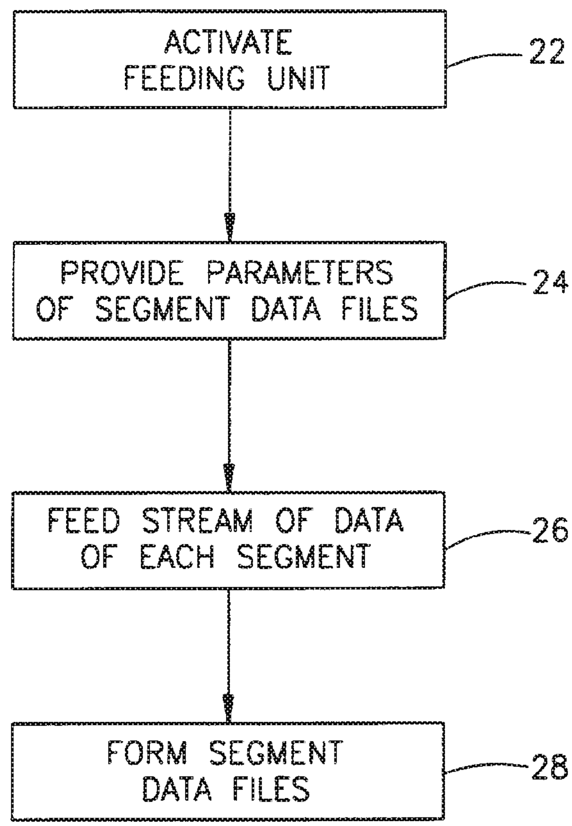

ACTIVATE FEEDING UNIT
22
PROVIDE PARAMETERS OF SEGMENT DATA FILES
24
FEED STREAM OF DATA OF EACH SEGMENT
26
FORM SEGMENT DATA FILES
28

FIG.2

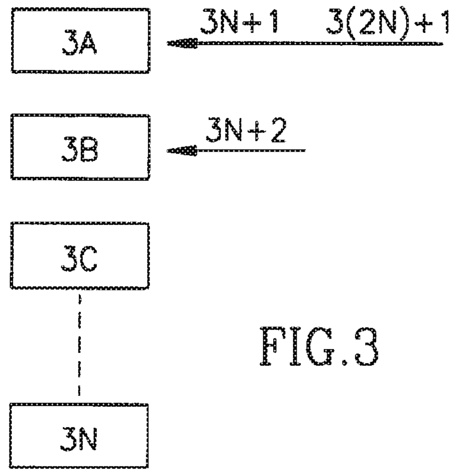


FIG.3

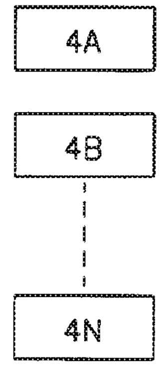


FIG.4

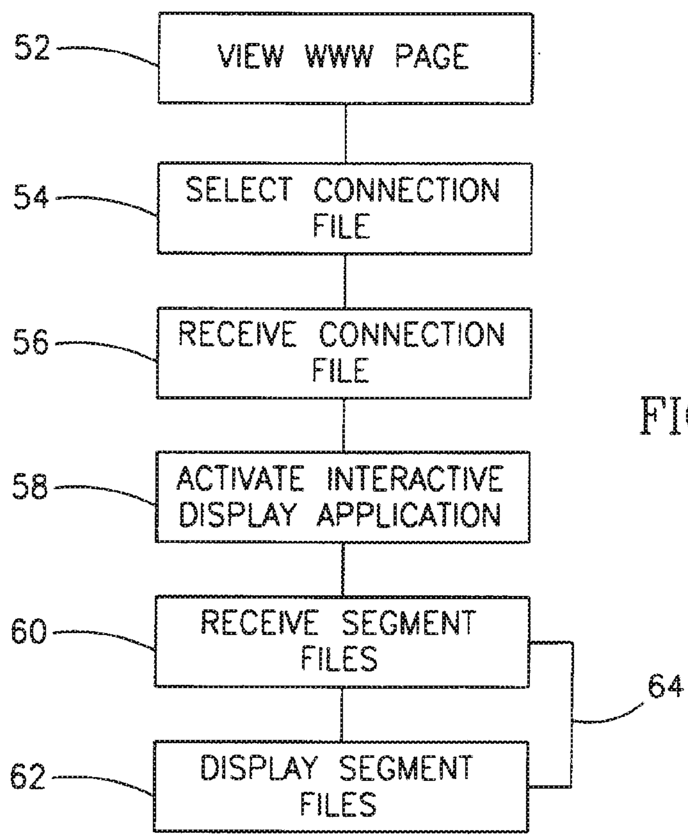


FIG.5

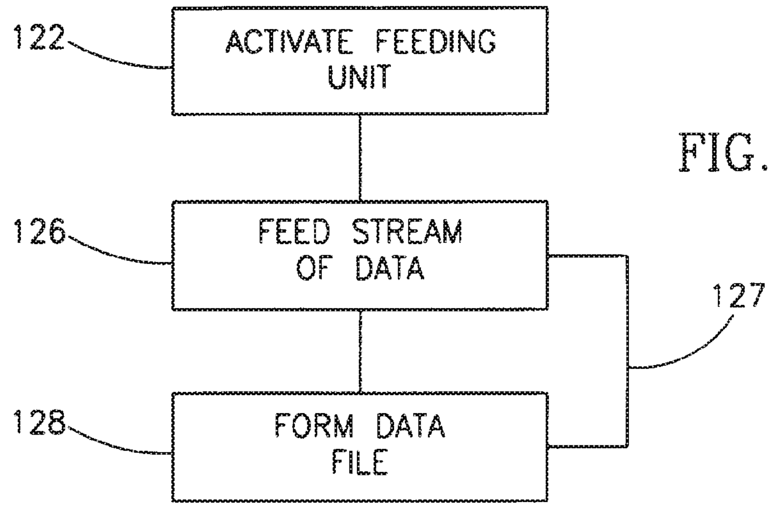

122
ACTIVATE FEEDING UNIT
126
FEED STREAM OF DATA
127
128
FORM DATA FILE

FIG.6

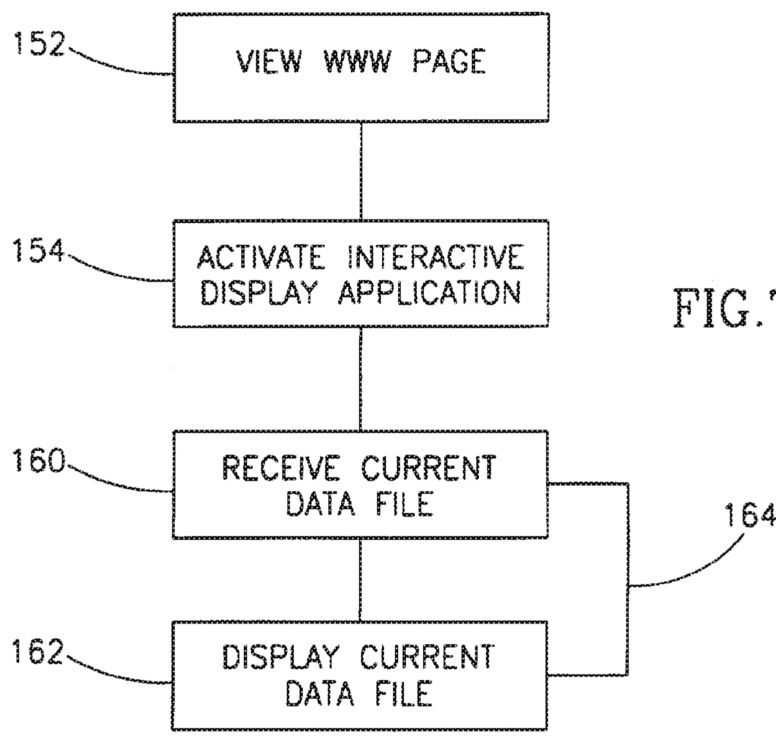

152
VIEW WWW PAGE
154
ACTIVATE INTERACTIVE DISPLAY APPLICATION
160
RECEIVE CURRENT DATA FILE
164
162
DISPLAY CURRENT DATA FILE

FIG.7

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 15/00

A3

(11) International Publication Number: WO 97/12447

(43) International Publication Date: 3 April 1997 (03.04.97)

(21) International Application Number: PCT/IL96/00104

(22) International Filing Date: 8 September 1996 (08.09.96)

(30) Priority Data:
115263 12 September 1995 (12.09.95) IL

(71) Applicant (*for all designated States except US*): VOCALTEC LTD. [IL/IL]; 1 Maskit Street, 46733 Herzelia (IL).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): COHEN, Alon [IL/IL]; 6 Yakinton Street, 75426 Rishon Lezion (IL).

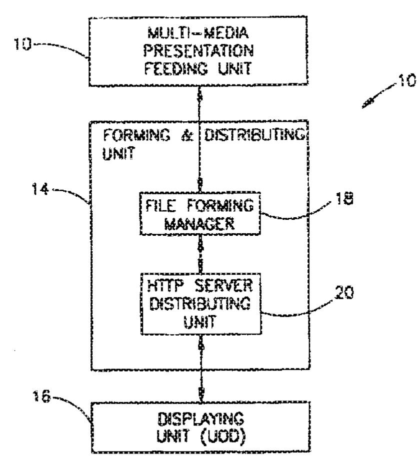
(74) Agent: A. TALLY EITAN - ZEEV PEARL & CO.; Lumir House, 22 Maskit Street, 46733 Herzlia (IL).

(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(88) Date of publication of the international search report: 04 September 1997 (04.09.97)

(54) Title: SYSTEM AND METHOD FOR DISTRIBUTING MULTI-MEDIA PRESENTATIONS IN A COMPUTER NETWORK



(57) Abstract

A method and a system for distributing a multi-media presentation in a computer network is provided. The method includes the steps of feeding at least one site of said computer network with a stream of data of the presentation and forming (18) in each site a plurality of data files, each data file including a segment of said multi-media presentation. The method further includes distributing the data files (20) to at least one display and displaying the distributed data files (16).

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL96/00104

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 15/00
US CL :395/806
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/806, 807; 348/7

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS--search terms: video-on-demand, multimedia or video? present?, network, distribut?, display?,

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 4,903,126 A (KASSATLY) 20 February 1990, entire document.	1-26
X,E	US 5,592,626 A (PAPADIMITRIOU et al.) 07 January 1997, entire document.	1-26
X,E	US 5,594,936 A (REBEC et al.) 14 January 1997, entire document.	1-26
X,E	US 5,614,940 A (COBBLEY et al.) 25 March 1997, entire document.	1-26
X,E	US 5,583,994 A (RANGAN) 10 December 1996, entire document.	1-26
X,E	US 5,583,561 A (BAKER et al.) 10 December 1996, entire document.	1-26

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
14 APRIL 1997	26 JUN 1997
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	Authorized officer JOSEPH R. BURWELL
Facsimile No. (703) 305-3230	Telephone No. (703) 305-3800

Form PCT/ISA/210 (second sheet)(July 1992)*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL96/00104

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,E	US 5,572,442 A (SCHULHOF et al.) 05 November 1996, entire document.	1-26
X,E	US 5,557,541 A (SCHULHOF et al.) 17 September 1996, entire document.	1-26

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04H 1/00	A1	(11) International Publication Number: WO 97/17775 (43) International Publication Date: 15 May 1997 (15.05.97)

(21) International Application Number: PCT/FI96/00594

(22) International Filing Date: 5 November 1996 (05.11.96)

(30) Priority Data:
955356 7 November 1995 (07.11.95) FI

(71) Applicant *(for all designated States except US)*: OY NOKIA AB [FI/FI]; Eteläesplanadi 12, FIN-00130 Helsinki (FI).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: SALOMÄKI, Ari [FI/FI]; Ojavainionkatu 10 D 28, FIN-33710 Tampere (FI).

(74) Agent: LUOTO, Kristian; Nokia Research Center, Pl 45, FIN-00211 Helsinki (FI).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.
In English translation (filed in Finnish).

(54) Title: MULTIMEDIA RECEPTION IN A DIGITAL BROADCASTING SYSTEM

(57) Abstract

To transfer a multimedia programme in a DAB system, an audio stream is divided into successive segments of variable length, which are marked by means of individual stream marker IDs placed at segment boundaries. The boundaries between segments indicate a change in the multimedia presentation. A scene in the multimedia programme, which, when active, is what the user sees on the display of the receiver, contains links that are waiting to be activated. When a marker associated with one of the links and acting as an excitation is decoded from the audio stream, it activates the link concerned. As a result, either the scene is displayed or the link causes a transition to another scene, which is displayed on the screen. The transition is invisible to the user, so the user perceives the presentation as starting with the right scene.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

Multimedia Reception in a Digital Broadcasting System

The present invention relates to the transfer of a multimedia programme and in particular to the mechanisms used to initiate such transfer in a receiver in a digital broadcasting system.

In the Digital Audio Broadcasting (DAB) system, which has been developed to allow an efficient utilization of frequency bands, the transmission path is completely digital. The system is designed to replace the analogue broadcasting system commonly used at present, which is based on the use of frequency modulation. DAB defines a digital radio channel based on multiple carriers which is applicable for the transmission of both audio and data services. In a completely digital transmission channel, it is possible to transmit a continuous data or audio stream, or the channel may be a packet channel. Packet transmission is more flexible and permits easier transmission of data units of a limited length. The DAB system is defined in ETSI (European Telecommunication Standards Institute) standard 300 401, February, 1995.

From the user's point of view, the highest level of abstraction in the DAB system is called ensemble, Fig. 1. It contains all the services that are available to the user in a given frequency band. A change from one ensemble to another in the receiver is effected by tuning into a different frequency band, just as one changes channels in current FM radio reception. The ensemble is divided into services, exemplified in Fig. 1 by Alpha Radio 1, Beta Radio and Alpha Radio 2. In addition, there may be data services, although these are not shown in the figure. Each service is consists of one or more service components, and each of these is placed in a subchannel, which may be either an audio channel or a data channel. For comparison, let it be stated that FM radio contains only one service and one service component (audio) in each channel. At the lowest level, the transmission frame, whose duration is either 24 ms or 96 ms depending on the DAB mode, consists of three chronologically consecutive parts. The first part is a Synchronizing Channel, which contains no service information. The next part is a Fast Information Channel FIC, which has a mode-specific fixed length. The last part is a Main Service Channel MSC, which contains all the subchannels. The position, size and number of subchannels within the MSC may vary, but the size of the MSC is constant. The MSC contains a maximum of 63 different audio and/or data subchannels. The subchannels are numbered on the basis of a so-called Channel Id from 0 to 62. Moreoover, the MSC may contain an Auxiliary Information Channel AIC, which has a fixed channel number 63. The AIC may carry the same type of information as the FIC.

At the transmitting end, in addition to audio services, the service supplier may also offer e.g. multimedia services, hypermedia services, file-based services and hypertext. From the audio information and data provided by the service suppliers, the DAB operator generates a DAB transmission signal, which comprises successive transmission frames as shown in the lower part of Fig 1.

In the receiver, the information channel FIC and the MSC, which contains the audio and data services, are separated from each other from the transmission frame. The subchannels are separated, channel-decoded and then passed on for further processing. From the FIC channel received, the customer obtains information about the services contained in the ensemble received and can thus select the service or services he/she wants. By combining subchannel service components in accordance with the application software, it is possible to compose e.g. a desired multimedia service.

As stated above, information can be transferred in packet mode, in which case data capacity can be reserved for service suppliers dynamically, or as a continuous stream. The maximum capacity in packet transfer is 1.728 Mbit/s. In continuous audio transfer, successive audio frames are transferred. Briefly speaking, to transfer information in data packets, the information data is first placed in the data field of a so-called data group. The data group contains header fields and after these a data field, in which the data to be transferred is placed. The length of the data field may vary and is at most 8191 bytes. The last field is the checksum of the data group. The data packets to be sent out to the transmission path are formed from the data group by simply chopping it into sections of equal length and placing each section in the data field of a data packet. Fill bits are used if the last data group section to be placed is shorter than the length of the data field of the data packet. The data packet length has one of four possible values, 24, 48, 72 or 96 bytes. Based on the packet headers, the data group can be assembled again in the receiver. Generally, a data group consists of the data fields of a number of packets transmitted in succession, but in the simplest case a single packet is sufficient to form a data group.

A continuous audio stream or data stream is transferred in frames, the structure of which is illustrated by Fig. 2. At the transmitting end, audio samples coming in at a frequency of 48 kHz and encoded into 16-bit format are divided into sub-bands, and the samples of the sub-bands are coded into the audio frame by making use of the masking effect of the human ear so that the incoming bit rate 768 kbit/s is reduced e.g. in the case of a mono channel to a rate of about 100 kbit/s. The four-byte header of the frame contains information intended for the decoders in the receiver, such as synchronisation data, bit rate data, and sampling frequency data. A bit allocation field coming after the checksum indicates how the bits are allocated to each one of the audio field sub-bands containing 36 encoded samples and which bits have been removed from the samples in making use of the masking effect. A scale factor selection information field indicates how the group of audio samples has been scaled (normalized) in the decoder. After this there is a field that contains the audio bits proper. The information in it corresponds to 24 ms of audio. The field contains 36 encoded sub-band audio samples divided into twelve triplets, each of which contains 3 sub-band samples. Thus, four triplets corresponds to 12 ms of audio. After this there are fill bits if the number of audio bits amounts to less than the audio field length. Finally there are an X-PAD and an F-PAD field, the meanings of which are described next.

Fig. 3 presents the last part of the audio frame. Each audio frame contains bytes that transmit data relating to the programme (Programme Associated Data). This data is in synchronism with the audio data in the frame. The PAD-bytes of successive frames make up a so-called PAD channel. The field consisting of the two last bytes, called the fixed PAD field, is intended for the transmission of real-time information related to audio, but it can also be used as a very slow data channel. The PAD channel can be extended by employing a so-called extended PAD field X-PAD, which is intended for the transmission of additional information to the listener, such as text associated with the audio, e.g. the lyrics of songs. The X-PAD field may be absent altogether, and its length in each frame can be four bytes, a so-called short X-PAD, in which case it is located in the frame area which is better protected against errors, indicated by the shading in the figure, or its length may change from frame to frame, in which case only a part (4 bytes) of it is in the well-protected area. Between the PAD fields there is a Scale Factor Error Check - Cyclic Redundancy Check field ScF-CRC associated with the audio field. The frame always has a fixed-length F-PAD field, and if an X-PAD field exists, its length is encoded in the F-PAD. The X-PAD field has at its beginning one or more contents indicators CI. The CI is a number which indicates the nature or application type of the data placed in the X-PAD data field or in its sub-fields. According to the specification, the maximum number of application types available is 287. Numbers 0, 2-11, 32 and 33 are defined under item 7.4.3 of the specification.

The DAB system allows the transmission of multimedia type services, but the multimedia techniques currently used are not adequate for this purpose. At present, each multimedia producer uses its own technical solutions for digital audio and video, presentation script language, coding, protocols, operating systems, etc. So far there is no standardized method for generating a complex, interactive multimedia presentation using the producer's computer, storing it in a data medium, transmitting it over a transfer network and reproducing the presentation on another computer. The producer has to store e.g. a multimedia book on a compact disc in numerous different formats, such as CD-I (Compact Disc Interactive), MPEG-1 and QuickTime (a system used by Apple). A recording in appropriate format is then transferred to an industry-standard computer PC, a MacIntosh or a Unix computer, for these to be able to present the multimedia book. Transfer over a network or data exchange between heterogeneous systems is not possible. The main problem is a lack of international standards for the creation and presentation of the contents of multimedia. In particular, the final multimedia script lacks conditional links and spatial as well as temporal relationships between content elements. For example, the JPEG and MPEG standards only describe the contents of information objects, but they cannot be used to describe the relationships between the objects in a multimedia presentation. To solve this problem, in other words, to define and standardize the structural information of a multimedia presentation, the ISO (International Organization for Standardization) has established a working group called MHEG (Multimedia and Hypermedia Information Coding Experts Group), which has made a proposal for a multimedia standard, known by the same designation.

In its philosophy, the standard follows the layered structure of the OSI model, in which the abstract syntax and the transfer syntax are separated from each other. The standard is based on an object-oriented approach. It has been developed in five parts, of which the first part, called ASN.1 (Abstract Syntax Notation 1), is a complete definition of objects, whereas the fifth part

MHEG-5 describes the implementation at application level, with special focus on TV applications. ASN.1 is also used in MHEG-5. MHEG-5 is defined in the proposed standard ISO/IEC CD 13522-5, September 20, 1995. An MHEG-5 application is composed of scenes and objects common to different scenes. A scene is used to present information (text, audio, video, and so on), whose behaviour is based on the triggering of events. For example, pressing a button visible on the screen starts a video sequence or activates the sound. At least one scene is active at any given instant. Navigating within a presentation thus means moving from one scene to another.

To make the present invention easier to understand, certain MHEG concepts are now briefly described. Links are objects which contain a trigger for triggering an event and a reference to an action object, which again contains a list of elementary events. Thus, a link is associated with a given event. When a certain condition is encountered, the event is triggered and the elementary events (e.g. the starting, running and closing of a video sequence) are executed in the order prescribed by the list of elementary events. A container created for the transfer contains a combination of MHEG objects, so it can be thought of as a complex object consisting of simple basic objects. The container may contain e.g. JPEG, MPEG and text files. Containers can be linked to each other. For the receiver to be able to present the received multimedia programme correctly, it must be provided with a certain software package, called the MHEG engine. It is a process or a number of processes that are able to interpret the encoded MHEG objects in accordance with the specification.

Running an MHEG application on-demand in a multimedia receiver connected to a fixed network is basically quite simple. The receiver first identifies the starting object in the received data, downloads it and prepares it. The starting object may be any one of the objects in the container. Usually this is the first scene. After the starting object has been prepared, one or more linked objects are triggered, and this may result in the loading of several objects referenced by the elementary event. This is also the way an audio stream is started. The receiver naturally starts the reception of an audio stream right from the beginning.

The proposed MHEG multimedia is excellently suited for use in the DAB system. In this case, the principle could be as illustrated by Fig. 4. At the transmitting end, the service supplier encodes his multimedia service components, which may have a different internal format, to convert them into objects consistent with the MHEG specification. The objects can be placed in containers. The DAB operator places the containers or objects in a DAB multiplex and transmission frames to be transferred via a packet channel and/or as continuous audio and data. The receiver decodes the sub-channels from the multiplex and passes the objects decoded according to the MHEG specification to an MHEG engine, which decodes the multimedia presentation from them.

In the DAB system, however, the situation is more problematic as compared with a fixed network. A multimedia presentation is very likely to start with audio. Shortly after the start of the audio stream comes a starting image, which may be a still picture. However, as DAB is a broadcasting system, the receiver is frequently switched to a multimedia service in the middle of a programme and therefore in the middle of an audio stream. The first part of the presentation is therefore not present in the memory of the receiver, so the starting image is missed and there are no mechanisms for invoking starting it. The only alternative is to wait for a retransmission so that reception can be started from the very beginning. However, there should be a starting mechanism that would allow multimedia reception even after transmission has already begun.

This invention presents a solution to the problem described above. The solution is characterized by what is said in the independent claims.

According to the invention, the audio stream is divided into successive segments of different lengths and the segments are marked. For the marking, a specific marker is provided at the boundary between segments. Segment boundaries indicate a change in the multimedia presentation. The change may be e.g. the disappearance of a still picture. A portion of a multimedia presentation that contains still pictures, video and text contains links. Besides the starting scene, such links are also present in other scenes. A marker in the audio stream activates a link in a given scene, whereupon the presentation continues as programmed by the producer. When the receiver is switched on, the decoder decodes the markers found in the audio stream and sends them to the MHEG engine. At the same time, the MHEG engine has received and decoded objects belonging to the presentation and generated a scene which is not displayed. The links in the scene are waiting to be activated, and when a marker associated with one of the links and acting as an excitation is decoded from the audio stream, it activates the link concerned. As a result, either the scene is displayed or the link causes a transition to another scene, which is displayed. The transition is invisible to the user, so the user perceives the presentation as starting with the right scene.

In the following, the invention is described in greater detail by referring to the attached drawings, in which:

Fig. 1 presents the levels of abstraction in the DAB system;

Fig. 2 presents an audio frame;

Fig. 3 presents the PAD fields of the audio frame;

Fig. 4 illustrates MHEG transmission;

Fig. 5 presents the F-PAD field;

Fig. 6a indicates how a marker is encoded in a short X-PAD field;

Fig. 6b indicates how a marker is encoded in a variable-length X-PAD field.

As is known, in a multimedia programme there must be some way to indicate the file that the receiver has to load first and from which the multimedia presentation is to be started. In the present application, the starting file is referred to as start-up file. In conjunction with the DAB system, this start-up file is preferably notified to the receiver via the data transfer protocol for multimedia files by the method described in patent application FI 954752 by the applicant. Other methods can be used as long as the receiver is enabled to find and load the start-up file. The start-up file contains a link which automatically starts the reception of the audio stream as well.

First, according to the invention, the start-up file contains a number of links associated with events that are awaiting to be triggered. When a trigger appears, the link activates certain events as determined by the producer, so the result of these events has been accurately defined.

Second, according to the invention, specific stream marker IDs are included in the audio stream. The length of a stream marker ID is two bytes or preferably three bytes. The stream marker IDs divide the audio stream into segments of varying lengths. The service supplier places the stream marker IDs in the PAD fields of the audio frames in such a way that the segment boundaries are certain clearly distinguishable changes in the multimedia presentation, e.g. the audio stream portion between any given stream marker IDs refers to a given scene and within this scene to a given still picture. In other words, as long as the still picture is visible, the sound decoded in the audio frames between the markers is to be heard via the speakers of the receiver.

Now, when the user switches the receiver to multimedia reception in the middle of a multimedia transmission, the MHEG engine of the receiver first finds the start-up file among the incoming files and decodes it as described in the above-mentioned patent application. It contains the mechanisms and references to the required files that are needed for the preparation of the first scene, which is then prepared by the MHEG engine. The scene is associated with so-called presentables, which are objects that the user can see or hear. However, these presentables are not activated as yet and the scene is therefore not displayed on the screen of the receiver. The start-up file also contains a command to start the reception of the audio stream associated with the multimedia, but in this case the reception begins in the middle of the audio stream. The first scene contains links that are triggered by a certain marker embedded in the audio stream. The receiver decodes the X-PAD fields of the audio frames and distinguishes the stream marker ID placed in the field and passes it to the MHEG engine. The MHEG engine directs the marker to the links, with the result that the marker triggers the events defined in at least one of the links. The events have been set in the MHEG language by the service supplier. They include loading the objects determined by the service supplier into the receiver and preparing them. The result is e.g. a new scene that the service supplier has meant to be displayed at this point of the audio stream. Its presentables are activated and the scene is displayed on the screen of the receiver. All the preceding actions are part of a chain of internal events in the programme that are not visible to the user. To the user's perception, the multimedia starts at the right point in relation to the audio. After this, the normal interactive procedure is followed. At this stage, objects that are no longer needed because the presentation has jumped from the starting scene directly to a later scene are cleared and removed.

As stated above, the stream marker ID is transferred in the X-PAD field of the audio frame. A stream marker ID is placed at each boundary between audio segments, in other words, the stream marker ID changes at each segment boundary. The stream marker ID refers to the audio information in the frame concerned as well as the audio information in subsequent frames until the stream marker ID changes. However, it is advantageous to place the same stream marker ID at regular intervals in other frames within the segment as well because this allows faster start-up of the MHEG presentation. It is not necessary to provide every frame with a stream marker ID. In this case the procedure could be such that the receiver decodes an odd number of successive stream markers and the MHEG engine carries out a majority vote, the resulting stream marker ID being then used to activate the link. This provides an advantage when in the vicinity of a segment boundary, because it prevents "premature" progress in the multimedia presentation.

The stream marker ID can be encoded in the X-PAD field in the manner shown in Fig. 5, 6a and 6b. Fig. 5 presents the F-PAD part defined by the specification that comes at the end of the audio frame. This part begins with a 2-bit field, F-PAD type. If it has the value "00", this means according to the specification that the first two bits in the following 6-bit data field are reserved for an X-PAD indicator. If the indicator bits are "01", this means that an X-PAD field is included and that it is a so-called short X-PAD comprising 4 bytes. If the bits are "10", this means that an X-PAD field is included and that the field is of variable length, called variable X-PAD. From the above information, the decoder detects the presence of an X-PAD field and also learns its type. It then examines the contents indicator CI of the X-PAD.

In the case of a short X-PAD, Fig. 6a, there is first an 8-bit field which is reserved for an application type indicator. According to the invention, this field is filled with the decimal number 1 (binary number 00000001) to indicate that the three data fields of the X-PAD contain a stream marker ID as used in the invention. The bit pattern of the stream marker ID can be selected by the service supplier. 24 bits provide a sufficient scope of variation. In the case of a short X-PAD, contents indicator value 00000001 thus means that the next three bytes contain a stream marker ID.

In the case of a variable X-PAD, the contents indicator is as illustrated by Fig. 6b. Its length is two bytes and the Length field indicates the number of bytes included in the X-PAD. In particular, if the length indicator has the value "000", this means that the X-PAD comprises four bytes. The maximum is 48 bytes. According to the specification, the next 5-bit field is reserved for the application type. According to the invention, this field is given the value of decimal 1 (binary number 00001). The "application type external" field is not in use. In the case of a variable X-PAD, contents indicator value 000 00001 thus means that the X-PAD field is four bytes long and contains a stream marker ID and that the stream marker ID is given in the first three bytes. The last byte is a checksum CRC with the polynome $x^8 + x^4 + x^3 + x^2 +1$ over the stream marker ID.

In the DAB specification, a meaning has already been defined for application type numbers 0, 2-11, 32 and 33, so the number of the application type referring to the stream marker ID must have a value other than those indicated above. The number 1 is still available and the applicant proposes that it be used for the purpose described in the present invention.

It is obvious to a person skilled in the art that technological development allows many different ways of implementing the basic idea of the invention. The invention and its embodiments are therefore not limited to the examples described above but may be varied within the framework of the claims.

Claims

1. Multimedia programme which has been produced in a specific programming language that tells how monomedia programmes are spatially and temporally linked to each other and in which programme a monomedia stream contains stream marker IDs, the reception of each of which is an event that triggers functions defined in a given link, and in which multimedia programme a number of objects form a scene intended to be displayed on a display device,

characterized in that

the monomedia stream is an audio stream and the stream marker IDs placed in it divide the audio stream into segments of variable length, each boundary between segments indicating a change in the multimedia presentation,

the objects forming a scene in the multimedia presentation have been provided with links whose triggering event is the reception of a stream marker ID associated with the link, so that the triggering results in a transition from one scene to another in the multimedia presentation.

2. Multimedia programme as defined in claim 1, **characterized** in that the programming language is MHEG (Multimedia and Hypermedia information coding Expert Group).

3. Transfer of a multimedia programme in a DAB broadcasting system, in which

the audio stream of the programme is transmitted in audio frames which have at their end a first field F-PAD of fixed length, which is intended for the transfer of data associated with the programme and contains data indicating whether a second field X-PAD for the transfer of data associated with the programme is present the at the end of the audio frame, said second field containing a contents indicator CI which indicates the nature of the data in the data field,

the rest of the multimedia components are transmitted as files, and the receiver assembles from the received programme a scene to be displayed on a display device,

characterized in that

the audio stream is divided into segments of variable length by providing those audio frames in the audio stream which involve a change in the multimedia presentation with an individual stream marker ID, which is decoded by the receiver and transferred to the software processing the multimedia presentation,

the objects forming a scene in the multimedia presentation contain links, the triggering event of at least one of which is the transfer of said individual stream marker ID to the software, such triggering causing the software to perform certain specified actions.

4. Transfer of a multimedia programme as defined in claim 3, **characterized** in that the specified actions cause the scene to be changed into a scene corresponding to the current audio stream.

5. Transfer of a multimedia programme as defined in claim 4, **characterized** in that, when the reception of the multimedia programme is started in the middle of the transmission, the first scene displayed on the display device is the scene corresponding to the current audio stream.

6. Transfer of a multimedia programme as defined in claim 3, **characterized** in that audio frames within the

segments also contain stream marker IDs, and that the stream marker ID at the beginning of the segment and those elsewhere in the segment refer to the same link.

7. Transfer of a multimedia programme as defined in claim 6, **characterized** in that, when the same segment contains several stream marker IDs, after removal of the CRC a majority vote is carried out and its result acts as a trigger that triggers the link.

8. Transfer of a multimedia programme as defined in claim 3, **characterized** in that the stream marker ID is placed in the second field X-PAD intended for the transfer of data associated with the programme.

9. Transfer of a multimedia programme as defined in claim 8, **characterized** in that the application type field included in the contents indicator CI contains an individual value indicating that the data field contains a stream marker ID and that the stream marker ID has been placed in the data field.

10. Transfer of a multimedia programme as defined in claim 3 or 9, **characterized** in that the stream marker ID is a 3-byte number.

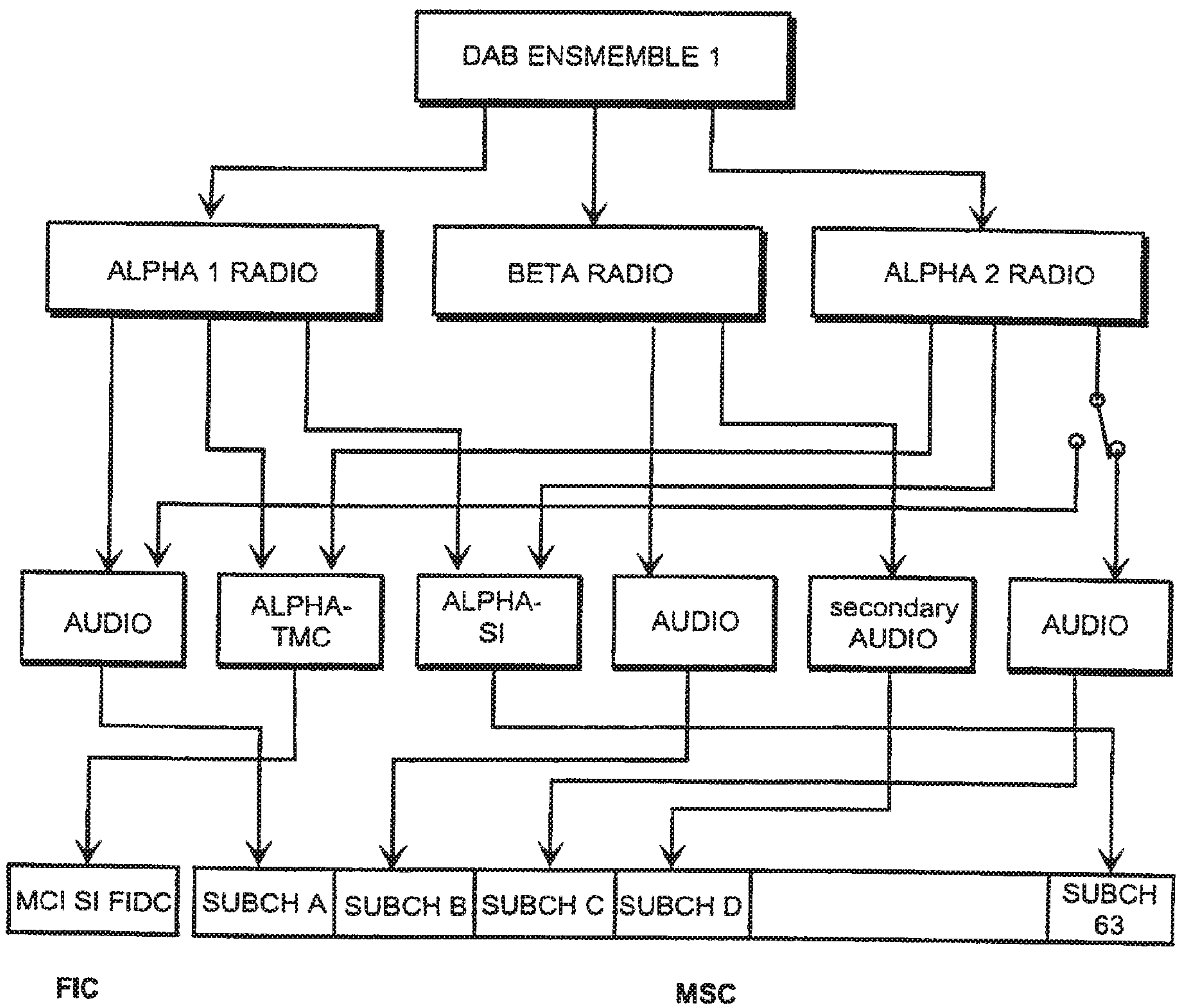

DAB ENSMEMBLE 1
ALPHA 1 RADIO
BETA RADIO
ALPHA 2 RADIO
AUDIO
ALPHA-TMC
ALPHA-SI
AUDIO
secondary AUDIO
AUDIO
MCI SI FIDC
SUBCH A
SUBCH B
SUBCH C
SUBCH D
SUBCH 63
FIC
MSC

Fig. 1

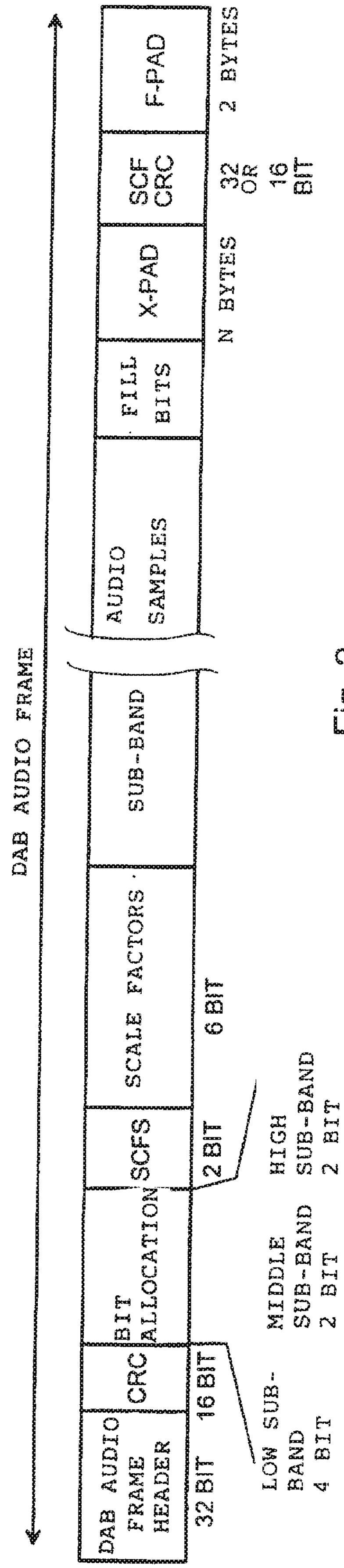

DAB AUDIO FRAME
DAB AUDIO FRAME HEADER
CRC
BIT ALLOCATION
SCFS
SCALE FACTORS
SUB-BAND
AUDIO SAMPLES
FILL BITS
X-PAD
SCF CRC
F-PAD
32 BIT
16 BIT
2 BIT
6 BIT
N BYTES
32 OR 16 BIT
2 BYTES
LOW SUB-BAND 4 BIT
MIDDLE SUB-BAND 2 BIT
HIGH SUB-BAND 2 BIT

Fig. 2

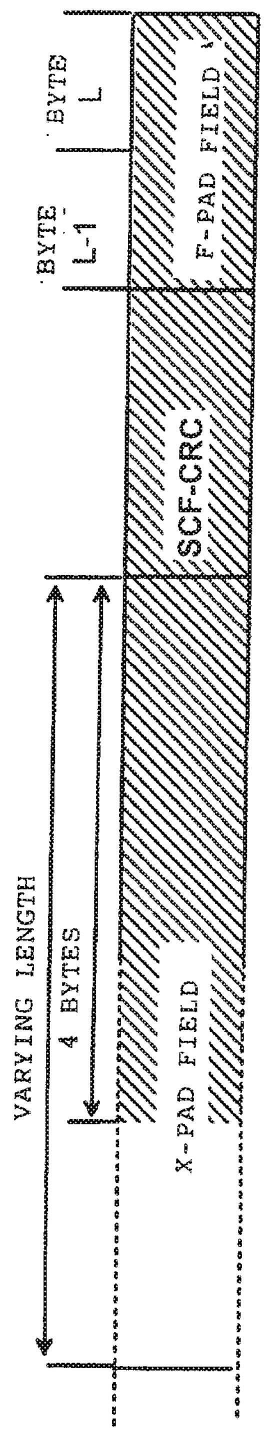

VARYING LENGTH
4 BYTES
X-PAD FIELD
SCF-CRC
BYTE L-1
BYTE L
F-PAD FIELD

Fig. 3

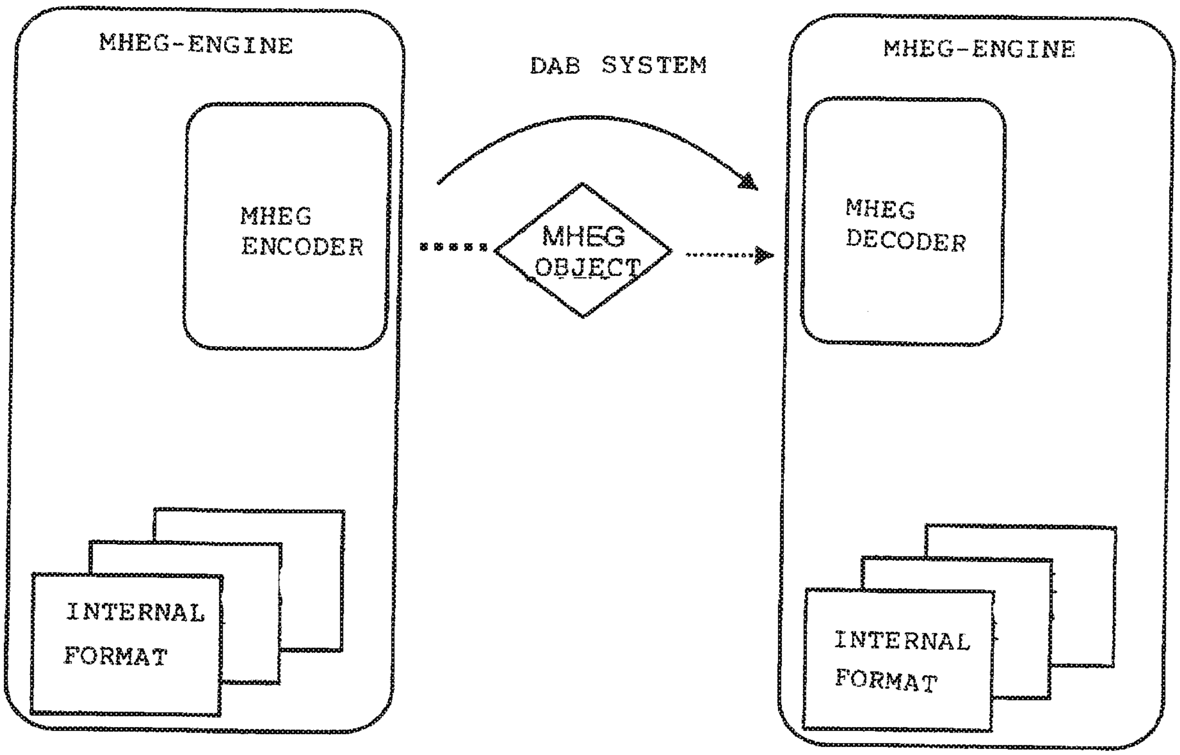


Fig. 4

BYTE L-1		BYTE L		
2 bits	6 bits	6 bits	1 bit	1 bit
F-PAD type	DATA FIELD OF BYTE L-1	DATA FIELD OF BYTE L	C1 Flag	Z

Fig. 5

8 bits	8 bits	8 bits
AppTy	MARK	ER
00000001		

Fig. 6 a

3 bits	5 bits	8 bits		
length	AppTy	AppTy ext		
000	00001			

Fig. 6 b

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 96/00594

A. CLASSIFICATION OF SUBJECT MATTER

IPC6: H04H 1/00
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC6: H04H, H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A,P	FR 2728089 A (ELECTRONICS AND TELECOMMUNICATIONS RESEARCH INSTITUTE-KR KOREA TELECOMMUNICATION AUTHORITY-KR), 14 June 1996 (14.06.96), page 6, line 12 - page 7, line 13, claim 1, abstract --	1-10
A,P	EP 0731575 A2 (NOKIA TECHNOLOGY GMBH), 11 Sept 1996 (11.09.96), see the whole document. -- -----	1-10

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" erlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
28 April 1997	29 -04- 1997
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Peter Hedman Telephone No. +46 8 782 25 00

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/FI 96/00594

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
FR 2728089 A	14/06/96	NONE	
EP 0731575 A2	11/09/96	FI 97840 B	15/11/96
		FI 951106 A	10/09/96

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04H 1/00

A1

(11) International Publication Number: WO 97/17776

(43) International Publication Date: 15 May 1997 (15.05.97)

(21) International Application Number: PCT/FI96/00595

(22) International Filing Date: 5 November 1996 (05.11.96)

(30) Priority Data:
955357 7 November 1995 (07.11.95) FI

(71) Applicant *(for all designated States except US)*: OY NOKIA AB [FI/FI]; Eteläesplanadi 12, FIN-00130 Helsinki (FI).

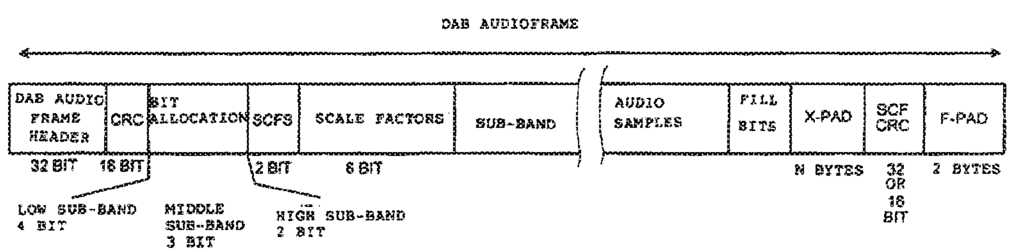
(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: SALOMÄKI, Ari [FI/FI]; Ojavainionkatu 10 D 28, FIN-33710 Tampere (FI). KASSLIN, Mika [FI/FI]; Torkkelinkatu 17 A 18, FIN-00500 Helsinki (FI). PEHKONEN, Jukka [FI/FI]; Uudenkyläntie 1 B 6, FIN-36720 Aitoo (FI).

(74) Agent: LUOTO, Kristian; Nokia Research Center, PL 45, FIN-00211 Helsinki (FI).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.
In English translation (filed in Finnish).

(54) Title: TRANSPORT OF AUDIO IN A DIGITAL BROADCASTING SYSTEM



(57) Abstract

In a digital broadcasting system (DAB), audio is transmitted as a continuous stream in audio frames. According to the invention, this is achieved by transmitting exactly one audio frame in the data field of each data group. Audio can be transferred as a file, in which case a given number of successive audio frames make up an audio file and the audio file is transferred in accordance with the file transfer protocol of the system. In this case, the transfer speed may vary. Audio can be transferred in packet mode as a continuous stream and the audio frames are transferred at the same speed by forming successive data groups from audio frames coming as a stream and by selecting the transfer speed of the data packets so that the transfer speed of an audio frame assigned to the data group is the same as the transfer speed of an audio frame transmitted as a continuous stream.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic	SD	Sudan
CF	Central African Republic		of Korea	SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

TRANSPORT OF AUDIO IN A DIGITAL BROADCASTING SYSTEM

The present invention relates to audio transport in a digital broadcasting system in which services can be transported as a continuous stream or in data packets.

The Digital Audio Broadcasting (DAB) system, which has been developed to allow an efficient utilization of frequency bands, uses a completely digital transmission path. The system is designed to replace the analogue broadcasting system commonly used at present, which is based on the use of frequency modulation. DAB defines a digital radio channel based on multiple carriers, which is applicable for the transmission of both audio and data services. A completely digital transmission channel may be either a continuous data stream channel or a packet channel. Packet transmission is more flexible and allows easier transmission of data units of finite length. The DAB system is presented in ETSI (European Telecommunication Standards Institute) standard 300 401, February, 1995.

From the user's point of view, the highest level of abstraction in the DAB system is called ensemble, Fig. 1. It contains all services existing in a given frequency band. A change from one ensemble to another is effected by tuning to a different frequency band, just as one changes channels in current FM radio reception. The ensemble is divided into services, exemplified in Fig. 1 by Alpha Radio 1, Beta Radio and Alpha Radio 2. In addition, there may be data services, although they are not shown in the figure. Each service is further divided into service components. A service component can be transported either via an audio channel or via a data channel. For comparison, let it be stated that FM radio contains only one service and one service component (audio) in each channel. At the lowest level, the transmission frame, whose duration is 24 ms or 96 ms depending on the DAB mode, consists of three chronologically consecutive parts. The first part is a synchronizing channel, which contains no service information. The next part is a fast information channel FIC, which has a mode-specific fixed length. The last part is a main service channel MSC, which contains all the subchannels. The position, size and number of subchannels within the MSC may vary, but the size of the MSC is constant. The MSC contains a maximum of 63 different audio and/or data subchannels. The subchannels are numbered on the basis of a so-called sub-Channel Id from 0 to 62. Moreover, the MSC may contain an auxiliary information channel AIC, which has a fixed channel number 63. The AIC may contain the same type of information as the FIC. The MSC information is transmitted using time interleaving such that in DAB mode I the MSC part of the frame is divided into four parts and these are placed in successive transmission frames. These parts are known as CIF (Common Interleaved Frame), so the MSC part of the frame in mode I contains four CIFs. In other modes no interleaving is used, so the MSC is the same as the CIF.

At the transmitting end, besides audio services, the service supplier may also provide data services and e.g. multimedia services. From the audio information and data supplied by the service suppliers, the DAB operator produces a DAB transmission signal, which consists of successive transmission frames such as those presented in the lower part of Fig. 1.

In the receiver, the information channel FIC and the channel MSC containing the audio and data services are separated from each other from the transmission frame. The subchannels are separated and channel decoded and then passed on for further processing. From the information in the received FIC channel, the user will know which services are included in the ensemble received and is thus able to select a desired service or services. By combining subchannel service components according to the application programme, it is possible to compose e.g. a desired multimedia service.

One advantage of the DAB system is that data capacity can be reserved for different service suppliers on a dynamic basis. The capacity may be 1.728 Mbit/s at most. The data is transmitted in packets as illustrated by Fig. 2, consisting of a header field, a data field and a checksum. The meanings of the fields are in accordance with the DAB standard. The packet header contains packet length data (Pkt Len), a continuity index (Cont Ind), first/last packet data (First/Last), an address (Pkt Address) identifying the service component, a command (Command) and the actual data field length (Data Len). The data field contains the actual data to be transmitted plus fill bits if required. At the end is the packet checksum (Pkt CRC).

By combining the data fields of packets in the receiver, a so-called data group is formed, Fig. 2B. The packets are formed at the transmitting end from the data group by simply cutting it into sections and placing each section into the data field of a data packet. Generally a data group consists of the data fields of a number of consecutive packets transmitted. In the simplest case, one packet is sufficient to form a data group.

The data group is formed as illustrated by Fig. 3. The meanings of the abbreviations of the data group header and session header fields are as indicated in the table below:

Data Group header		Session header	
EXT FL	extension flag	LAST FL	last
CRC FL	CRC flag	SEG NUM	segment number
SES FL	session flags	RFA	reserved for future applications
DG TYPE	data group type	LEN IND	length of next address field
CONT IND	continuity index	ADDR FIELD	end user's address
REP IND	repetition index		
EXT FIELD	extension field		

These header fields are followed by the actual data and the data group checksum DG CRC.

A continuous audio stream is transmitted in frames having a structure as illustrated by Fig. 4. At the transmitting end, 16-bit PCM coded audio samples coming at a frequency of 48 kHz are divided into sub-bands and the samples of the sub-bands are encoded into the audio frame by making use of the masking effect of the human ear so that the incoming bit rate 768 kbit/s is reduced e.g. in the case of a mono channel to a rate of about 100 kbit/s. The four-byte header of the frame contains information intended for the decoders in the receiver, such as synchronization data, bit rate data, and sampling frequency data. A bit allocation field coming after the checksum indicates how the bits are allocated to each one of the audio field sub-bands containing 36 coded samples and which bits have been removed from the samples in making use of the masking effect. A Scale Factor Selection Information field indicates how the group of audio samples has been scaled (normalized) in the decoder. After this there is a field that contains the audio bits proper. The information in it corresponds to 24 ms of audio. The field contains 36 encoded sub-band audio samples divided into twelve triplets, each of which contains 3 sub-band samples. Thus, four triplets corresponds to 12 ms of audio. Next there are fill bits if the number of audio bits amounts to less than the audio field length. Finally there are an X-PAD and an F-PAD field, which transmit Programme Associated Data (PAD). This data is in synchronism with the audio of the frame. The PAD bytes of successive frames make up a so-called PAD channel.

The audio part of multimedia is proposed to be transmitted in audio frames, yet there may be certain reasons to transmit audio in packet mode as well. Packet mode transmission could in principle be applied e.g. to send audio files, which would first be stored in the memory of the receiver and then played back via the speakers at the appropriate time during the multimedia presentation. This type of audio transport has the advantage that the file can be transmitted at any bit rate, so the transport rate need not be the same as the fixed bit rate which is used in the transmission of audio frames or an audio stream and which has been assigned a number of allowed bit rate values in the DAB specification.

A disadvantage with this type of transmission is that the audio file has to be stored in memory if the transport rate is lower than the audio stream bit rate, or it needs to be buffered if the transport rate is higher than the audio stream bit rate. In the former case, playback cannot be started immediately upon reception of the file, whereas in the case of buffering, playback can be started immediately. However, the disadvantage is not a real one, because in most of the possible practical applications there is no need to transmit an audio file at the audio stream bit rate. The real problem is that the audio stream in audio frames cannot be transmitted in packet mode because there is no mechanism for indicating the audio frame boundaries.

However, there are certain applications in which it is desirable to implement audio transfer in real time packet mode. Real time packet transmission here means that the bit rate is the same in the transmission of both the audio stream and the audio packets, and that it would be possible in some way to extract from the packets the same information that is contained in the audio frames. This principle could be applied e.g. to locate bit errors in audio frames by comparing the received audio frame with the checksum CRC of the received packets that carry the same audio information and with the checksum CRC of the data group formed from the packets. In this way, the audio samples in the audio frames would end up being covered by CRC checking. Using packets for this purpose would involve extra auxiliary signalling of audio, which might be acceptable as a temporary solution. If audio transmission is to be protected on a lasting basis and as efficiently as packet transmission is, it would of course be preferable to improve the error protection of the audio bits in audio frames directly.

Another possible application of real time transfer of audio packets is found in the case where audio is to be addressed to a given user group only. This application makes use of the fact that the packets contain an address. As shown in the table above, the session header contains a field reserved for the end user's address, ADDR FIELD. This could be utilized to direct audio information to a given group, allowing real time packet transmission to be used as an information channel for different user groups. The DAB specification also defines the transmission of announcements via the fast information channel FIC. The definition describes the interruption of an active broadcast-type service by an announcement, but the transmission profile for the announcement can only be defined by specifying the services to be interrupted. There is no way to address the announcement to a given user group only.

Therefore, the object of the present invention is to achieve audio transport in packet mode so as to allow both addressed audio transport and indication of audio frames in a packet stream.

This object is achieved in the manner described in the independent claim.

According to the first embodiment of the invention, audio is transmitted over a packet channel as a file. An audio frame is placed in the data field of a data group. Thus, one segment in the file transfer protocol corresponds to one audio frame. From the packets received through the channel, a data group is assembled in the normal manner, so the data field of the data group will contain a file segment, which in this case is an audio frame. To enable the audio frames obtained from the data field to be arranged in the correct chronological order in the receiver, a file transfer protocol needs to be used. Since the transfer can be performed at any speed, the frames have to be stored or buffered in the receiver prior to presentation.

According to the second embodiment, the audio frames are transmitted over a packet channel in the form of a continuous stream. An audio frame is placed in the data field of a data group. The transmission speed is exactly the same as the audio channel bit rate. In this case, individual data groups are transmitted at exactly the correct pace, so the transmission consists of an endless train of data groups. Therefore, no file transfer protocol is needed. No buffering needs to be used, but the receiver presents the audio frames directly as it receives data groups from the packet channel.

The invention is illustrated by the attached drawings, in which

Fig. 1 presents a known DAB hierarchy,
Fig. 2a presents the structure of DAB packets,
Fig. 2b shows how a data group is formed from packets,
Fig. 3 presents the structure of a data group,
Fig. 4 presents a DAB audio frame, and
Fig. 5 illustrates the use of an IDG.

According to the invention, to allow the audio frame boundaries to be clearly indicated when audio information is transmitted in packet mode, one and only one audio frame in its entirety, including the PAD part, is assigned to each data group.

According to the first embodiment, the audio frames are transmitted as an audio file, in other words, the audio, which has a beginning, a duration and an end, constitutes one file. In accordance with the file transfer principle used in DAB, the file is divided into segments and each segment is placed in the data field of a data group. The protocol will be briefly described below. According to the invention, a segment is exactly equivalent to an audio frame. Since the audio is transmitted as a file, it is necessary to use a file transfer protocol to enable the receiver to arrange the data groups assembled from the packets into the correct sequence. The file transfer speed may be higher than, the same as or lower than the audio stream bit rate.

The file transfer protocol can be implemented according to the general basic principle proposed by the Eureka-147 project, in which each segment forms one data group. Successive segments of the file are numbered sequentially so that the first segment number in the session header is 0. The last segment of the file is indicated by a flag in the LAST field of the session header of the data group formed from it. The receiver receives the data packets and forms data groups out of them. If its checksum indicates that bit errors occurred during the transfer, the receiver picks up the data packets of the relevant data group from the retransmission of the file.

To enable the receiver to pick up the correct files from the packet stream transmitted and identify the file type in question so as to ensure proper file management, the Eureka-147 project includes a proposition that an additional special Information Data Group (IDG) be created. This is a file transfer descriptor, i.e. it provides the necessary information about the file it refers to and it is multiplexed with the file segments.

Fig. 5 illustrates the idea of the IDG in file transfer. One IDG is associated with only one file. It is placed at least at the beginning of the packet stream relating to the file, i.e. at the beginning of the file transfer, but IDGs can also be placed in the middle of the packet stream, in other words, the IDG may appear at any point during the file transfer or the IDG can be transmitted some time before the actual file transfer, so it can be used to announce a file transfer before it is started. In Fig. 5, files X, Y and Z are transferred and the IDGs referring to them are identified by corresponding letters. The important thing that can be accomplished by means of the IDG is contained in its data field, which is called the 'file descriptor'. The file descriptor can be used to give the receiver the required detailed information about the file to be transferred. The file descriptor includes a field containing so-called transfer parameters (T-parameters). A T-parameter called File Type declares the type of the file, so the application software of the receiver is able to decide which algorithm to use for file analysis and interpretation of its contents.

The applicant proposes that a new file type parameter called "DAB audio" be added to let the receiver know that the incoming file announced by the IDG is an audio file.

According to the DAB specification, the information about the services is transmitted over the fast information channel (FIC). It is used to announce the location and nature of the service components. A service description for each service is placed in a separate field, which contains a parameter field called service component description. One of the parameters is service component type and this parameter is set to File Transfer.

From the FIC channel information the receiver thus learns that a file is being transferred and from the IDG information that the file is an audio file. The receiver is therefore able to receive the file, decode the audio and present it. Depending on the file transfer speed, the receiver must either store the audio file or use buffering for it.

According to the second embodiment, the audio frames are transmitted in packet mode, yet as a continuous audio stream. The transfer speed is exactly the same as it would be if audio frames were used. At the transmitting end, as audio frames are coming as a continuos stream, each audio frame is first assigned to a data group, then packets are formed from the data group and transmitted. The audio frame also contains the PAD. Thus, the transmission is not a file transfer procedure as in the first embodiment, so no file transfer protocol is needed. Therefore, no information data groups IDG are needed, either. An audio data group may never cross the CIF boundary, so the entire data group must be transferred in a single common interleaved frame CIF. In other words, the data group must be transferred in a single transmission frame. If repetition is used at the data group level, then the data group and all the repetitions of it must be transmitted in a single CIF, i.e. each repetition is transmitted in a single CIF.

When a data group is formed, a session header is not necessarily needed, but it is advisable to use it because it contains an address field ADDR FIELD, which is intended for the end user's address. By means of this address, an audio transmission can be addressed to a desired user group. If the session header is used, then the SEG NUM field is used as a counter that is incremented by one on every data group. This helps the receiver keep in synchronism, because if a data group fails to be transmitted or is defective, the playback is delayed by the duration of an audio frame. The flag in the LAST FL field is always zero because the transmission is a continuous audio stream, albeit in packet mode.

The data group can be interleaved with other packet mode service components in the same subchannel, but the data group must still fit into a single CIF. If a data group is interleaved with other service components carrying audio data groups in the same subchannel, then all the data groups must be transmitted in a single CIF.

As in the case of the first embodiment, the applicant proposes that the service component type parameter "DAB Audio Stream" be sent in the service component description field in the fast information channel FIC.

It is obvious to a person skilled in the art that technological development permits many different ways of implementing the basic idea of the invention. The invention and its embodiments are thus not limited to the examples described above but may be varied within the framework of the claims.

Claims

1. Procedure for the transfer of audio in a digital broadcasting system, in which an audio programme transmitted as a continuous stream is transferred in the form of audio frames and in which, in packet mode, the information to be transmitted is assigned to the data field of a data group (DG) and the data group is divided into segments which are placed in the data fields of the data packets,

characterized in that the audio is transmitted in packet mode by placing the audio frame in the data field of the data group.

2. Procedure as defined in claim 1, **characterized** in that a certain number of successive audio frames make up an audio file and the audio file is transferred in accordance with the communication protocol of the system.

3. Procedure as defined in claim 2, **characterized** in that the transfer rate of the audio frames can be freely selected within the limits of the packet transfer speed of the system.

4. Procedure as defined in claim 1, **characterized** in that successive data packets are formed from successive audio frames and that the transfer speed of the data packets transferred as a continuous train is so selected that the transfer speed of an audio frame placed in the data group is the same as the transfer speed of an audio frame transmitted as a continuous stream.

5. Procedure as defined in claim 5, **characterized** in that the entire data group is transmitted in a single transmission frame, in which case the data group is not time-interleaved into several transmission frames.

6. Procedure as defined in claim 1, **characterized** in that a segment number field in the data group header is used as a counter which is incremented by one each time a data group is transmitted.

7. Procedure as defined in claim 1, **characterized** in that an address field in the data group header is used to address a given user group.

8. Procedure as defined in claim 1, **characterized** in that the receiver is informed via mechanisms consistent with the system that the data field of the data group contains audio information.

9. Procedure as defined in claim 1, **characterized** in that the digital broadcasting system is DAB.

10. Procedure as defined in claim 9, **characterized** in that the information that the data field of the data group contains audio information is conveyed in the File Type field of the Information Data Group (IDG).

11. Procedure as defined in claim 9, **characterized** in that the information that the audio is transferred as a file is conveyed via the Fast Information Channel (FIC) by setting the service component type parameter to File Transfer.

12. Procedure as defined in claim 9, **characterized** in that the information that the audio is transferred as an audio stream is conveyed via the Fast Information Channel (FIC) by setting the service component type parameter to DAB Audio Stream.

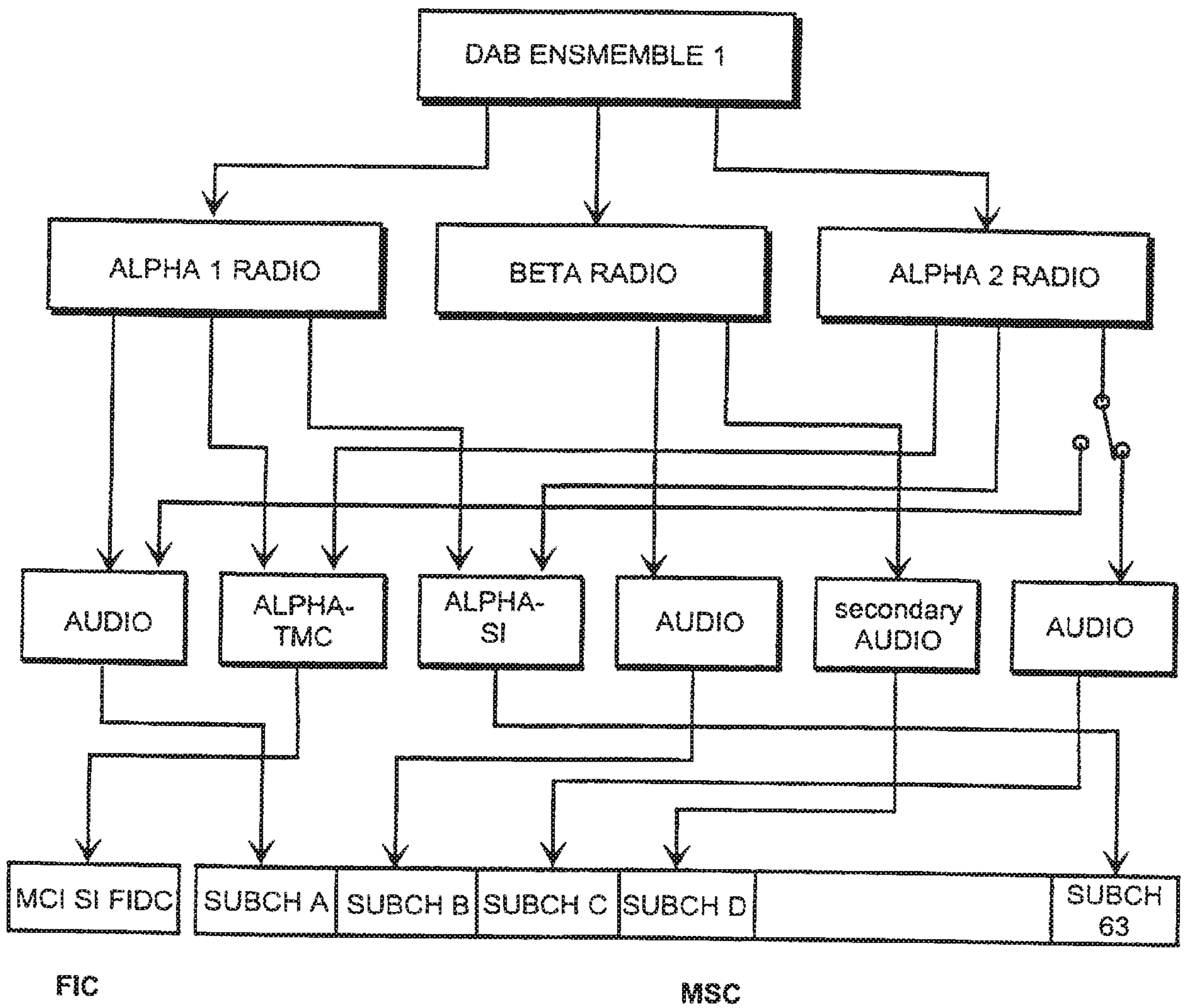

DAB ENSMEMBLE 1
ALPHA 1 RADIO
BETA RADIO
ALPHA 2 RADIO
AUDIO
ALPHA-TMC
ALPHA-SI
AUDIO
secondary AUDIO
AUDIO
MCI SI FIDC
SUBCH A
SUBCH B
SUBCH C
SUBCH D
SUBCH 63
FIC
MSC

Fig. 1

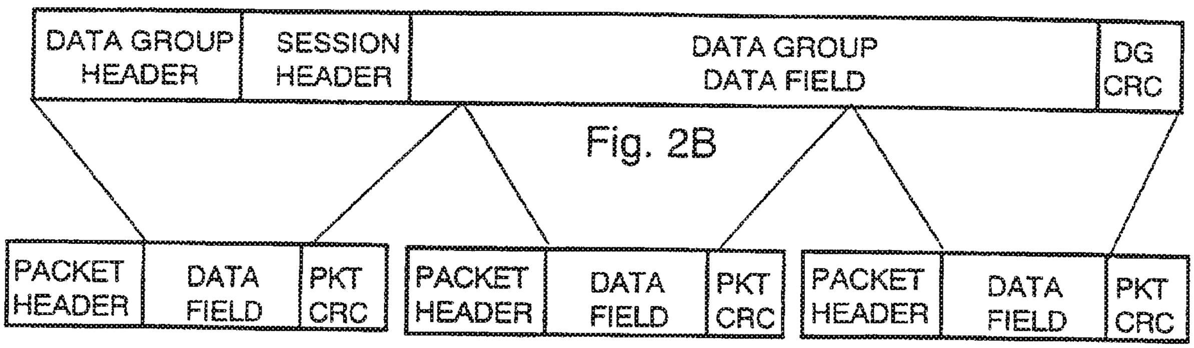


Fig. 2B

Fig. 2A

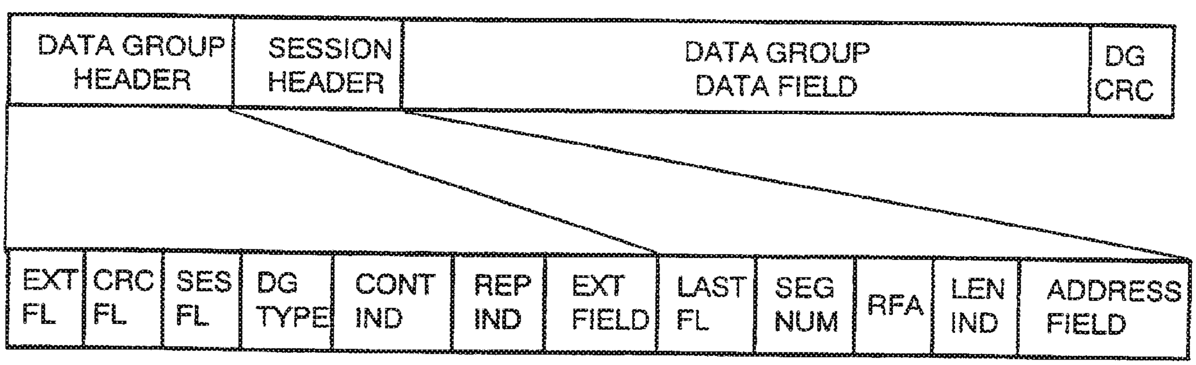


Fig. 3

3/3

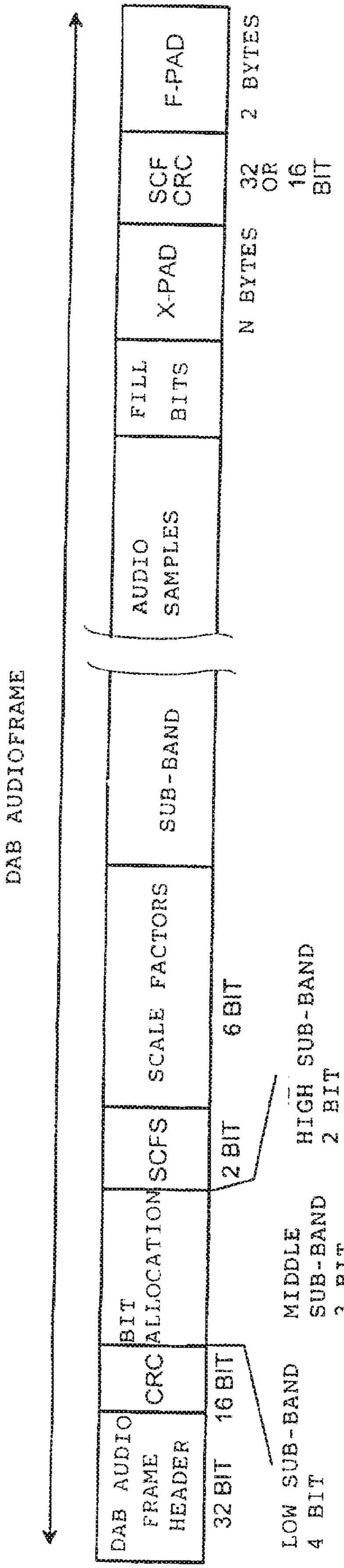

DAB AUDIOFRAME
DAB AUDIO FRAME HEADER
CRC
BIT ALLOCATION
SCFS
SCALE FACTORS
SUB-BAND
AUDIO SAMPLES
FILL BITS
X-PAD
SCF CRC
F-PAD
32 BIT
16 BIT
2 BIT
6 BIT
N BYTES
32 OR 16 BIT
2 BYTES
LOW SUB-BAND 4 BIT
MIDDLE SUB-BAND 3 BIT
HIGH SUB-BAND 2 BIT

Fig. 4

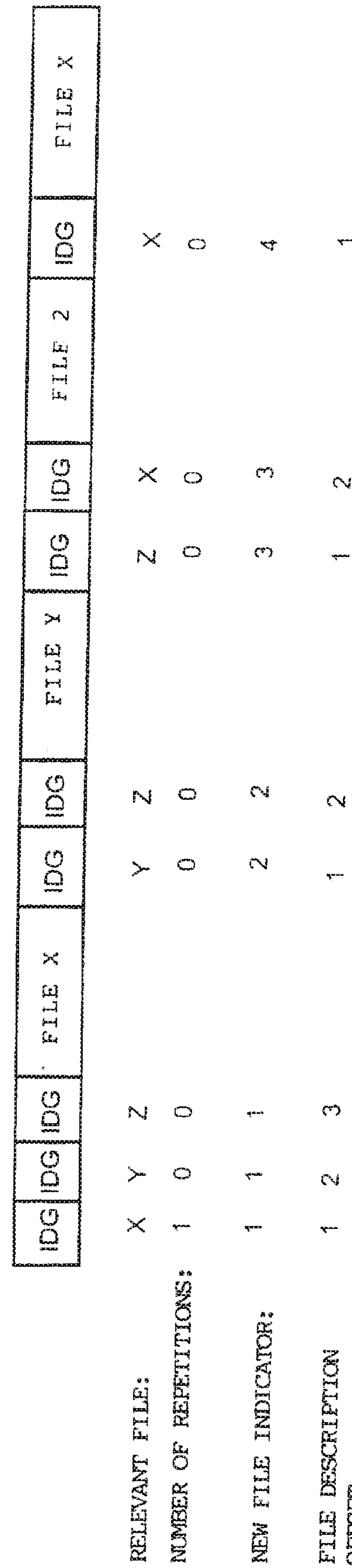

IDG
IDG
IDG
FILE X
IDG
IDG
FILE Y
IDG
IDG
FILE 2
IDG
FILE X
RELEVANT FILE:
NUMBER OF REPETITIONS:
NEW FILE INDICATOR:
FILE DESCRIPTION OFFSET:
X Y Z
1 0 0
1 1 1
1 2 3
Y Z
0 0
2 2
1 2
Z X
0 0
3 3
1 2
X
0
4
1

Fig. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 96/00595

A. CLASSIFICATION OF SUBJECT MATTER

IPC6: H04H 1/00
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC6: H04H, H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	DE 4422015 C1 (ROBERT BOSCH GMBH), 3 August 1995 (03.08.95), column 4, line 29 - column 5, line 19 --	1-7
A	FR 2718905 A1 (FRANCE TELECOM), 20 October 1995 (20.10.95), claim 1, abstract --	1-7
A	US 5010549 A (KATOU ET AL), 23 April 1991 (23.04.91), column 1, line 32 - line 68 -- -------	1

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" erlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
29 April 1997	30 -04- 1997
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Christian Rasch Telephone No. +46 8 782 25 00

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/FI 96/00595

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
DE 4422015 C1	03/08/95	AU 2611095 A	05/01/96
		CA 2192958 A	21/12/95
		EP 0765547 A	02/04/97
		WO 9534965 A	21/12/95
FR 2718905 A1	20/10/95	CA 2187867 A	26/10/95
		EP 0757878 A	12/02/97
		WO 9528810 A	26/10/95
US 5010549 A	23/04/91	JP 1272247 A	31/10/89

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04N	A2	(11) International Publication Number: WO 97/22201 (43) International Publication Date: 19 June 1997 (19.06.97)

(21) International Application Number: PCT/US96/19226

(22) International Filing Date: 12 December 1996 (12.12.96)

(30) Priority Data:
60/008,531 12 December 1995 (12.12.95) US

(71)(72) Applicants and Inventors: CAMPBELL, Roy, H. [US/US]; University of Illinois at Champaign-Urbana, Dept. of Computer Science, 1304 W. Springfield, Urbana, IL 61801 (US). TAN, See-Mong [SG/US]; University of Illinois at Champaign-Urbana, Dept. of Computer Science, 1304 W. Springfield, Urbana, IL 61801 (US). XIE, Dong [CN/NO]; University of Illinois at Champaign-Urbana, Dept. of Computer Science, 1304 W. Springfield, Urbana, IL 61801 (US). CHEN, Zhigang [CN/US]; University of Illinois at Champaign-Urbana, Dept. of Computer Science, 1304 W. Springfield, Urbana, IL 61801 (US).

(74) Agents: BERNSTEIN, Frank, L. et al.; Sughrue, Mion, Zinn, Macpeak & Seas, Suite 800, 2100 Pennsylvania Avenue, N.W., Washington, DC 20037-3202 (US).

(81) Designated States: CN, JP, KR, RU, US, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: METHOD OF AND SYSTEM FOR TRANSMITTING AND/OR RETRIEVING REAL-TIME VIDEO AND AUDIO INFORMATION OVER PERFORMANCE-LIMITED TRANSMISSION SYSTEMS

(57) Abstract

The architecture of numerous networks, including the Internet with its World Wide Web (WWW) browsers and servers, support full file transfer for document retrieval. In order for the WWW to support continuous media, it is necessary to transmit video and audio on demand and in real-time, as well as new protocols for real-time data. The invention extends the architecture of the WWW to encompass the dynamic, real-time information space of video and audio. The inventive method, called Vosaic, short for Video Mosaic, incorporates real-time video and audio into standard hypertext pages and which are displayed in place. Video and audio transfers occur in real-time; there is no file retrieval latency. The video and audio result in compelling Web pages. Real-time video and audio data can be effectively served over the present day Internet with the proper transmission protocol. The invention includes a real-time protocol, called a video datagram protocol (VDP), for handling real-time video over the WWW. VDP minimizes inter-frame jitter and dynamically adapts to the client CPU load and network congestion. The video server in accordance with the invention dynamically changes transfer protocols, adapting to the request stream. The invention also is applicable to other networks using Internet-type protocols such as TCP/IP, including local area networks, metropolitan area networks, and wide area networks.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

METHOD OF AND SYSTEM FOR TRANSMITTING AND/OR RETRIEVING REAL-TIME VIDEO AND AUDIO INFORMATION OVER PERFORMANCE-LIMITED TRANSMISSION SYSTEMS

FIELD OF THE INVENTION

The present invention relates to a method of and system for transmitting and/or retrieving real-time video and audio information. The inventive method compensates for congested conditions and other performance limitations in a transmission system over which the video information is being transmitted. More particularly, the invention relates to a method of transmitting and/or retrieving real-time video and audio information over the Internet, specifically the World Wide Web.

BACKGROUND OF THE INVENTION

"Surfing the Web" has entered the common vocabulary relatively recently. Individuals and businesses have come to use the Internet both for electronic mail (e-mail) and for access to information, commonly over the World Wide Web (WWW, or the Web). As modem speeds have increased, so has Web traffic.

Web browsers, such as National Computer Security Association (NCSA) Mosaic, allow users to access and retrieve documents on the Internet. These documents most often are written in a language called HyperText Markup Language (HTML). Traditional information systems design for World Wide Web clients and servers has concentrated on document retrieval and the structuring of document-based information, for example, through hierarchical menu systems as are used in Gopher, or links in hypertext as in HTML.

Current information systems architecture on the Web has been driven by the static nature of document-based information. This architecture is reflected in the use of the file transfer mode of document retrieval and the use of stream-based protocols, such as TCP. However, full file transfer and TCP are unsuitable for continuous media, such as video and audio, for reasons which will be discussed in greater detail below.

The easy-to-use, point-and-click user interfaces of WWW browsers, first popularized by Mosaic, have been the key to the widespread adoption of HTML and the World Wide Web by the entire Internet community. Although traditional WWW browsers perform commendably in the static information spaces of HTML documents, they are ill-suited for handling continuous media, such as real time audio and video.

Earlier Web browsers, such as Mosaic, required a user to wait until a document had been retrieved completely before displaying the document on the screen. Even at the faster transfer speeds which have been become possible in recent years, the delay between retrieval request and display has been frustrating for many users. Particularly in view of the astronomical increase in Internet traffic, during especially busy times, congestion over the Internet has negated at least some of the speed advantages users have obtained by getting faster modems.

Video and audio files tend to be much larger than document files in many instances. As a result, the delay involved in waiting for an entire file to download before it is displayed is even greater for video and audio files than for document files. Again, during busy times, Internet congestion would make the delays intolerable. Even in networks which are separate from the Internet, transmission of sizable video and audio files can result in long waits for file transfer prior to display.

Multimedia browsers such as Mosaic have been excellent vehicles for browsing information spaces on the Internet that are made up of static data sets. Proof of this is seen in the phenomenal growth of the Web. However, attempts at the inclusion of video and audio in the current generation of multimedia browsers have been limited to transfer of pre-recorded and canned sequences that are retrieved as full files. While the file transfer paradigm is adequate in the arena of traditional information retrieval and navigation, it becomes cumbersome for real time data. The transfer times for video and audio files can be very large. Video and audio files now on the Web take minutes to hours to retrieve, thus severely limiting the inclusion of video and audio in current Web pages, because the latency required before playback begins can be unacceptably long. The file transfer method of browsing also assumes a fairly static and unchanging data set for which a single uni-directional transfer is adequate for browsing some piece of information. Real time sessions such as videoconferences, on the other hand, are not static. Sessions happen in real time and come and go over the course of minutes to days.

The Hypertext Transfer Protocol (HTTP) is the transfer protocol used between Web clients and servers for hypertext document service. The HTTP uses TCP as the primary protocol for reliable document transfer. TCP is unsuitable for real time audio and video for several reasons.

First, TCP imposes its own flow control and windowing schemes on the data stream. These mechanisms effectively destroy the temporal relations shared between video frames and audio packets.

Second, unlike static documents and text files, in which data loss can result in irretrievable corruption of the files, reliable message delivery is not required for video and audio. Video and audio streams can tolerate frame losses. Losses are seldom fatal, although of course they can be detrimental to picture and sound quality. TCP retransmission, a technique which facilitates reliable document and text transfer, causes further jitter and skew internally between frames and externally between associated video and audio streams.

Progress has been made in facilitating transfer of static, document-based information. Web browsers such as Netscape(tm) have enabled documents to be displayed as they are retrieved, so that the user does not have to wait for the entire document to be retrieved prior to display. However, the TCP protocol which is used to transfer documents over the Web is not conducive to real-time display of video and audio information. Transfers of such information over TCP can be herky-jerky, intermittent, or delayed.

Several products have attempted to combine real time video with Web browsers like Netscape(tm) by invoking external player programs. This approach is clumsy, using standard TCP/IP Internet protocols for video retrieval. Also, external viewers have not fully integrated video into the Web browser.

Several commercial products, such as VDOlive and Streamworks, allow users to retrieve and view video and audio in real time over the World Wide Web. However, these products use either vanilla TCP or UDP for network transmission. Without resource reservation protocols in use within the Internet, TCP or UDP alone do not suffice for continuous media. Adaptable and media-specific protocols are required. Video and audio can also only be viewed in a primitive, linear, VCR-mode. The issues of content preparation and reuse are also not addressed.

Sun Microsystem's HotJava product enables the inclusion of animated multimedia in a Web browser. HotJava allows the browser to download executable scripts written in the Java programming language. The execution of the script at the client end enables the animation of graphic widgets within a Web page. However, HotJava does not employ an adaptive algorithm that is customized for video transfer over the WWW.

While the foregoing problems of video and audio transmission over networks have been discussed in the context of the Internet, the problems are by no means limited to the Internet. Any network which experiences congestion, or has computers connected to it which experience excessive load, can encounter the same difficulties when transferring video and audio files. Whether the network is a local area network (LAN), a metropolitan area network (MAN), or a wide area network (WAN), transmission congestion and processor load limitations can pose severe difficulties for video and audio transmission using current protocols.

In view of the foregoing, it would be desirable to reduce the delays in display of video and audio files over networks, including LANs, MANs, WANs, and/or the Internet.

It also would be desirable to provide a system which enables real-time display of video and audio files over LANs, MANs, WANs, and/or the Internet.

Moreover, multiple views of the same video and audio should be supported. Parts of a video and audio clip, or the whole clip, can be used for different purposes. A single physical copy of a large video and audio document should support different access patterns and uses. All or part of the original continuous media document should be contained within other documents without copying. Content preparation would be simplified, and the flexible reuse of video content would be efficiently supported.

SUMMARY OF THE INVENTION

The inventors have concluded that to truly support video and audio in the WWW, one requires:

1) the transmission of video and audio on-demand, and in real time; and

2) new protocols for real time data.

The inventors' research has resulted in a technique that the inventors call Vosaic, short for Video Mosaic, a tool that extends the architecture of vanilla NCSA Mosaic to encompass the dynamic, real time information space of video and audio. Vosaic incorporates real time video and audio into standard Web pages and the video is displayed in place. Video and audio transfers occur in real time; as a result, there is no retrieval latency. The user accesses real time sessions with the familiar "follow-the-link" point and click method that has become well-known in Web browsing. Mosaic was considered to be a preferred software platform for the inventors' work at the time the invention was made because it is a widely available tool for which the source code is available. However, the algorithms which the inventors have developed are well-suited for use with numerous Internet applications, including Netscape(tm), Internet Explorer(tm), HotJava(tm), and a Java-based collaborative work environment called Habanero. Vosaic also is functional as a stand-alone video browser. Within Netscape(tm), Vosaic can work as a plug-in.

In order to incorporate video and audio into the Web, the inventors have extended the architecture of the Web to provide video enhancement. Vosaic is a vehicle for exploring the integration of video with hypertext documents, allowing one to embed video links in hypertext. In Vosaic, sessions on the Multicast Backbone (Mbone) can be specified using a variant of the Universal Resource Locator (URL) syntax. Vosaic supports not only the navigation of the Mbone's information space, but also real time retrieval of data from arbitrary video servers. Vosaic supports the streaming and display of real time video, video icons and audio within a WWW hypertext document display. The Vosaic client adapts to the received video rate by discarding frames that have missed their arrival deadline. Early frames are buffered, minimizing playback jitter. Periodic resynchronization adjusts the playback to accommodate network congestion. The result is real time playback of video data streams.

Present day httpd ("d" stands for "daemon") servers exclusively use the TCP protocol for transfers of all document types. Real time video and audio data can be effectively served over the present day Internet and other networks with the proper choice of transmission protocols.

In accordance with the invention, the server uses an augmented Real Time Protocol (RTP) called Video Datagram Protocol (VDP), with built-in fault tolerance for video transmission. VDP is described in greater detail below. Feedback within VDP from the client allows the server to control the video frame rate in response to client CPU load or network congestion. The server also dynamically changes transfer protocols, adapting to the request stream. The inventors have identified a forty-four-fold increase in the received video frame rate (0.2 frames per second (fps) to 9 fps) with VDP in lieu of TCP, with a commensurate improvement in observed video quality. These results are described in greater detail below.

On demand, real time video and audio solves the problem of playback latency. In Vosaic, the video or audio is streamed across the network from the server to the client in response to a client request for a Web page containing embedded videos. The client plays the incoming multimedia stream in real time as the data is received in real time.

However, the real time transfer of multimedia data streams introduces new problems of maintaining adequate playback quality in the face of network congestion and client load. In particular, as the WWW is based on the Internet, resource reservation to guarantee bandwidth, delay or jitter is not possible. The delivery of Internet protocol (IP) packets across the international Internet is typically best effort, and subject to network variability outside the control of any video server or client.

A number of the network congestion and client load issues that arise on the Internet also pertain to LANs, MANs, and WANs. Therefore, the technique of the invention could well be applicable to these other network types. However, the focus of the inventors' work, particularly so far as the preferred embodiment is concerned, has been in an Internet application.

In terms of supporting real time video on the Web, inter-frame jitter greatly affects video playback quality across the network. (For purposes of the present discussion, jitter is taken to be the variance in inter-arrival time between subsequent frames of a video stream.) A high degree of jitter typically causes the video playback to appear "jerky". In addition, network congestion may cause frame delays or losses. Transient load at the client side may prevent the client from handling the full frame rate of the video.

In order to accomplish support for real time video on busy networks, and in particular on the Web, the inventors created a specialized real time transfer protocol for handling video across the Internet. The inventors have determined that this protocol successfully handles real time Internet video by minimizing jitter and incorporating dynamic adaptation to the client CPU load and network congestion.

In accordance with another aspect of the invention, continuous media organization, storage and retrieval are provided. In the present invention, continuous media consist of video and audio information. There are several classes of so-called meta-information which describe various aspects of the continuous media itself. This meta-information includes the inherent properties of the media, hierarchical information, semantic description, as well as annotations that provide support for hierarchical access, browsing, searching, and dynamic composition of the continuous media.

To accomplish these and other objects, the invention provides a method and a system for real time transmission of data on a network which links a plurality of computers. The method and system involve at least two, and typically a larger number of networked computers, wherein, during real time transmission of data, parameters affecting the potential rate of data transmission in the system (*e.g.* network and/or performance) are monitored periodically, and the information derived from the feedback used to moderate the rate of real-time data transmission on the network.

According to one embodiment, first and second computers are provided, the second computer having a user output device connected to it. To establish real-time transmission, the first and second computers first establish communication with each other. The computers determine transmission performance between them, and also communicate processing performance (*e.g.* processor load) of the second computer. The first computer transmits data to the second computer for output on the user output device in real time. The rate of transmitting data is adjusted as a function of network performance and/or processor performance.

In accordance with a further preferred embodiment, the first computer has a resident program which provides for real time transmission of data, and which determines network performance. The second computer has a resident program which enables receipt of data and routing of that data to the user output device in real time. The second computer's program may condition the data further, and also may communicate processor performance information to the first computer. The program in the first computer may degrade or upgrade real time data transmission rates to the second computer based on the network and/or processor performance information received.

In accordance with a still further preferred embodiment, the first and second computers communicate with each other over two channels, one channel passing control information between the two computers, and the other channel passing data for real time output, and also feedback information, such as network and/or processor performance information. The integrity of the second channel need not be as robust as that of the first channel, in view of the dynamic allocation ability of the real time transmission.

Communication between the first and second computers may involve static data, such as for document transmission, as well as continuous media, such as for video and audio transmission. Preferably, the inventive method and system are applied to handling of continuous media.

In normal, larger applications, the first computer, or server, will have a number of computers, or clients, with which the server will communicate, using the dual-channel, feedback technique of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects and features of the invention will become apparent from the following detailed description with reference to the accompanying drawings, in which:

Fig. 1 shows a four-item video menu as part of the invention;

Fig. 2 is a diagram of the internal structure of the invention;

Fig. 3 shows a video control panel in accordance with the invention;

Fig. 4 shows structure of a server configured in accordance with the invention;

Fig. 5 depicts the connection between a server and a client in accordance with the invention;

Fig. 6 depicts retransmission and size of a buffer queue;

Fig. 7 depicts a transmission queue;

Fig. 8 is a flow graph for moderating transmission flow;

Figures 9-13 are flow charts depicting operation of the invention, and in particular, operation of a server and its associated clients;

Fig. 14 shows the hardware environment of one embodiment of the present invention;

Figs. 15a-15g show interface screens which demonstrate the invention;

Fig. 16 is a graph of a frame rate adaptation in accordance with the invention;

Fig. 17 depicts structure of continuous media;

Fig. 18 depicts hierarchical organization and indexing of an example of continuous media;

Fig. 19 contains a list of keyword descriptions for providing links to continuous media;

Fig. 20 shows a display screen of the invention side by side with the hierarchical architecture of the continuous media to be displayed;

Fig. 21 is a screen displaying the results of a key word search;

Fig. 22 is a screen displaying an example of hyperlinks embedded in video data;

Fig. 23 depicts dynamic composition of video streams; and

Fig. 24 depicts interpolation of hyperlinks in video streams.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

As was mentioned earlier, Vosaic is based on NCSA Mosaic. Mosaic concentrates on HTML documents. While all media types are treated as documents, each media type is handled differently. Text and inlined images are displayed in place. Other media types, such as video and audio files, or special file formats (*e.g.*, Postscript(tm)) are handled externally by invoking other programs. In Mosaic, documents are not displayed until fully available. The Mosaic client keeps the retrieved document in temporary storage until all of the document has been fetched. The sequential relationship between transferring and processing of documents makes the browsing of large video/audio documents and real time video/audio sources problematic. Transferring such documents require long delay times and large client side storage space. This makes real time playback impossible.

Real time video and audio convey more information if directly incorporated into the display of a hypertext document. For example, the inventors have implemented real time video menus and video icons as an extension of HTML in Vosaic. Figure 1 depicts a typical four-item video menu which can be constructed using Vosaic. Video menus present the user with several choices. Each choice is in the form of a moving video. One may, for example, click on a video menu item to follow the link, and watch the clip in full size. Video icons show a video in an small, unobtrusive icon-sized rectangle within the HTML document. Embedded real time video within WWW documents greatly enhances the look and feel of a Vosaic page. Video menu items convey more information about the choices available than simple textual descriptions or static images.

Looking more closely at the internal structure of Vosaic, HTML documents with video and audio integrated therein are characterized by a variety of data transmission protocols, data decoding formats, and device control mechanisms (*e.g.*, graphical display, audio device control, and video board control). Vosaic has a layered structure to meet these requirements. The layers, which are depicted in Figure 2, are document transmission layer 200, document decoding layer 230, and document display layer 260.

A document data stream flows through these three layers by using different components from different layers. The composition of components along the data path of a retrieved document occurs at run-time according to document meta-information returned by an extended HTTP server.

As discussed earlier, TCP is only suitable for static document transfers, such as text and image transfers. Real time playback of video and audio requires other protocols. The current implementation in the Vosaic document transmission layer 200 includes TCP, VDP and RTP. Vosaic is configured to have TCP support for text and image transmission. Real time playback of real time video and audio uses VDP. RTP is the protocol used by most Mbone conferencing transmissions. A fourth possible protocol is for interactive communication (used for virtual reality, video games and interactive distance learning) between the web client and server.

The decoding formats currently implemented in document decoding layer 230 include:

For images: GIF and JPEG

For video: MPEG1, NV, CUSEEME, and Sun CELLB

For audio: AIFF and MPEG1

MPEG1 includes support for audio embedded in the video stream. The display layer 260 includes traditional HTML formatting and inline image display. The display has been extended to incorporate real time video display and audio device control.

Standard URL specifications include FTP, HTTP, Wide Area Information System (WAIS), and others, covering most of the currently existing document retrieval protocols. However, access protocols for video and audio conferences on the Mbone are neither defined nor supported. In accordance with the invention, the standard URL specification and HTML have been extended to accommodate real time continuous media transmission. The extended URL specification supports Mbone transmission protocols using the mbone keyword as a URL scheme, and on-demand continuous media protocols using cm (for "continuous media") as the URL scheme. The format of the URL specifications for the Mbone and continuous real time are as follows:

mbone://address:port:ttl:format

cm://address:port:format/filepath

Examples are given below:

mbone://224.2.252.51:4739:127:nv

cm://showtime.ncsa.uiuc.edu:8080:mpegvideo/puffer.mpg

cm://showtime.ncsa.uiuc.edu:8080:mpegaudio/puffer.mp2

The first URL encodes an Mbone transmission on the address 224.2.252.51, on port 4739, with a time to live (TTL) factor of 127, using nv (for "network video") video transmission format. The second and third URLs encode continuous media transmissions of MPEG video and audio respectively.

Incorporating inline video and audio in HTML necessitates the addition of two more constructs to the HTML syntax. The additions follow the syntax of inline images closely. Inlined video and audio segments are specified as follows:

<video src="address:port/filepath option=cyclic|control">

<audio src="address:port/filepath option=cyclic|control">

The syntax for both video and audio is made up of a src part and an options part. Src specifies the server information including the address and port number. Options specifies how the media is to be displayed. Two options are possible: control or cyclic. The control display option pops up a window with a control panel and the first frame of the video is displayed, with further playback controlled by the user. Figure 3 shows a page with a video control panel, as will be described.

The cyclic display option displays the video or audio clip in a loop. The video stream may be cached in local storage to avoid further network traffic after the first round of display. This is feasible when the size of video or audio clip is small. If the segment is too large to be stored locally at the client end, the client may also request the source to send the clip repeatedly. Cyclic video clips are useful for constructing video menus and video icons.

If the control keyword is given, a control panel is presented to the user. A control interface, also shown in Figure 3, allows users to browse and control video clips. The following user control buttons are provided:

Rewind: Play the video backwards at a fast speed.

Play: Start to play the video.

Fast Forward: Play the video at a faster speed. In accordance with the preferred embodiment, this is implemented by dropping frames at the server site. Determination of circumstances surrounding frame dropping, and implementation of frame dropping techniques, are discussed in greater detail below.

Stop: Ends the playing of the video.

Quit: Terminates playback. When the user presses "Play" again, the video is restarted from the beginning.

Real time video and audio use VDP as a transfer protocol over one channel between the client and the server. Control information exchange uses a TCP connection between the client and server. Thus, there are two channels of communication between the client and the server, as will be described.

Vosaic works in conjunction with a server 400, a preferred configuration of which is shown in Fig. 4. The server 400 uses the same set of transmission protocols as does Vosaic, and is extended to handle video transmission. Video and audio are transmitted with VDP. Frames are transmitted at the originally recorded frame rate of the video. The server uses a feed forward and feedback scheme to detect network congestion and automatically delete frames from the stream in response to congestion.

In previously preferred embodiments, the server 400 handled HTTP as well as continuous media. However, HTTP applications can be handled outside of Vosaic, so inclusion of HTTP, and of an HTTP handler no longer is essential to the implementation. Also, among continuous media formats, the inventors had experimented with MPEG, but since have confirmed that Vosaic works well with numerous video and audio standards, including (but by no means limited to) H.263, GSM, and G.723.

The main components of the server 400, shown in Figure 4, are a main request dispatcher 410, an admission controller 420, continuous media (cm) handler 440, audio and video handlers 450, 460, and a server logger 470.

In operation, the main request dispatcher 410 receives requests from clients, and passes them to the admission controller 420. The admission controller 420 then determines or estimates the requirements of the current request: these requirements may include network bandwidth and CPU load. Based on knowledge of current conditions, the controller 420 then makes a decision on whether the current request should be serviced.

Traditional HTTP servers can manage without admission control because document sizes are small, and request streams are bursty. Requests simply are queued before service, and most documents can be handled quickly. In contrast, with continuous media transmissions in a video server, file sizes are large, and real time data streams have stringent time constraints. The server must ensure that it has enough network bandwidth and processing power to maintain service qualities for current requests. The criteria used to evaluate requests may be based on the requested bandwidth, server available bandwidth, and system CPU load.

In accordance with a preferred embodiment of the invention, the system limits the number of concurrent streams to a fixed number. However, the admission control policy is flexible; a more sophisticated policy is within the inventors' contemplation, and in this context would be within the abilities of the ordinarily skilled artisan.

Once the system grants the current request, the main request dispatcher 410 hands the request to cm handler 440, which then hands the appropriate part of the request to the corresponding audio or video handler 450, 460. While the video and audio handlers use VDP, as described below, in accordance with the invention, the server design is flexible enough to incorporate more protocols.

The server logger 470 is responsible for recording the request and transmission statistics. Based on studies of access patterns of the current Web servers, it is expected that the access patterns for a video enhanced Web server will be substantially different from those of traditional WWW servers that support mainly text and static images.

The server logger 470 records the statistics for the transmission of continuous media in order to better understand the behavior of requests for continuous media. The statistics include the network usage and processor usage of each request, the quality of service data such as frame rate, frame drop rate, and jitter. The data will guide the design of future busy Internet video servers. These statistics are also important for analyzing the impact of continuous media on operating systems and the network.

Video Datagram Protocol (VDP)

Looking now at the protocol for transmitting video in real time, the inventive video datagram protocol, or VDP, is an augmented real time datagram protocol developed to handle video and audio over the Web. VDP design is based on making efficient use of the available network bandwidth and CPU capacity for video processing. VDP differs from RTP in that VDP takes advantage of the point-to-point connection between Web server and Web client. The server end of VDP receives feedback from the client and adapts to the network condition between client and server and the client CPU load. VDP uses an adaptation algorithm to find the optimal transfer bandwidth. A demand resend algorithm handles frame losses. VDP differs from Cyclic-UDP in that it resends frames upon request instead of sending frames repeatedly, hence preserving network bandwidth, and avoiding making network congestion worse.

In accordance with the invention, the video also contains embedded links to other objects on the Web. Users can click on objects in the video stream without halting the video. The inventive Vosaic Web browser will follow the embedded hyperlink in the video. This promotes video to first class status within the World Wide Web. Hypervideo streams can now organize information in the World Wide Web in the same way hypertext improves plain text.

VDP is a point-to-point protocol between a server program which is the source of the video and audio data, and a client program which allows the playback of the received video or audio data. VDP is designed to transmit video in Internet environments. There are three problems the algorithm must overcome:

- bandwidth variance in the network,
- packet loss in the network, and
- the variable bit rate (VBR) nature of some compressed video formats.

The amount of available bandwidth may be less than that required by the complete video stream, due to fluctuating bandwidth in the network, or due to high bandwidth stretches of VBR video. Packet loss may also adversely affect playback quality.

VDP is an asymmetric protocol. As shown in Figure 5, between the client 500 and the server 550, there are two network channels 520, 540. The first channel 520 is a reliable TCP connection stream, upon which video parameters and playback commands (such as Play, Stop, Rewind and Fast Forward) are sent between client and server. These commands are sent on the reliable TCP channel 520 because it is imperative that playback commands are transmitted reliably. The TCP protocol provides that reliable connection between client and server.

The second network channel 540 is an unreliable user datagram protocol (UDP) connection stream, upon which video and audio data, as well as feedback messages are sent. This connection stream forms a *feedback loop*, in which the client receives video and audio data from the server, and feeds back information to the server that the server will use to moderate its rate of transmission of data. Video and audio data is transmitted on this unreliable channel because video and audio can tolerate losses. It is not essential that all data for such continuous media be transmitted reliably, because packet loss in a video or audio stream causes only momentary frame or sound loss.

Note that while, in accordance with a preferred embodiment, VDP is layered directly on top of UDP, VDP can also be encapsulated within Internet standards such as RTP, with RTCP as the feedback channel.

VDP Transmission Mechanism

After the admission controller 420 (Figure 4) in server 550 (Figure 5) grants the request from the client 500, the server 550 waits for the play command from the client. Upon receiving the play command, the server starts to send the video frames on the data channel using the recorded frame rate. The server end breaks large frames into smaller packets (for example, 8 kilobyte packets), and the client end reassembles the packets into frames. Each frame is time-stamped by the server and buffered at the client side. The client controls the sending of frames by sending server control commands, like stop or fast forward, on the control channel.

VDP Adaptation Algorithm

The VDP adaptation algorithm dynamically adapts the video transmission rate to network conditions along the network span from the client to the server, as well as to the client end's processing capacity. The algorithm degrades or upgrades the server transmission rate depending on feed forward and feedback messages exchanged on the control channel. This design is based on the consideration of saving network bandwidth.

Protocols for the transmission of continuous media over the Internet, or over other networks for that matter, need to preserve network bandwidth as much as possible. If a client does not have enough processor capacity, it may not be fast enough to decode video and audio data. Network connections may also impose constraints on the frame rate at which video data can be sent. In such cases, the server must gracefully degrade the quality of service. The server learns of the status of the connection from client feedback.

Feedback messages are of two types. A first type, the frame drop rate, corresponds to frames received by the client but which have been dropped because the client did not have enough CPU power to keep up with decoding the frames. The second type, the packet drop rate, corresponds to frames lost in the network because of network congestion.

If the client side protocol discovers that the client application is not reading received frames quickly enough, it updates the frame loss rate. If the loss rate is severe, the client sends the information to the server. The server then adjusts its transmission speed accordingly. In accordance with a preferred embodiment, the server slows down its transmission if the loss rate exceeds 15%, and speeds up if the loss rate is below 5%. However, it should be understood that the 15% and 5% figures are engineering thresholds, which can vary for any number of reasons, depending on conditions, outcomes of experiments, and the like.

In response to a video request, the server begins by sending out frames using the recorded frame rate. The server inserts a special packet in the data stream indicating the number of packets sent out so far. On receiving the feed forward message from the server, the client may then calculate the packet drop rate. The client returns the feedback message to the server on the control channel. In accordance with a preferred embodiment, feedback occurs every 30 frames. Adaptation occurs very quickly -- on the order of a few seconds.

Demand Resend Algorithm

The compression algorithms in some media formats use inter-frame dependent encoding. For example, a sequence of MPEG video frames has I, P, and B frames. I frames are frames that are intra-frame coded with JPEG compression. P frames are frames that are predictively coded with respect to a past picture. B frames are frames that are bidirectionally predictive coded.

MPEG frames are arranged into groups with sequences that correspond to the pattern I B B P B B P B B. The I frame is needed by all P and B frames in order to be decoded. The P frames are needed by all B frames. This encoding method makes some frames more important than the others. The display quality is strongly dependent on the receipt of important frames. Since data transmission can be unreliable over the Internet, there is a possibility of frame loss. If, in a sequence group of MPEG video frames I B B P B B P B B recorded at 9 frames/sec, the I frame is lost, the entire sequence becomes undecodable. This undecodability produces a one second gap in the video stream.

Some protocols, such as Cyclic-UDP, use a priority scheme in which the server sends the important frames repeatedly within the allowable time interval, so that the important frames have a better chance of getting through. VDP's demand resend is similar to Cyclic-UDP in that, in VDP, the responsibility of determining which frames are resent is put on the client based on its knowledge of the encoding format used by the video stream. However, unlike Cyclic-UDP, VDP does not rely on the server's repeated retransmission of frames, because such repeated retransmission would be more likely to cause unacceptable jitter. Accordingly, in an MPEG stream, the VDP algorithm may choose to request retransmissions of only the I frames, or of both the I and P frames, or all frames. VDP employs a buffer queue at least as large as the number of frames required during one round trip time between the client and the server. The buffer is full before the protocol begins handing frames to the client from the queue head. New frames enter at the queue tail. A demand resend algorithm is used to generate resend requests to the server in the event a frame is missing from the queue tail. Since the buffer queue is large enough, it is highly likely that re-sent frames can be correctly inserted into the queue before the application requires it.

The following is the client/server setup negotiation, in which a client computer contacts the video server to request a video or audio file. Referring to Figure 5, which is a schematic depiction of a client-server channel setup, the sequence is as follows:

- The client 500 first contacts the server 550 by initiating a reliable TCP network connection to the server over channel 520.
- If the connection is successfully set up, the client 500 then chooses a UDP port (say *u*), and establishes communication over channel 540. The client 500 then sends to the server 550, over the port *u*, the name of the video or audio file requested.

- If the server 550 finds the requested file, and the server 550 can accept the video or audio connection, then the client 500 prepares to receive data on UDP port *u*.
- When the client 500 wishes to receive data from the server 550, the client sends a Play command to the server 550 on the reliable TCP charnel 520. The server 550 will then start streaming data to the client 500 at port *u*.

The particular setup sequence just described, which the currently preferred implementation of VDP uses, illustrates how the two connections, reliable and unreliable, are set up. However, the particular sequence is not essential to the proper functioning of the adaptive algorithm.

The VDP server 550 is in charge of transmitting requested video and audio data to the client 500. The server receives playback commands from the client through the reliable TCP channel, and sends data on an unreliable UDP channel to the client. It also receives feedback messages from the client, informing it of the conditions detected at the client. It uses these feedback messages to moderate the amount of data transmitted in order to smooth out transmission under congested conditions.

The server streams data at the proper rate for the type of data requested. For example, a video that is recorded at 24 frames per second will have its data packetized and transmitted such that 24 frames worth of data is transmitted every second. An audio segment that is recorded at 12 Kbit/s will be packetized and transmitted at that same rate.

For its part, the client sends playback commands, including Fast Forward, Rewind, Stop and Play, to the server on the reliable TCP channel. It also receives video and audio data from the server on the unreliable UDP channel.

As packets arriving from the network are subject to some degree of jitter, a *playout buffer* is used to smooth jitter between continuous media frames. The playout buffer is of some length l, measured in frame time. For reasons described later, $l = p \times$ RTT, where RTT is the Round Trip Time between the client and the server, and p is some factor ≤ 1.

Figure 6 depicts retransmission and size of the buffer queue. On the client side 610, a playout buffer 620 is also used to allow retransmission of important frames which are lost. VDP uses a *retransmit once* scheme, *i.e.* retransmit requests for a lost frame are only sent once. The protocol does not require that data behind the lost packet be held up for delivery until the lost packet is correctly delivered. Packets are time stamped and have sequence numbers. Lost frames are detected at the tail of the queue. A retransmission request 650 is sent to the server side 660 if a decision is made on the client side 610 that a frame has been lost (a packet with a sequence number more than what was expected arrives). p must be greater than or equal to 1 in order that the lost frame have enough *time* to arrive before its slot arrives at the head of the queue. The exact value of p is an engineering decision.

The protocol must also guard against retransmission causing a *cascade* effect. Since a retransmitted frame increases the bandwidth of data when it is transmitted again, it may cause further loss of data. Retransmit requests issued for these subsequent lost packets can trigger more loss again. VDP avoids the cascade effect by limiting retransmits. As a retransmission takes one round trip time from sending the retransmission request to having the previously lost data arrive, the limit is one retransmission request for any frame within a *retransmit window* 630, equal to $w \times$ RTT for $w > 1$.

The VDP adaptive algorithm detects two types of congestion. The first type, network congestion, results from insufficient bandwidth in the network connection to sustain the frame rate required for video and audio. The second type, CPU congestion, results from insufficient processor bandwidth required for decoding the compressed video and audio.

To identify and address both types of congestion, feedback is returned to the server in order for the server to moderate its transmission rate. Moderation is accomplished by *thinning* the video stream, either by not sending as many frames, or by reducing image quality by not sending high resolution components of the picture. Audio data is never thinned. The loss of audio data results in glitches in the playback, and are more perceptually disturbing to the user than is degradation of video quality. Thinning techniques for video data are well known, and so need not be described in detail here.

When the network is congested, there is insufficient bandwidth to accommodate all the traffic. As a result, data that would normally arrive fairly quickly is delayed in the network, as network queues build up in intermediate routers between client and server. Since the server transmits data at regular intervals, the interval between subsequent data packets increases in the presence of network congestion.

The protocol thus detects congestion by measuring the inter-arrival times between subsequent packets. Inter-arrival times exceeding the expected value signal the onset of network congestion ; such information is fed back to the server. The server then thins the video stream to reduce the amount of data injected into the network.

Because of packet jitter within the network, inter-arrival times between subsequent packets may vary in the absence of network congestion. A *low-pass filter* is used to remove the transient effects of packet jitter. Given the difference in arrival time between packets i and packets $i + 1$ of δt, the inter-arrival time t_{i+1} at time $i + 1$ is:

$$t_{i+1} = (1 - \alpha) \times t_i + \alpha \times \delta t, \; 0 \leq \alpha \leq 1 \qquad (1)$$

The filter provides a cumulative history of the inter-arrival time while removing transient differences in packet inter-arrival times.

Packet loss is also indicative of network congestion. As the amount of queuing space in network routers is finite, excessive traffic may be dropped if there is not enough queue space. In VDP, packet loss exceeding and engineering threshold is also indicative of network congestion.

CPU congestion occurs when there is too much data for the client CPU to decode. As VDP transports compressed video and audio data, the client processor is required to decode the compressed data. Some clients may possess insufficient processor bandwidth to keep up. In addition, in modern time sharing environments, the client's processor is shared between several tasks. A user starting up a new task may reduce the amount of processor bandwidth available to decode video and audio. Without adaptation to CPU congestion, the client will fall behind in decoding the continuous media data, resulting in slow motion playback. As this is undesirable, VDP also detects CPU congestion on the client side.

CPU congestion is detected by directly measuring if the client CPU is keeping up with decoding the incoming data.

Figure 7 depicts buildup of a queue of continuous media information in the presence of network congestion. Figure 8 depicts a flow graph for handling feedback and transmission/reception adaptation under varying loads and levels of congestion.

Figures 9-13 are flow charts depicting the sequence of VDP operations at the respective client and the server sides. In Figure 9, depicting a top level operational flow at the client side, the connection setup sequence is initiated. If the setup is successful, video/audio transmission and playback is initiated. If the setup is not successful, operation ends.

In Figure 10, depicting the flow of setup of a client connection, first a TCP connection is set up, and then a request is sent to the server. If the request is granted, the connection is considered successful, and playback is initiated. If the request is not granted, the server sends an error message, and the TCP connection is terminated.

In Figure 11, once the TCP connection is set up successfully, and communication established successfully with the server, a UDP connection is set up. Round trip time (RTT) is estimated, and then buffer size is calculated, and the buffer is set up. The client then receives packets from the UDP connection, and decodes and displays video and audio data. The presence or absence of CPU congestion is detected, and then the presence or absence of network congestion is detected. If congestion at either point is detected, the client sends a message to the server, telling the server to modify its transmission rate. If there is no congestion, the user command is processed, and the client continues to receive packets from the UDP connection. As can be seen from the Figure, a feedback loop is set up in which transmission from the server to the client is modified based on presence of congestion. Thus, rather than the client simply telling the server to continue sending, the client actually tells the server, under circumstances of congestion, to modify its sending rate.

Figure 12 shows the server's side of the handling of client requests. The server accepts requests from a client, and evaluates the client's admission control request. If the request can be granted, the server sends a grant, and initiates a separate process to handle the client's request. If the request cannot be granted, the server sends a denial to the client, and goes back to looking for further client requests.

Figure 13 depicts the server's internal handling of a client request. First, a UDP connection is set up. Then, RTT is estimated. Video/audio parse information then is read in, and an initial transfer rate is set. If the server receives a message from the client, asking for a modification of the transfer rate, the server adjusts the rate, and then sends out packets accordingly. If there is no request for transfer rate modification, then the server continues to send out packets at the previous (most recent) transfer rate. If the client has sent a playback command, then the server looks for an adaptation message, and continues to send packets. If the client has sent a "quit" command, the TCP and UDP connections are terminated.

Figure 14 shows, in broad outline, the hardware environment in which the present invention operates. A plurality of servers and clients are connected over a network. In the preferred embodiment, the network is the Internet, but it is within the contemplation of the invention to replace other network protocols, whether in LANs, MANs, or WANs, with the inventive protocol, since the use of TCP/IP is not limited to the Internet, but indeed pertains over other types of networks.

Figures 15a-15g, similarly to Figures 1 and 3, show further examples of types of display screens which a user would encounter in the course of using Vosaic. Figures

15a-15d depict various frames of a dynamic presentation. Figure 15a shows an introductory text screen. Figure 15b shows two videos displayed on the same screen, using the present invention. Figure 15c shows a total of four videos displayed on the same screen. Figure 15d illustrates the appearance of the screen at the end of the videos presented in Figure 15c.

Figure 15e shows the source which invokes the presentation depicted in Figures 15a-15d. Figure 15f illustrates an interface screen with hyperlinks in video objects, in the boxed area within the video. Also, similarly to Figure 3, a control panel is shown with controls similar to those of a videocassette recorder (VCR), to control playback of videos. Clicking on the hyperlinked region in Figure 15f results in the page shown in Figure 15g, which is the video to be played.

The inventors carried out several experiments over the Internet. The test data set consisted of four MPEG movies, digitized at rates ranging from 5 to 9 fps, with pixel resolution ranging from 160 by 120 to 320 by 240. Table 1 below identifies the test videos that were used.

Name	Frame Rate (fps)	Resolution	Number of Frames	Play Time (secs)
model.mpg	9	160 by 120	127	14
startrek.mpg	5	208 by 156	642	128
puffer.mpg	5	320 by 240	175	35
smalllogo.mpg	5	320 by 240	1622	324

Table 1: MPEG test movies.

The videos listed in Table 1 ranged from a short 14 second segment to one of several minutes duration.

In order to observe the playback video quality, the inventors based the client side of the tests in the laboratory. In order to cover the widest possible range of configurations, servers were set up corresponding to local, regional and international sites relative to the geographical location of the laboratory. A server was used at the

National Center for Supercomputing Applications (NCSA) for the local case. NCSA is connected to the local campus network at the University of Illinois/Champaign-Urbana via Ethernet. For the regional case, a server was used at the University of Washington. Finally, a copy of the server was set up at the University of Oslo in Norway to cover the international case. Table 2 below lists the names and IP addresses of the hosts used for the experiments.

Name	IP Address	Function
indy1.cs.uiuc.edu	128.174.240.90	local client
showtime.ncsa.uiuc.edu	141.142.3.37	local server
agni.wtc.washington.edu	128.95.78.229	regional server
gloin.ifi.uio.no	129.240.106.18	international server

Table 2: Hosts used in our tests.

Name	% Dropped Frames	Jitter (ms)
model	0	8.5
startrek	0	5.9
puffer	7.5	43.6
smalllogo	0.5	22.5

Table 3: Local test.

Name	% Dropped Frames	Jitter (ms)
model	0	46.3
startrek	0	57.1
puffer	0	34.3
smalllogo	0.2	50.0

Table 4: Regional test.

Name	% Dropped Frames	Jitter (ms)
model	0	20.1
startrek	0	22.0
puffer	19	121.4
smalllogo	0.8	46.7

Table 5: International test.

Tables 3-5 show the results for sample runs using the test videos by the Web client accessing the local, regional and international servers respectively. Each test involved the Web client retrieving a single MPEG video clip. An unloaded Silicon Graphics (SGI) Indy was used as the client workstation. The numbers give the average frame drop percentage and average application-level inter-frame jitter in milliseconds for thirty test runs. Frame rate changes because of to the adaptive algorithm were seen in only one run. That run used the puffer.mpg test video in the international configuration (Oslo, Norway to Urbana, USA). The frame rate dropped from 5 fps to 4 fps at frame number 100, then increased from 4 fps to 5 fps at frame number 126. The rate change indicated that transient network congestion caused the video to degrade for a 5.2 second period during the transmission.

The results indicate that the Internet supports a video-enhanced Web service. Inter-frame jitter in the local configuration is negligible, and below the threshold of human observability (usually 100 ms) in the regional case. Except for the puffer.mpg runs, the same holds true for the international configuration. In the puffer.mpg case, the adaptive algorithm was invoked because of dropped frames and the video quality was degraded for a 5.2 second interval. The VDP buffer queue efficiently minimizes frame jitter at the application level.

The last test exercised the adaptive algorithm more strongly. Using the local configuration, a version of smalllogo.mpg recorded at 30 fps at a pixel resolution of 320 by 240 was retrieved. This is a medium size, high quality video clip, requiring significant computing resources for playback. Figure 16 shows a graph of frame rate versus frame sequence number for the server transmitting the video.

The client side buffer queue was set at 200 frames, corresponding to about 6.67 seconds of video. The buffer at the client side first filled up, and the first frame was handed to the application at frame number 200. The client workstation did not have enough processing capacity to decode the video stream at the full 30 fps rate. The client side protocol detected a frame loss rate severe enough to report to the server at frame number 230. In accordance with a presently preferred embodiment, transmission is degraded when the frame loss rate exceeds 15%. Transmission is upgraded if the loss rate is below 5%.

The server began degrading its transmission at frame number 268, that is, within 1.3 seconds of the client's detection that its CPU was unable to keep up. The optimal transmission level was reached in 7.8 seconds, corresponding to a 9 frame per second transmission rate. Stability was reached in a further 14.8 seconds. The deviation from optimal did not exceed 3 frames per second in either direction during that period. The results show a fundamental tension between large buffer queue sizes that minimize jitter and server response times.

The test with very high quality video at 30 fps with a frame size of 320 by 240 represents a pathological case. However, the results show that the adaptive algorithm is an attractive way to reach optimal frame transmission rates for video in the WWW. The test implementation changes the video quality by 1 frame per second at each iteration. It is within the contemplation of the invention to employ non-linear schemes based on more sophisticated policies.

In accordance with another aspect of the invention, continuous media organization, storage and retrieval is provided. Continuous media consist of video and audio information, as well as so-called meta-information which describes the contents of the video and audio information. Several classes of meta-information are identified in order to support flexible access and efficient reuse of continuous media. The meta-information encompasses the inherent properties of the media, hierarchical information, semantic description, as well as annotations that provide support for hierarchical access, browsing, searching, and dynamic composition of continuous media.

As shown in Figure 17, the continuous media integrates video and audio documents with their meta-information. That is, the meta-information is stored together with the encoded video and audio. Several classes of meta-information include:

- Inherent properties: The encoding scheme specification, encoding parameters, frame access points and other media-specific information. For example, for a video clip encoded in the MPEG format, the encoding scheme is MPEG, and the encoding parameters include the frame rate, bit rate, encoding pattern, and picture size. The access points are the file offsets of important frames.
- Hierarchical structure: Hierarchical structure of video and audio. For example, a movie often consists of a sequence of clips. Each clip is made of a sequence of shots (scenes), while each shot includes a group of frames.
- Semantic descriptions: Descriptions of the parts, or of the whole video/audio document. Semantic descriptions facilitate search. Searching through large video and audio clips is hard without semantic description support.
- Semantic Annotations: Hyperlink specifications for objects inside the media streams. For example, for an interesting object in a movie, a hyperlink can be provided which leads to related information. Annotation information allows the browsing of continuous media and can integrate video and audio with static data types like text and images.

Inherent properties assist in the network transmission of continuous media. They also provide random access points into the document. For example.

substantial detail has been provided above, describing the inventive adaptive scheme for transmitting video and audio over packet-switched networks with no quality of service guarantees. The scheme adapts to the network and processor load by adjusting the transmission rate. The scheme relies on the knowledge of the encoding parameters, such as the bit rate, frame rate and encoding pattern.

Information about frame access points enables frame-based addressing. Frame addressing allows accesses to video and audio by frame number. For example, a user can request a portion of a video document from frame number 1000 to frame number 2000. Frame addressing make frames the basic access unit. Higher level meta-information, such as structural information and semantic descriptions, can be built by associating a description with a range of frames.

The encoding within the media stream often includes several of the inherent properties of meta-information. These parameters are extracted and stored separately, as on-the-fly extraction is expensive. On-the-fly extraction unnecessarily burdens the server and limits the number of requests that the server can serve concurrently.

A video or audio document often possesses a hierarchical structure. An example of hierarchical information in a movie is shown in Figure 18. The movie example in that Figure, "Engineering College and CS Department at UIUC" consists of the clips "Engineering College Overview" and " CS Department Overview". Each of these clips is composed of a sequence of shots; in the case of "Engineering College Overview," the sequence consists of "Campus Overview", "Message from Dean," and others. The hierarchical structure describes the organizational structure of continuous media, making hierarchical access and non-linear views of continuous media possible.

Semantic descriptions describe part or the whole video/audio document. A range of frames can be associated with a description. As shown in Figure 19, the shots in the example movies are associated (indexed) with keywords. Semantic annotations describe how a certain object within a continuous media stream is related to some other object. Hyperlinks can be embedded to indicate this relationship.

Continuous media allows multiple annotations and semantic descriptions. Different users can describe and annotate in different ways. This is essential in supporting multiple views on the same physical media. For example, a user may describe the campus overview shot in the example movie as "UIUC campus", while another user may associate it with "Georgian style architecture in the United States Midwest". That user may have a link from his/her presentation to introduce the UIUC campus, while another user may use relative frames of the same video segment to describe Georgian-style architecture.

Supporting multiple views considerably simplifies content preparation. This is because only one copy of the physical media is needed. Users can use part or the whole copy for different purposes.

The meta-information described above is essential in supporting flexible access and efficient reuse. The hierarchical information can be displayed along with the video to provide the user a view of the overall structure of the video. It allows the user to access to any desired clip, and any desired shot. Figure 20 shows an implementation of the video player in Vosaic; specifically, a movie is shown along with its hierarchical structure. Each node is associated with a description. A user can click on nodes of the structure and that portion of the movie will be shown in the movie window.

Hierarchical access enables a non-linear view of video and audio, and facilitates greatly the browsing of video and audio materials. Video and audio documents traditionally have been organized linearly. Even though traditional access methods, such as the VCR type of operations, or the slide bar operation, allow arbitrary positioning inside video and audio streams, finding the interesting parts within a video presentation is difficult without strong contextual knowledge, since video and audio express meanings through the temporal dimension. In other words, a user cannot easily understand the meaning of one frame without seeing related frames and shots. Displaying hierarchical structure and descriptions provides users with a global picture of what the movie and each part is about.

Searching capability can be supported by searching through the semantic description. For example, the keyword descriptions in Figure 19 can be queried. The search of keyword tour will return all the tours in the movie, *e.g.*, One Lab Tour, DCL Tour, and Instructional Lab Tour. One implementation of a search is shown in Figure 21, in which the matched entries for the query are listed.

Browsing is supported through hyperlinks embedded within video streams and through hierarchical access. Hyperlinks within video streams are an extension of the general hyperlink principle, in this case, making objects within video streams anchors for other documents. As shown in Figure 22, a rectangle outlining a black hole object indicates that it is a anchor, and upon clicking the outline, the document to which it is linked is fetched and displayed (in this case, an HTML document about black holes). Hyperlinks within video streams integrate and facilitate inter-operation between video streams and traditional static text and images.

Continuous media also allows dynamic composition. A video presentation can use parts of existing movies as components. For example, a presentation of Urbana-Champaign can be a video composed of several segments from other movies. As shown in Figure 23, the campus overview segment can be used in the composition. The specification of this composition is done through hyperlinks.

Vosaic's architecture is based on continuous media, as outlined above. Meta-information is stored on the server side together with the media clips. Inherent properties are used by the server in order to adapt the network transmission of continuous media to network conditions and client processor load. Semantic description and annotations are used for searching video material and hyperlinking inside video streams. In the design and implementation of tools for the extraction and construction of continuous media meta-information, a parser was developed to extract inherent properties from encoded MPEG video and audio streams. A link-editor was implemented for the specification of hyperlinks within video streams. There also are tools for video segmentation and semantic description editing.

Frame addressing uses the video frame and the audio sample as basic data access units to video and audio, respectively. During the initial connection phase between Vosaic server and client, the start and end frames for specific video and audio segments are specified. The default settings are the start and the end frame of the whole clip. The server transmits only the specified segment of video and audio to the client. For example, for a movie that is digitized as a whole and is stored on the server, the system allows a user to request frame number 2567 to frame number 4333. The server identifies and retrieves this segment, and transmits the appropriate frames to the client.

A parser has been developed for extracting inherent properties from MPEG video and audio streams. The parsing is done off-line. The parse file contains:

1. picture size, the frame rate, pattern,
2. average frame size, and
3. offset for each frame

in the clip file.

A example parse file is shown below:

```
#
#
# ---------------------------------------------------------------------------
# cs.mpg.par
#
# Parse file for MPEG stream file
# This file is generated by mparse, a parse tool for MPEG stream file.
# For more information, send mail to:
#
# zchen@cs.uiuc.edu
# Zhigang Chen, Department of Computer Science
# University of Illinois at Urbana-Champaign
#
# format:
# i1 h_size v_size frame rate bit rate frames total size
# i2 ave_size i_size p_size b_size ave_time i_time, p_time, b_time
# p1 pattern of first sequence
# p2 pattern of the rest of the  sequence
# hd header_start header_end
# frame_number frame_type start_offset frame_size frame_time
# ed end start
# ---------------------------------------------------------------------------
#
i1 160 112 15 262143 12216 8941060
i2 731 2152 510 76 12511 20911 10443 8826
p1 7 ipbbibb
p2 7 ipbbibb
hd 0 12
0 1 12 2234 20377
...
```

A link editor enables the user to embed hyperlinks into video streams. The specification of a hyperlink for a object within video streams includes several parameters:

1. The start frame where the object appears and the object's position.
2. The end frame where the object exists and the object's position.

The positions of the object outline are interpolated for frames nestled in between the first and last frames specified. A simple scheme using linear interpolation is shown in Figure 24. The position of the outline in the start frame (frame 1) and end frame (frame 100) are specified by the user. For frames in between, the position is interpolated, as shown, for example, in the frame 50.

In the currently preferred embodiment, linear interpolation is employed, and works well for objects with linear movement. However, for better motion tracking, sophisticated interpolation methods, such as spline interpolation, may be desirable.

With respect to dynamic composition of video, for example, Figure 21 illustrates the result of a search on a video database. The search result is a server-generated dynamic composition of the matched clips. The resulting presentation is a movie made up of the video clips in the search result.

In general, users may use the dynamic composition facilities of the invention to create and author continuous media presentations by reusing video segments through this facility. The organization of video through dynamic composition reduces the need for the copying of large video and audio documents.

Video segmentation and semantic description editing currently is performed manually. Video frames are grouped and descriptions are associated with the groups. The descriptions are stored and used for search and hierarchical structure presentation.

Meta-information and continuous media have been the subject of several studies. The Informedia project at CMU has proposed the use of automatic video segmentation and audio transcript generation for building large video libraries. Algorithms have been proposed for video segmentation. Hyperlinks in video streams have been proposed and implemented in the Hyper-G distributed information system, as well as in a World Wide Web context in Vosaic.

While previous work has focused on a particular aspect of meta-information, for example, in terms of support for search only, or for hyperlinking only, the present invention categorizes and integrates continuous media meta-information in order to support continuous media network transmission, access methods, and authoring. This approach can be generalized for static data. The generalized approach encourages the integration of continuous media with static media, document retrieval with document authoring. Multiple views of the same physical media are possible.

By integrating meta-information in the continuous media approach, flexible access and efficient reuse of continuous media in the World Wide Web are achieved. Several classes of meta-information are included in the continuous media approach. Inherent properties help network transmission of and provide random access to continuous media. Structural information provides hierarchical access and browsing. Semantic specifications allow search in continuous media. Annotations enable hyperlinks within video streams, and therefore facilitates the browsing and organization of irregular information in continuous media and static media through hyperlinks. The support of multiple semantic descriptions and annotations makes multiple views of the same material possible. Dynamic composition of video and audio is made possible by frame addressing and hyperlinks.

While the invention has been described in detail with reference to preferred embodiments, it is apparent that numerous variations within the scope and spirit of the invention will be apparent to those of working skill in this technological field. Consequently, the invention should be construed as limited only by the appended claims.

What is claimed is:

1. System for transmitting real-time continuous media information over a network, said continuous media information comprising video information and audio information, said system comprising:

a server;

a client connected to said server;

communicating means for communicating control information between said server and said client, and for transmitting said continuous media information from said server to said client; and

moderating means for causing said server to change its rate of transmission of said video information when a quality of transmission of said video information changes by a predetermined amount within a predetermined time.

2. A system as claimed in claim 1, wherein a change in said quality of transmission of said video information includes a change in an amount of loss of said video information.

3. A system as claimed in claim 1, wherein a change in said quality of transmission of said video information includes a change in an amount of jitter in said video information.

4. A system as claimed in claim 1, wherein a change in said quality of transmission of said video information includes a change in an amount of latency in said video information.

5. A system as claimed in claim 1, further comprising a plurality of clients connected to said server, said communicating means communicating said control

information between said server and each of said clients, said control information being transmitted separately between said server and each respective one of said clients.

6. A system as claimed in claim 1, wherein said communicating means comprises:

a first channel for communicating said control information between said server and said client; and

a second channel for transmitting said continuous media information from said server to said client.

7. A system as claimed in claim 6, further comprising performance means, responsive to said client, for compiling first performance information about said client and providing an output to said server accordingly, said moderating means causing said server to change its rate of transmission of said video information when said quality of transmission of said video information changes by said predetermined amount between consecutive measurements of said first performance information.

8. A system as claimed in claim 7, wherein said second channel also transmits said output of said performance means from said client to said server.

9. A system as claimed in claim 7, wherein said performance means further is responsive to said communicating means for compiling second performance information about said communicating means and providing a further output to said server, said moderating means causing said server to change its rate of transmission of said video information when said quality of transmission of said

video information changes by said predetermined amount between consecutive measurements of said first and second performance information.

10. A system as claimed in claim 6, wherein said first channel includes a first communications protocol.

11. A system as claimed in claim 7, wherein said first communications protocol is Transmission Control Protocol (TCP).

12. A system as claimed in claim 6, wherein said network is the Internet.

13. A system as claimed in claim 1, wherein said moderating means causes said server to transmit said video information at a slower rate when said predetermined amount is above an engineering threshold.

14. A system as claimed in claim 1, wherein said moderating means causes said server to transmit said video information at a faster rate when said predetermined amount is below an engineering threshold.

15. A system as claimed in claim 7, wherein said moderating means causes said server to transmit said video information at a slower rate when said predetermined amount is above an engineering threshold.

16. A system as claimed in claim 7, wherein said moderating means causes said server to transmit said video information at a faster rate when said predetermined amount is below an engineering threshold.

17. A system as claimed in claim 9, wherein said moderating means causes said server to transmit said video information at a slower rate when said predetermined amount is above an engineering threshold.

18. A system as claimed in claim 9, wherein said moderating means causes said server to transmit said video information at a faster rate when said predetermined amount is below an engineering threshold.

19. A system as claimed in claim 1, wherein said server comprises:

a main request dispatcher for receiving requests from said client for transmission of said continuous media information;

an admission controller, responsive to said main request dispatcher, for determining whether to service said requests, and advising said main request dispatcher accordingly; and

a continuous media handler for processing requests for continuous media information from said main request dispatcher.

20. A system as claimed in claim 19, wherein said continuous media handler separates said requests for continuous media information into requests for video information and requests for audio information, said server further comprising:

a video handler for processing said requests for video information; and

an audio handler for processing said requests for audio information.

21. A system as claimed in claim 9, wherein said server comprises a logger for recording statistics concerning said first and second performance information.

22. A system as claimed in claim 1, wherein said control information includes a play command from said client to said server to play said continuous media information; a stop command from said client to said server to halt transmission of said continuous media information; a rewind command from said client to said server to play said continuous media information in a reverse direction; a fast forward command from said client to said server to cause said server to play said continuous media information at a faster speed; and a quit command from said client to said server to terminate playback of said continuous media information.

23. A method of transmitting continuous media information over a network, said network having a server and a client connected to it, said continuous media information comprising video information and audio information, said method comprising:

transmitting a request, from said client to said server, for transmission of said continuous media information;

transmitting said continuous media information from said client to said server;

sending control signals from said client to said server to control said transmitting of said continuous media information;

receiving said continuous media information at said client in accordance with said sending step;

detecting congestion in said client and, if there is, advising said server accordingly; and

altering a rate of transmission of said continuous media information from said server to said client based on an outcome of said detecting step.

24. A method as claimed in claim 23, further comprising the step of detecting congestion on said network and, if there is, advising said server accordingly;

said altering step being performed based on an outcome of at least one of said client congestion detecting step or said network congestion detecting step.

25. A method as claimed in claim 23, wherein said network is the Internet.

26. A method as claimed in claim 23, wherein said step of sending control signals is performed over a first channel, and said step of transmitting continuous media information is performed over a second, different channel.

27. A method as claimed in claim 26, wherein said first channel includes a first communications protocol.

28. A method as claimed in claim 27, wherein said first communications protocol is a reliable transfer protocol for transmitting said control signals.

29. A method as claimed in claim 27, wherein communication over said first channel is established before communication over said second channel is established.

30. A method as claimed in claim 27, further comprising the steps of:

after said request is transmitted from said client to said server, evaluating said request at said server to determine whether said request can be granted; and

if said request can be granted, transmitting a grant from said server to said client.

31. A method as claimed in claim 29, further comprising the steps of:

after said request is evaluated at said server, and it is determined that said request can be granted, establishing communication between said client and said server over said second channel;

estimating a round trip time (RTT) for travel of data between said server and said client over said second channel; and

setting an initial transfer rate for transmission of said continuous media information from said server to said client.

32. A method as claimed in claim 30, further comprising the step of, if said request cannot be granted, terminating communication between said server and said client over said first channel.

33. A method of organizing continuous media information, comprising:

dividing said continuous media information into groups of frames; and

for each of said groups of frames, providing at least one keyword corresponding thereto, so that entry of said keyword causes a pointer to be placed at a beginning of said corresponding group of frames.

34. A method as claimed in claim 33, further comprising the step of providing at least one hyperlink in said continuous media information, so that activation of said hyperlink causes a pointer to be placed at a location in said continuous media information corresponding to said hyperlink.

35. A method as claimed in claim 34, further comprising the step of, for each of a plurality of continuous media information, providing at least one hyperlink, so as to

enable compilation of a presentation of continuous media information through activation of each said hyperlink.

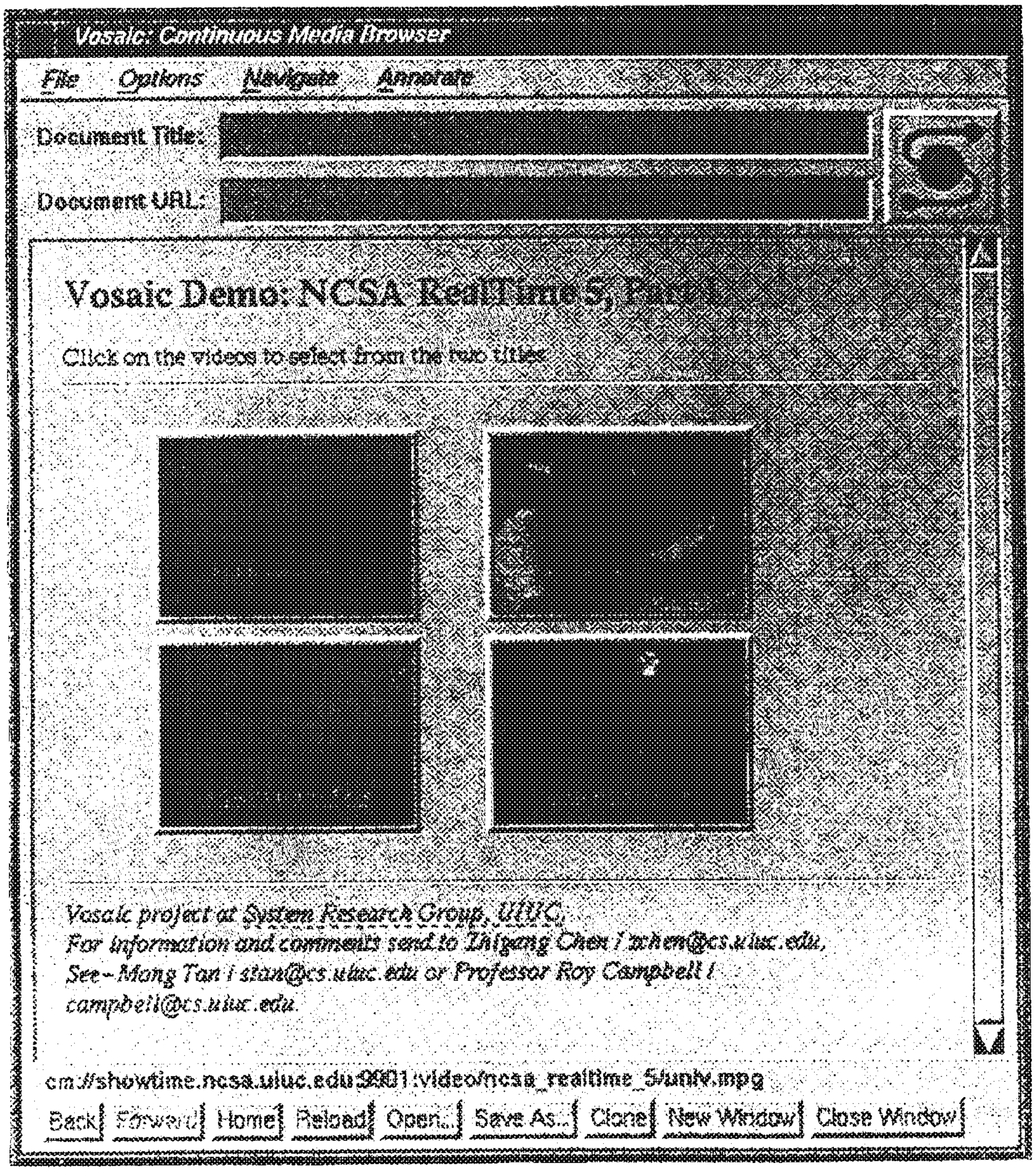

Vosaic: Continuous Media Browser
File Options Navigate Annotate
Document Title:
Document URL:
Vosaic Demo: NCSA RealTime 5, Part
Click on the videos to select from the two titles.
Vosaic project at System Research Group, UIUC.
For information and comments send to Zhigang Chen / zchen@cs.uiuc.edu,
See-Mong Tan / stan@cs.uiuc.edu or Professor Roy Campbell /
campbell@cs.uiuc.edu.
cm://showtime.ncsa.uiuc.edu:9001:video/ncsa_realtime_5/univ.mpg
Back Forward Home Reload Open... Save As... Clone New Window Close Window

FIG.1

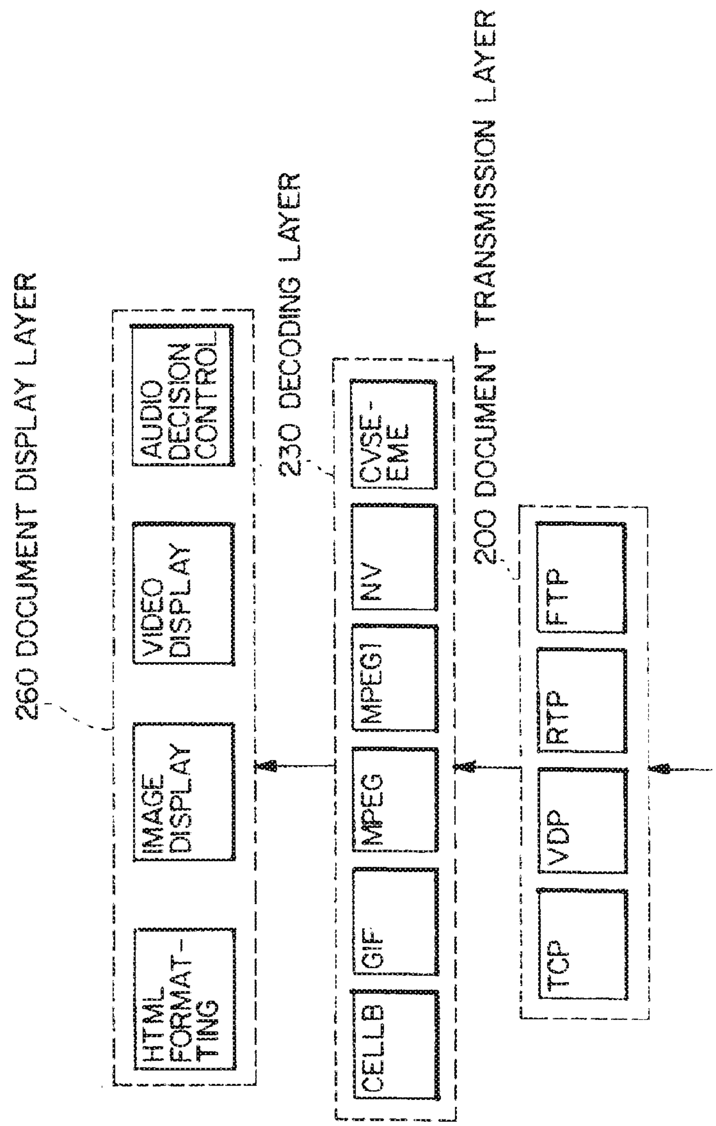

260 DOCUMENT DISPLAY LAYER
HTML FORMAT-TING
IMAGE DISPLAY
VIDEO DISPLAY
AUDIO DECISION CONTROL
230 DECODING LAYER
CELLB
GIF
MPEG
MPEG1
NV
CVSE-EME
200 DOCUMENT TRANSMISSION LAYER
TCP
VDP
RTP
FTP

FIG. 2

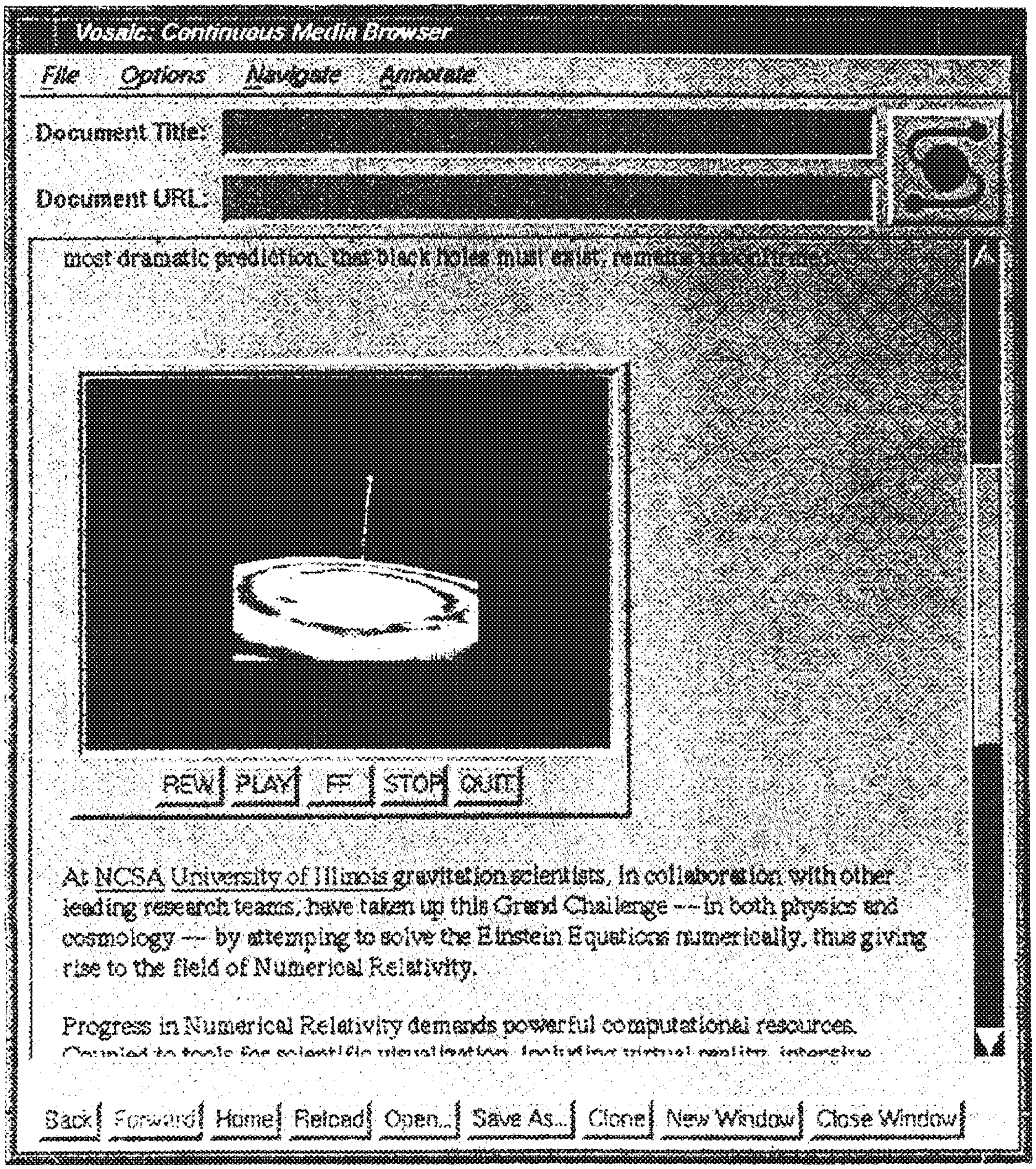

Vosaic: Continuous Media Browser
File Options Navigate Annotate
Document Title:
Document URL:
most dramatic prediction, that black holes must exist, remains unconfirmed.
REW PLAY FF STOP QUIT
At NCSA University of Illinois gravitation scientists, in collaboration with other leading research teams, have taken up this Grand Challenge -- in both physics and cosmology -- by attemping to solve the Einstein Equations numerically, thus giving rise to the field of Numerical Relativity.
Progress in Numerical Relativity demands powerful computational resources.
Back Forward Home Reload Open... Save As... Clone New Window Close Window

FIG.3

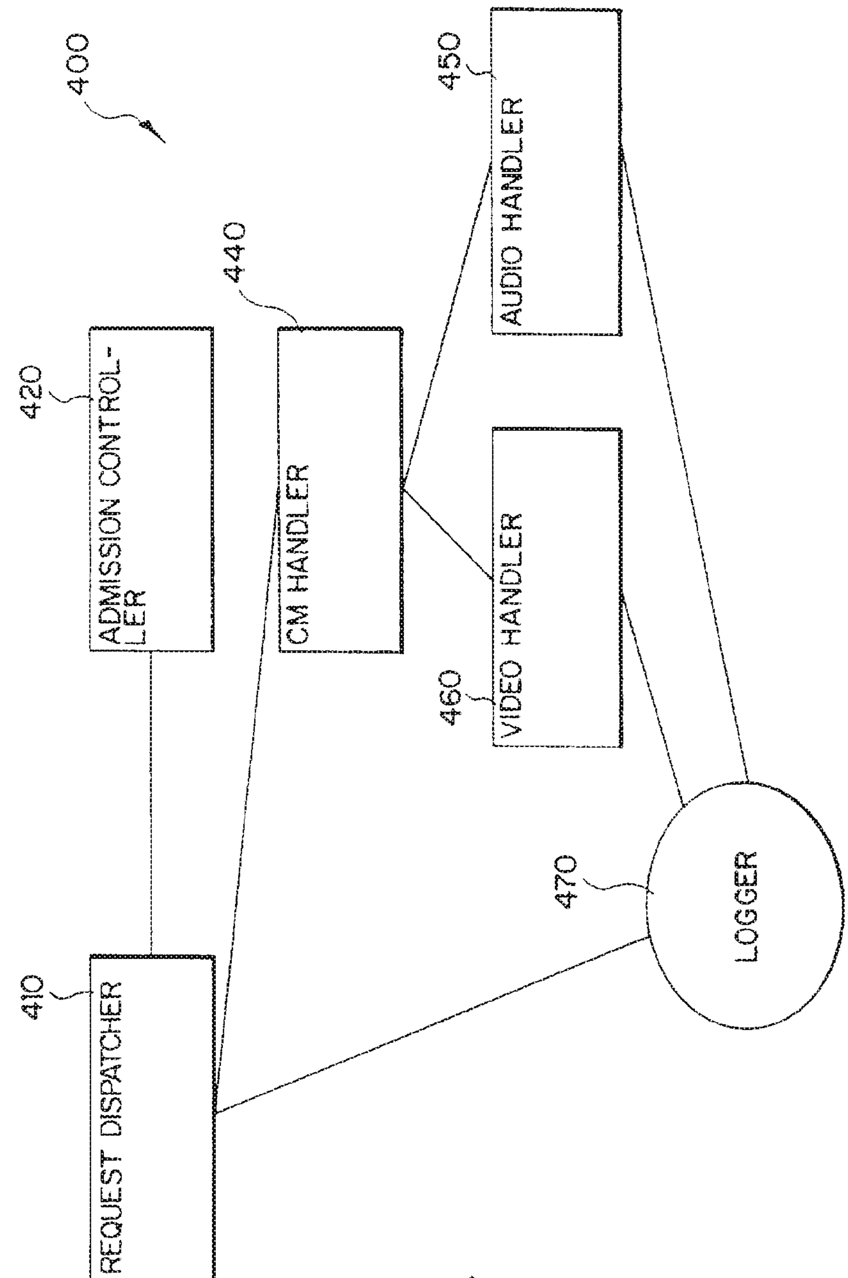

400
410
REQUEST DISPATCHER
420
ADMISSION CONTROL-
LER
440
CM HANDLER
450
AUDIO HANDLER
460
VIDEO HANDLER
470
LOGGER

FIG. 4

FIG. 5

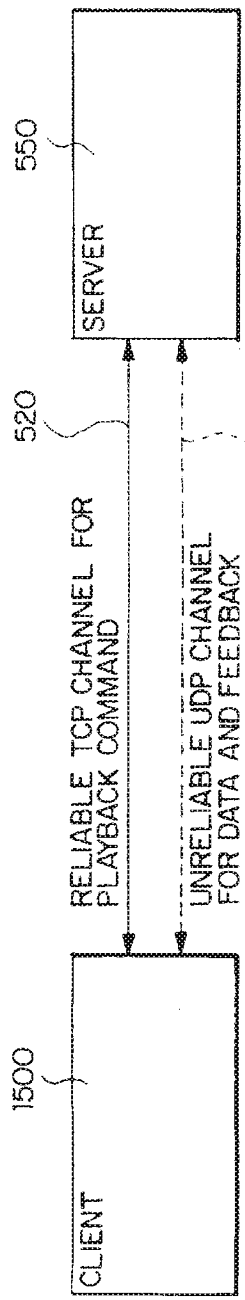

CLIENT
1500
SERVER
550
RELIABLE TCP CHANNEL FOR PLAYBACK COMMAND
520
UNRELIABLE UDP CHANNEL FOR DATA AND FEEDBACK
540

FIG. 7

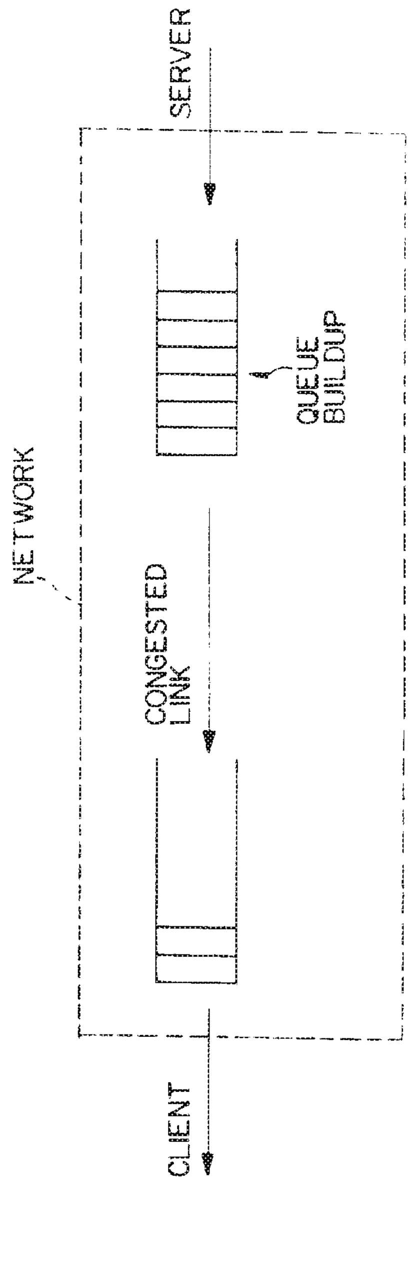

NETWORK
SERVER
QUEUE BUILDUP
CONGESTED LINK
CLIENT

FIG. 6

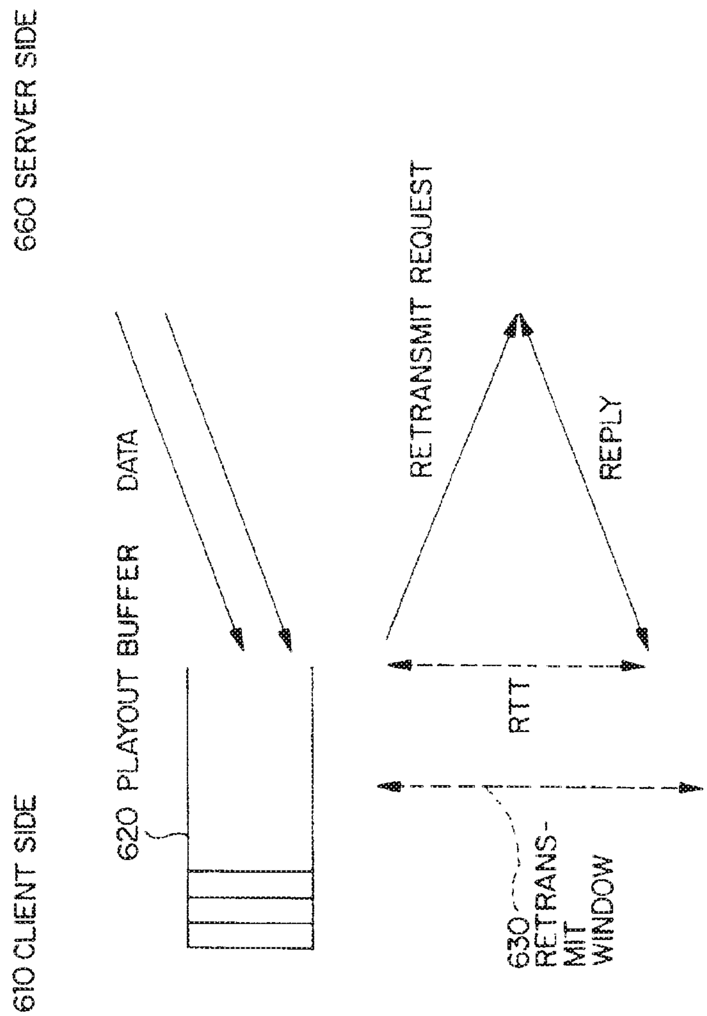
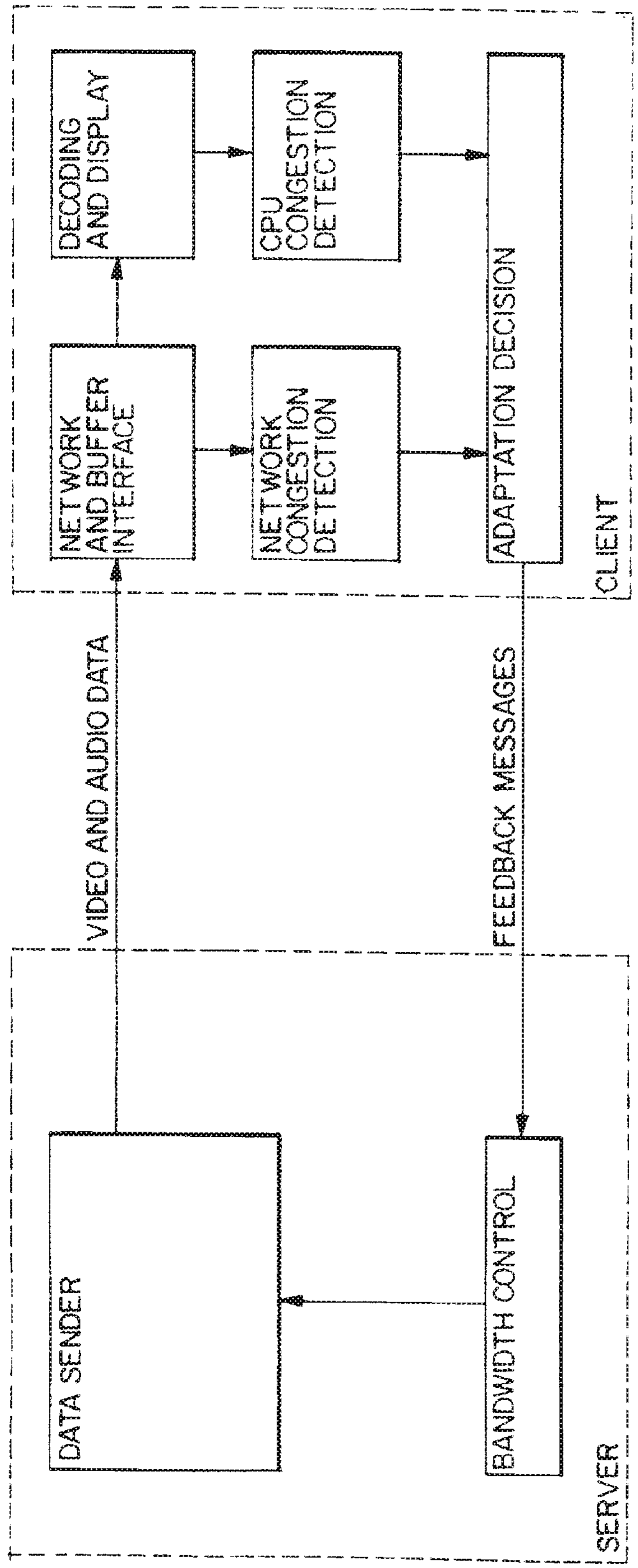

660 SERVER SIDE
610 CLIENT SIDE
DATA
620 PLAYOUT BUFFER
RETRANSMIT REQUEST
REPLY
RTT
630 RETRANS-MIT WINDOW

FIG. 8


DECODING AND DISPLAY
CPU CONGESTION DETECTION
NETWORK AND BUFFER INTERFACE
NETWORK CONGESTION DETECTION
ADAPTATION DECISION
CLIENT
VIDEO AND AUDIO DATA
FEEDBACK MESSAGES
DATA SENDER
BANDWIDTH CONTROL
SERVER