



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2010년11월29일
(11) 등록번호 10-0997296
(24) 등록일자 2010년11월23일

- (51) Int. Cl.
 H04L 29/06 (2006.01) H04L 12/56 (2006.01)
 H04N 7/16 (2006.01) H04N 7/173 (2006.01)
 (21) 출원번호 10-2005-7002700
 (22) 출원일자(국제출원일자) 2003년07월31일
 심사청구일자 2008년06월09일
 (85) 번역문제출일자 2005년02월17일
 (65) 공개번호 10-2005-0071478
 (43) 공개일자 2005년07월07일
 (86) 국제출원번호 PCT/US2003/024175
 (87) 국제공개번호 WO 2004/017602
 국제공개일자 2004년02월26일
 (30) 우선권주장
 10/632,003 2003년07월30일 미국(US)
 60/403,995 2002년08월17일 미국(US)
 (56) 선행기술조사문헌
 US20020012443 A1*
 EP01189403 A1
 WO2000048375 A1
 WO2001098920 A1
 *는 심사관에 의하여 인용된 문헌

- (73) 특허권자
 디즈니엔터프라이즈, 인크.
 미합중국캘리포니아주버어뱅크시사우스뷰나비스
 타스트리트500(우편번호91521)
 (72) 발명자
 와트슨 스코트 에프
 미국 캘리포니아 91287 산타 클라리타 마이클 크
 레스트 드라이브15355
 하셀턴 에릭 씨
 미국 메릴랜드 21113 오턴톤 #103 미드웨이 브랜
 치 드라이브 2616
 (뒷면에 계속)
 (74) 대리인
 이재민

전체 청구항 수 : 총 18 항

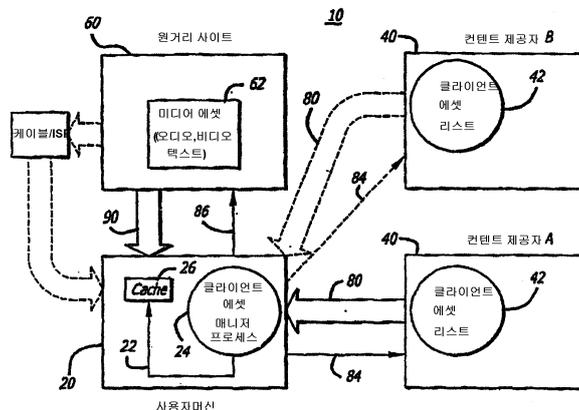
심사관 : 김재문

(54) 제한된 네트워크 대역폭을 통한 대량의 미디어 에셋의 동적프리젠테이션과 전달을 위한 시스템

(57) 요약

컨텐츠 제공자로부터 예정된 제한에 기반하는 미디어 컨텐츠는 미디어를 관람하기 전에 사용자 장치의 로컬 캐시에 전달된다. 클라이언트 에셋 매니저는 사용자장치에 구비되고 컨텐츠 제공자에서의 에셋리스트, 미디어 에셋은 원거리 사이트에 구비된다.

대표도 - 도1



(72) 발명자

프리맨 에릭

미국 워싱턴 98110 바인브리지 아이슬랜드 워드 애
비뉴 5719

프리맨 엘리자베스 엠

미국 워싱턴 98110 바인브리지 아이슬랜드 워드 애
비뉴 5719

라버지 아론 피

미국 코네티컷 06013 버링톤 라이언 로드 62

프리츠 아담 티

미국 워싱턴 98121 시애틀 #710 웨스턴 애비뉴
2922

특허청구의 범위

청구항 1

삭제

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템에 있어서,

클라이언트 프로세스에 네트워크를 통해 콘텐츠 제공자에 의해 이용가능하도록 제작되고 임의의 사용자에게 열람되기 전에 상기 클라이언트 프로세스에 공개되는 것이 가능한 에셋리스트와; 상기 클라이언트 프로세스는 사용자 장치에서 작동가능하고,

소정 제한이 만족될 때 상기 사용자장치에 네트워크를 통해 원거리로부터 이용가능하도록 제작된 에셋 및; 상기 에셋은 상기 에셋의 전달이 완성된 후에 상기 사용자 장치에 저장되고,

상기 네트워크를 통해 원거리로부터 사용자에게 이용가능하고 상기 에셋과 통합되는 것이 가능한 리얼타임컨텐츠 스트림;을 포함하며,

상기 에셋리스트와 상기 리얼타임컨텐츠 스트림은, 상기 에셋리스트와 상기 리얼타임컨텐츠 스트림의 전달이 완성된 후에 사용자 장치에 저장되는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 27

제26항에 있어서,

네트워크를 통해 원거리로부터 사용자장치에 의해 수신되는 콘텐츠 스트림과 전달되는 에셋을 통합하기 위한 통합도구를 포함하는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 28

제26항에 있어서,

상기 예셋은 오디오콘텐츠, 2진데이터콘텐츠, 비디오콘텐츠, 텍스트콘텐츠 또는 멀티미디어 파일 중의 적어도 하나인 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 29

제26항에 있어서,

상기 예셋리스트는 클라이언트프로세스에 콘텐츠 제공자에 의해 제공되는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 30

제26항에 있어서,

상기 클라이언트 프로세스는 콘텐츠제공자의 콘텐츠 웹사이트를 액세스하는 단계를 포함하는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 31

제26항에 있어서,

상기 예셋은, 소정 제한이 아이들(IDLE)되는 사용자장치 또는 동작레벨 이하의 대역폭사용(a bandwidth usage) 중에 적어도 하나를 포함할 때, 이용될 수 있는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 32

제26항에 있어서,

상기 예셋은, 소정 제한이 사용자의 CPU 사용(a cpu usage) 또는 동작레벨 이하의 사용자장치의 메모리 사용(a memory usage)을 포함할 때, 이용될 수 있는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 33

제26항에 있어서,

상기 클라이언트 프로세스는 상기 사용자장치에 네트워크를 통해 상기 콘텐츠 제공자로부터 상기 예셋의 전달을 초기화할 수 있는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 34

제26항에 있어서,

상기 예셋은 로컬 캐시에 저장되는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 35

제34항에 있어서,

상기 리얼타임 콘텐츠와 결합하여 저장된 예셋을 상기 리얼타임 콘텐츠 스트림으로부터 프리젠틱하는 수단을 포함하는 것을 특징으로 하는 네트워크를 통해 콘텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 36

제26항에 있어서,

상기 예셋은, 소정 제한이 일정 시각을 포함하면 이용가능하도록 제작되는 것을 특징으로 하는 네트워크를 통해

컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 37

제34항에 있어서,

사용자장치 CPU 사용, 대역폭사용, 로컬캐시사용 및 사용자장치 활성타이머로부터 소정 제한을 결정하는 수단을 부가하여 포함하는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 38

제34항에 있어서,

상기 예셋이 사용자장치에서 이용할 수 없는 경우에 리얼타임컨텐츠와 결합하는 대용예셋을 프리젠틱하는 수단을 포함하는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 39

제26항에 있어서,

상기 예셋리스트는 주기적으로 업데이트되는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 40

제26항에 있어서,

상기 클라이언트 프로세스는 다수의 예셋리스트와 결합되는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 41

제26항에 있어서,

상기 예셋리스트는 만료일, 콜백 URL, 클라이언트 측 토큰, 드로틀 파라미터, 리프레쉬 레이트 파라미터, 삭제 예셋 플래그, 도움말링크(a help link) 및 리소스 경로정보 중에 적어도 하나를 포함하는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 42

제26항에 있어서,

상기 예셋은 사용자장치에 예셋을 전달하기 전에 인터넷 정보제공업자 또는 케이블 제공자 중의 적어도 하나에 전달되는 것을 특징으로 하는 네트워크를 통해 컨텐츠 스트림을 사용자에게 프리젠틱하기 위한 시스템.

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

저장과 관람을 위한 셋톱박스에서 미디어 에셋을 수신하는 방법에 있어서,

셋톱박스에서 콘텐츠 제공자로부터 임의의 사용자에게 열람되기 전에 클라이언트 프로세스에 공개되는 미디어 에셋 리스트(a media asset list)를 수신하는 단계와; 상기 미디어 에셋 리스트는 다운로드되는 미디어 에셋의 리스트(a list of media asset)와 미디어 에셋의 위치에 대한 정보를 포함하고,

상기 셋톱박스에서 상기 클라이언트 프로세스를 구동하는 단계와; 상기 클라이언트 프로세스는 어떤 미디어 에셋이 셋톱박스에 전달되는지를 판단하기 위해 상기 미디어 에셋 리스트를 읽는 것이 가능하고, 상기 클라이언트 프로세스는 소정 제한에 따라 디지털 미디어 에셋의 전달을 관리하고,

소정 제한이 만족되면 상기 셋톱박스에 상기 콘텐츠 제공자로부터 디지털 미디어 에셋을 다운로드하는 단계와;

상기 셋톱박스에서 다운로드된 디지털 미디어 에셋을 저장하는 단계; 및

TV 또는 다른 디스플레이장치에서 사용자에게 의한 리얼타임 콘텐츠 스트림과 저장된 디지털 미디어 에셋을 통합하는 단계;를 포함하는 것을 특징으로 하는 저장과 관람을 위한 셋톱박스에서 미디어 에셋을 수신하는 방법.

명세서

기술분야

[0001] 본 발명은 네트워크 대역폭의 이용이 불충분하더라도 네트워크를 통해 사용자 또는 소유자에게 보증된 서비스 품질(QoS, the network Quality of Service)을 갖는 고품질 미디어 기반 콘텐츠를 전달하고 프리젠틱하는 기술에 관한 것이다.

배경기술

[0002] 본 발명자는 네트워크 기술분야에서 진보적이고 풍부한 경험을 가지고 있다. 대량의 미디어 에셋을 전달하는 기술 - 미디어 에셋은 오디오, 비디오, 플래시, 게임, 데이터 또는 다른 디지털 미디어 포맷이 될 수 있다 - 네트워크 대역폭/작업량이 더 필요로 하고 있다. 예를 들어 오디오와 비디오의 경우에서 높은 비트레이트의 에셋이 에셋의 비트레이트와 사용자의 대역폭이 적어도 동일하다면 실제로 전달될 수 있다. 다시 말해 결과는 중얼거림, 정지 및 콘텐츠 버퍼링을 완성하는 차선의 사용자 경험이다.

[0003] 다시 말해 대량으로 실행가능한 게임은 비디오와 같은 동일한 리얼타임 제한을 가지고 있지 않은 반면에 에셋의 다운로드드는 빠른 네트워크에서 사용자를 위한 상당한 시간과 오버헤드를 필요로 한다. 시장에서 "다운로드 메니저"의 숫자가 사용자를 위해 주의깊게 되고 콘텐츠공급자는 사용자의 주의를 필요로 함이 없이 멋지고 명백한 방법 내에서 사용자장치에 에셋을 다운로드를 지능적이고 적합하게 사용자장치(예; 컴퓨터, 메모리 및/또는 프로세서를 갖는 셋톱박스, 장치)에 에셋의 다운로드를 관리하는 것이 요망될 것이다.

[0004] 이러한 점들에서 시간 시프트된 방법 내에서 제한된 폭을 이용하여 사용자에게 크고 고품질의 미디어 에셋을 전달하고 관리하는 것이 필요하다. 즉, 이용가능한 대역폭과 아이들(IDLE) 사이클을 통해 사용자에게 콘텐츠를 전달되게 하는 것이 필요하고 고품질의 콘텐츠가 필요될 때 콘텐츠는 수요에 따라 용이하게 이용할 수 있고 집념을 필요로하는 사용자 경험이 행해진다. 사용자는 실제로 이용할 수 있는 것보다 효과적인 대역폭을 가지고 있는 환상을 제공한다. 결국 전달과 표현 플랫폼(웹브라우저, 플래시 및 다른 플랫폼을 포함하고 한정하지는 않는다)과 콘텐츠 퍼블리시 시스템으로 연결없이 통합되는 기술을 위해 필요하다. 본 발명은 이러한 것과 다른 기능을 달성하고 이전 기술의 한계를 극복하는 것이다.

[0005] 이해를 돕기 위해 이하의 정의는 어플리케이션 전체에 걸쳐 제공될 것이다; 그러나 이러한 정의는 리스트된 용어의 이해를 용이하도록 한 것으로 한정하지는 않는다.

[0006] 설명:

[0007] 1. 데이터처리량 - 특정 시간내에서 하나의 장소에서 다른 장소로 전달되는 데이터의 양. 일반적으로 데이터처리량은 kbps, Mbps 및 Gbps 로 측정된다.

[0008] 2. 네트워크 서비스품질(QoS, the network Quality of Service) - 위 용어는 보증된 데이터처리량 레벨을 특정화하는 것이다.

- [0009] 3. 클라이언트 프로세스 - 캐시/디스플레이 관리를 수신하거나 서버프로세스로부터 알 수 있는 클라이언트 사에 서의 프로세스가 바람직한 상태로 라인에서 캐시 현재상태를 가져오는 것을 실행하고 하나를 트리거하거나 사용 자들에게 경고를 한다.
- [0010] 4. 캐시 - "알려진" 이용가능성 또는 QOS 를 갖는 에셋의 저장. 이러한 컨텍스트 내에서의 캐시는, QOS 가 컨 텐트 요구사항을 만나고 에셋에 이용되는 미디어 보다 일반적으로 더 높다,
- [0011] 5. 서버프로세스 - 캐시의 상태를 관리하는 클라이언트를 위해 필요로되는 정보를 클라이언트에게 제공한다. 가장 간단한 실행에서 프로세스는 동적서버가 플레이리스트를 발생하는 것과 유사하다. 더욱 정교한 수행은 에 셋의 특정 세트에 대한 다른 프로세스의 진행을 알려주는 클라이언트에 대한 제어를 제공한다.
- [0012] 6. 만료일 - 에셋의 만료일, 에셋이 로컬캐시로부터 제거될 때를 지시한다.
- [0013] 7. 콜백 URL - 다운로드된 에셋을 복구되게 한다.
- [0014] 8. 클라이언트-측-토큰 - 아이템이 다운로드될 때 세트를 위한 토큰 또는 쿠키. 이것은 로컬 시스템에서 에셋 의 존재를 결정하도록 클라이언트 또는 어플리케이션을 허용한다.
- [0015] 9. 엠바고 날짜 - 이것은 에셋이 이용될 수 있는 최종날짜를 지시한다.
- [0016] 10. 삭제 - 에셋이 일정삭제를 위해 표기될 수 있게 지시되는 것이다. 이것은 에셋의 취소를 허용한다.
- [0017] 11. 다시읽기 레이트 - 클라이언트가 변화를 위해 에셋리스트를 얼마나 자주 체크하는지를 결정.
- [0018] 12. 리소스경로 - 리소스의 네트워크 위치는 에셋리스트와 결합.
- [0019] 13. 미디어 에셋 - 텍스트, 오디오 비디오 또는 2진 파일/데이터 중의 적어도 하나.
- [0020] 14. 아이템 - 단일 미디어 파일
- [0021] 15. 링크 - 미디어 파일에 대한 URL
- [0022] 16. 히트카운트 Url - 파일이 성공적으로 다운로드된 이후에 ping하는 URL. 파라미터, 시간이 수초안에 URL 지시 부의 말단에 추가되고 많은 다운로드가 일어난다.
- [0023] 17. 디스플레이 Url - 사용자가 디스플레이 메뉴 아이템을 선택했을 때 디스플레이되는 클라이언트 프로세스 디 스플레이를 위한 URL.
- [0024] 18. 트랙워드쿠키 - 선택적인 요소가 존재한다면 쿠키이름에 의해 특정화되는 쿠키 내에서 에셋의 리스트로 더 해지는 에셋을 지시한다.
- [0025] 19. 쿠키이름 - 쿠키의 이름은 트랙워드쿠키요소를 갖는 모든 다운로드된 에셋을 리스트한다. 이러한 쿠키는, ad 서버가 다운로드되는 것을 알도록 ad 서버를 위해 효과적이다. 상기 쿠키의 형태는 코마에 의해 분리되는 파일의 이름을 구성할 것이다.
- [0026] 20. 쿠키도메인 - 다운로드되는 에셋 쿠키를 세트하기 위한 도메인. 다중도메인이 세미콜론 또는 콤마에 의해 분리된다면 상세화될 수 있다.
- [0027] 21. /레그서버 - 시스템과 함께 액티브 엑스컨트롤을 등록하고 스타트업 폴더로 클라이언트 프로세스를 더한다.
- [0028] 22. /셋다운 - 클라이언트 프로세스의 또 다른 구동을 정지한다.
- [0029] 23. /언레그서버 - 액티브 엑스컨트롤을 언레지스터하고 스타트업 폴더로부터 슛컷을 제거한다. 또한 클라이 언트 프로세스(clientprocess.exe)의 구동을 정지하고 COM 오브젝트 레지스트리를 제거한다.
- [0030] 24. CDN - 콘텐츠 분산 네트워크. 콘텐츠서버의 연합그룹은 3rd 부분에 의해 소유되고 동작된다. 실제로 CDN 서비스는 서버들의 분산된 수집을 이용하여 추가적인 용량을 제공한다.

발명의 상세한 설명

[0031] 본 발명은 미디어 콘텐츠가 미디어를 열람하기 전에 소정 제한에 따라 사용자 장치의 콘텐츠 제공자로부터 로컬 캐시에 전달되는 시스템 및 방법을 제공하는 것을 목적으로 한다.

본 발명에서 에셋리스트는 클라이언트장치에 다운로드되는 미디어에셋에 관한 정보를 포함하고 콘텐츠제공자로

부터 사용자장치에 전달된다. 상기 에셋리스트는 미디어에셋의 위치에 관계되는 정보 또는 URL을 포함한다.

- [0032] 클라이언트 에셋 매니저 프로세스는 사용자장치 내에 구비되고 콘텐츠 제공자로부터 에셋의 다운로드를 위해 구비된다. 상기 에셋 매니저는 원거리 사이트에서 구비되는 미디어 에셋을 요청하는 에셋리스트를 이용한다. 상기 클라이언트 프로세스는, 특정 제한이 발생될 때 주기적으로 사용자장치의 에셋의 전달을 관리한다. 예를 들어 에셋은 최적 네트워크폭 이용성, 사용자 장치 메모리 서비스의 보증된 성질이 있을 때 사용자에게 전달된다.
- [0033] 본 발명은 연속적이고, 언인터럽트되고 사용자의 미디어 에셋의 효과적인 전달에 의해 미디어 콘텐츠의 연결없는 디스플레이를 제공하는 것이다. 관람수단과 에셋을 통합하여 사용자는 언인터럽트되고 연속적인 리얼-타임 버퍼링을 요하지 않는 미디어 콘텐츠의 스트림이 제공된다.
- [0034] 본 발명은 하나 또는 그 이상의 미디어 에셋, 예를 들면 로컬 에셋스토어의 네트워크를 통하여 지능적이고 적합한 오디오 콘텐츠, 비디오 콘텐츠, 영화, 게임 등을 전달하기 위한 개선된 방법을 제공하는 것이다. 또한 로컬 에셋스토어는 클라이언트 또는 소비자등에 의해 이용될 수 있고 고품질의 서비스가 보장된다. 본 발명은 또한 에셋의 국소적 이용에 기반하여 효과적으로 이음새가 없는 프리젠테이션과 에셋을 조합하는 적응방법을 포함한다.
- [0035] 본 발명은 네트워크를 통해 에셋의 전달을 위한 방법을 포함하는 것이다. 상기 방법은 사용자장치에 네트워크를 통해 에셋리스트를 적용하는 것을 포함하는 것이다. 상기 방법은 사용자장치에 동작하는 클라이언트를 포함하는 것이다. 상기 클라이언트는 사용자장치에 에셋을 전달하고 다운로드하여 에셋리스트를 참조한다. 상기 클라이언트는 적어도 하나의 소정 제한이 만족될 때 다운로드한 에셋을 관리한다.
- [0036] 콘텐츠제공자는 사용자장치에 디지털 에셋을 구비하고 사용자가 필요로할 때 사용을 위해 즉시 이용가능하다. 이것은 사용자 또는 프로세스에 의해 요청될 때 일어날 수 있거나 출자금서비스에 기반하여 콘텐츠 공급자에 의해 초기화된다.

실시예

- [0045] 개요
- [0046] 미디어 에셋의 충분한 전달에 의한 연속적이고, 언인터럽트하고 이음새 없는 디스플레이의 미디어 콘텐츠가 사용자에게 전달된다. 특히 에셋은 특정 제한이 발생될 때(예., 네트워크 대역폭, 사용자장치메모리, 보증된 QOS) 주기적으로 사용자장치에 전달된다. 관람수단을 갖는 에셋을 통합하여, 사용자는 리얼타임 버퍼링을 필요로하지 않는 언인터럽트하고 연속적인 미디어 콘텐츠의 스트림이 제공된다.
- [0047] 하나 또는 그 이상의 대량의 미디어 에셋, 예컨대 오디오 콘텐츠, 비디오 콘텐츠, 게임 등이 로컬 에셋스토어에 네트워크를 통해 지능적이고 적합하게 전달된다. 또한 로컬 에셋스토어는 클라이언트 또는 소비자 등에 의해 이용될 수 있고 고품질의 서비스가 보장된다. 본 발명은 또한 에셋의 국소적 이용에 기반하여 효과적으로 이음새가 없는 프리젠테이션과 에셋을 조합하는 적응방법을 포함한다.
- [0048] 로컬 에셋스토어의 실시에는 QOS가 보장될 때 사용자장치가 연결되는 LAN에 의해 제공되는 데이터저장장치 또는 캐시가 될 수 있다. 국소적 이용성에 의해 에셋은 소비자로컬에서 이용가능하거나 프로세스저장시스템(클라이언트장치와 로컬스토어 사이에서 고속의 네트워크를 갖는 로컬스토어)에서 이용가능함을 의미한다.
- [0049] 프로세스 또는 어플리케이션이 사용자의 장치에서 구동되는 바람직한 실시예에서 클라이언트로서 참조된다. 상기 클라이언트는 사용자의 로컬 스토어에서 콘텐츠의 캐시를 관리하도록 구비된다. 어플리케이션, 웹페이지 및 멀티미디어 프리젠테이션은 이미 상주하는 대량의 미디어 에셋을 조직하는 캐시에 의존할 수 있다. 어플리케이션, 웹페이지 및 멀티미디어 프리젠테이션은 콘텐츠 캐시를 질의할 수 있고 동적으로 사용자는 이용가능한 콘텐츠를 경험할 수 있다. 이러한 어플리케이션은 클라이언트 또는 서버측 어플리케이션이 될 수 있다.
- [0050] 상기 클라이언트는 사용자의 장치에서 인스톨되고 인터넷으로부터 다운로드되는 서비스가 될 수 있다. 상기 클라이언트는 컴퓨터 오퍼레이팅 시스템의 요소로서 공급될 수 있거나 다른 소프트웨어 어플리케이션과 변들로 될 수 있다.
- [0051] 상기 클라이언트는 소비자를 위한 미디어 에셋의 캐시를 관리한다. 실시예에 있어서 캐시 내에 포함되는 에셋은 콘텐츠 제공자에 의해 제공되는 에셋리스트에 의해 지시된다. 실시예에서 클라이언트는 네트워크 영역으로부터 특정 그리고 회전가능한 시간간격에서 에셋리스트를 얻는다. 에셋리스트 내의 정보에 기반하여 클라이언

트는 예정된 제한이 만족될 때 원거리로부터 에셋의 다운로드를 관리한다.

- [0052] 클라이언트는 하나의 에셋리스트 보다 그 이상의 것과 결합되게 허용되어 많은 서비스 제공자 또는 다중 비즈니스 유닛의 서비스 제공자가 다운로드를 관리하는 동일한 클라이언트의 이용을 만들고 에셋리스트를 분리시켜 유지한다. 또한 클라이언트는 클라이언트가 인스톨되거나 업데이트될 때 에셋리스트의 위치를 알 수 있을 것이다. 상기 인스톨은 웹페이지로부터 초기화된다. 클라이언트가 클라이언트 장치 내에 인스톨되지 않았다면 웹 페이지 내에서의 소프트웨어는 이것을 검출하고 다운로드할 수 있도록 사용자를 자극하고 클라이언트를 인스톨한다. 웹페이지 내에서 스크립팅(액티브 엑스 컨트롤과 조합)을 통해 클라이언트에 에셋리스트를 추가하는 것이 가능하다.
- [0053] 상기 클라이언트는 예정된 제한에 기반한 다운로드된 에셋을 관리한다. 상술한 바와 같이 소정 제한은 네트워크 대역폭 유효성, 사용자장치 메모리, 하루시각 및 서비스의 보증된 품질을 포함한다. 예를 들어 클라이언트의 실시예에서 사용자장치가 아이들(IDLE)인 경우에 다운로드가 일어난다. 또 다른 실시예에서 클라이언트 프로세스는 네트워크활성이 소정 레벨 이하일 때 다운로드한다. 또 다른 실시예에서 클라이언트 프로세스는 사용자 장치의 CPU 및 메모리처리를 측정한다. 또 다른 실시예에서 클라이언트 프로세스는 하루시간에 기반하여 네트워크처리가 가장 낮을 때 다운로드한 에셋을 관리한다. 상기 클라이언트 프로세스는 부가적으로 다운로드 에셋을 관리하는 제한의 조합을 이용한다.
- [0054] 다운로드 용량은 사용자에게 의해 온/오프 스위치될 수 있다. 상기 사용자는 장치가 네트워크에 있을 때 다운로드가 일어나는 것을 상세화할 수 있다(LAN 과 다이얼업 연결). 예를 들어 사용자는 에셋이 아침관람을 위해 오후 늦게 몇시간 동안 다운로드되도록 콘텐츠 서비스에 신청할 수 있다.
- [0055] 또한 에셋의 지능적이고 적응적인 관리는 사용자에게 의한 제어하에서 명백한"다운로드 매니저"를 위해 필요없는 것이 가능하다. 지능적으로 몇몇 예정된 제한이 만족될 때 에셋이 클라이언트에 전달되는 것을 의미한다(네트워크의 대역폭, 하루시간, QOS 등). 적응적으로 콘텐츠 제공자 사이트 또는 사용자 장치 사이트에서 에셋리스트는 사용자 관람물에 기반하여 자동적으로 업데이트된다.
- [0056] 에셋리스트는 클라이언트의 로컬 스토어의 콘텐츠를 광고하고 관리하기 위해 콘텐츠 생산자를 위한 수단으로서 이용된다. 그러므로 이러한 기능은 몇몇 낮은 지점에서 어플리케이션, 웹 페이지 또는 플래시-베이스 프리젠테이션으로 통합되는 미디어 에셋을 광고하는 콘텐츠 생산자를 허용한다. 예를 들어 생산자는 위치를 포함하는 에셋, 프리젠테이션의 날짜, 최종 공표시스템으로 에셋과 결합될 수 있는 텍스트&관련정보를 입력한다. 상기 정보는 관련 에셋리스트로 공표되고 포함된다. 공표시스템은 에셋을 조직하는 웹페이지 또는 프리젠테이션을 공표(즉시, 소정날짜, 미래시간)할 수 있다.
- [0057] 네트워크를 통해 에셋을 전달하기 위한 방법은 i)클라이언트 프로세스는 사용자장치에서 동작을 하고 클라이언트에 네트워크를 통해 콘텐츠 제공자에게 의한 에셋리스트를 제공하는 단계 및 ii) 예정된 제한이 만족될 때 사용자장치에 네트워크를 통해 에셋리스트에 부합하는 에셋을 전달하는 단계를 포함한다. 예를 들어 에셋은 이진데이터, 오디오콘텐츠, 비디오 콘텐츠, 텍스트 콘텐츠, 또는 멀티미디어파일 중의 적어도 하나가 될 수 있다. 예정된 제한은 하루시간, 사용자장치상태(즉, 장치가 이용되는지 아이들링 되는지), 대역폭처리(대역폭처리가 예정된 동작레벨 이하인지), 사용자 장치 CPU 처리, 사용자장치 메모리처리(즉, 메모리처리가 예정된 동작레벨 이하인지)를 포함한다. 부가하여 상기 에셋은 사용자장치의 에셋은 사용자장치의 로컬캐시 내에 저장될 수 있다. 또한 클라이언트는 사용자에게 언인터럽트되고 이음이 없는 콘텐츠의 프리젠테이션을 제공하도록 이음이 없이 콘텐츠 제공자로부터 리얼타임콘텐츠로 저장된 에셋을 통합한다. 또한 에셋리스트는 예정된 제한에 기반하여 주기적으로 다양한 콘텐츠를 수신하는 능력을 갖는 클라이언트 프로세스를 제공하여 자주 업데이트될 수 있다.
- [0058] 상기 시스템은 이하의 것을 포함하는 언인터럽트된 콘텐츠 스트림을 사용자에게 제공한다: (i) 에셋리스트는 클라이언트 프로세스에 인터넷을 통해 콘텐츠제공자에게 의해 이용가능하게 제작하고 클라이언트 프로세스는 사용자 장치에서 동작한다,(ii) 예정된 제한이 만족될 때 사용자장치에 네트워크를 통해 원거리로부터 이용가능하도록 만들고 원거리의 정보는 에셋리스트로부터 얻어지고 (iii) 콘텐츠 스트림과 전달된 에셋을 통합하도록 통합 도구는 인터넷을 통해 원거리로부터 사용자장치에 의해 수신된다. 사용자는 언인터럽트되고 연속적인 콘텐츠 스트림으로 프리젠티된다.
- [0059] 상기 생산자는 저대역폭 미디어 에셋과 부합 텍스트정보를 들어갈 수 있고 텍스트정보는 페이지의 두개의 서로 다른 버전을 생산하기 위한 풍부한 미디어 에셋에서 정보를 갖는 후반부에 결합될 수 있다. 사용자는 여러가지

이하의 이유로 인해 풍부한 미디어 에셋을 다운로드 할 수 없다; (1) 비호환시스템 (2) 시스템이 다운로드를 수신하는 온라인상태가 아니다 (3) 사용자는 프리미엄 서비스를 지불하지 않는다 (4) 사용자는 충분하지 않은 디스크 공간을 가지고 있다 (5) 사용자는 컴퓨터를 사용하고 컴퓨터가 다운로드하기 위해 사용되는 리소스가 되는 것을 원하지 않는다. 이러한 경우에서 대안적인 경험을 전달하는 능력이 중요하게 여겨진다.

[0060] 광고를 갖는 콘텐츠가 전달될 수 있다. 예를 들어 순차적으로 일치하는 광고가 로컬캐시에 존재할 때 플레이되는 콘텐츠를 허용한다. 하나의 실시예에서 두개의 미디어 에셋이 사용자장치에 존재하는지를 결정하기 위한 스크립팅을 이용하여 이루어질 수 있다. 바람직한 실시예에서 이러한 기능은 에셋리스트 번들을 이용하여 달성되고 두개의 에셋이 로컬장치에 있을 때 토큰을 세트한다.

[0061] 사용자는 로컬장치에서 다운로드되고 캐시되는 미디어를 동적으로 상세화할 수 있다. 실시예에서 이것은 각각의 사용자를 위한 에셋리스트를 관리하는 서버측 데이터베이스에 의해 달성된다. 예를 들어 주중에 사용자가 주말에 DVD 영화를 관람하는 것을 원하는 것을 상세화할 수 있다. 이것은 에셋리스트에 추가되는 에셋에서 초래되고 시간을 통해 에셋을 다운로드하는 클라이언트에서 초래된다. 사용자는 상기 리스트에 에셋을 추가하도록 충전될 수 있고 디지털 권한 관리가 에셋을 제어하는데 이용될 수 있다. 다른 실시예에서 이러한 방법은 이하와 같은 콘텐츠 제공자로부터 "푸쉬"로 조합된다; 서비스에 가입한 사용자는 매일 아침 저녁뉴스를 볼 수 있다. 이러한 결과로 서비스는 다운로드되는 새로운 쇼를 구별하기 위해 저녁마다 에셋리스트로 새로운 에셋을 삽입한다.

[0062] 또한 시스템은 ISP 와 시스템 전달 케이블 제공자와 작업된다. 실시예에서 생산자가 에셋을 공표할 때 에셋은 ISP와 장치 또는 시스템을 전달하는 케이블 제공자에게 이동다. 상기 클라이언트는 에셋으로부터 얻어지고 에셋 ISP 을 얻어 로컬캐시에 에셋을 구비한다. 부가하여 콘텐츠 제공자는 근접한 에셋에 기반하여 사용자 경험을 최적화할 수 있다. 예를 들어 웹사이트와 관련된 뉴스는 풍부한 미디어 에셋이 이용할 수 없다면 사진과 같은 뉴스내용을 제공하지만 비디오 에셋이 로컬적으로 저장된다면 웹페이지는 웹 페이지 상의 이미지 장소에 비디오를 대체할 수 있다. 이러한 것은 비디오와 같은 비디오 문자 "카피" 를 포함하고 이미지는 동일한 내용이 될 수 있고 서로 다른 문자와 캡션을 필요로 할 수 있다. 바람직한 실시예에서 이것은 에셋 토큰에 부합하는 것을 통해 에셋의 존재를 검출하는 서버측 어플리케이션을 갖는 것에 의해 달성될 수 있다. 토큰이 존재하는지 안하는지에 따라 다른 페이지가 소비자를 위해 생성될 수 있다. 다른 바람직한 실시예에서 이것은 웹브라우저에서의 스크립팅을 통해 클라이언트 측에서 행해진다. 이것은 웹페이지에서 스크립팅을 이용하여 서로 다른 페이지가 미디어 에셋 토큰의 존재에 기반한 동적 HTML을 통해 생성된다.

[0063] 본 발명에 의해 제공되는 하나의 잇점은 네트워크 대역폭 처리를 최적화하도록 콘텐츠 제공자를 허용하는 것이다. 특히 처리가 피크 이하로 떨어질 때 콘텐츠의 전달을 증가하여 대역폭의 골이 대역폭이용이 최적화되도록 채워진다.

[0064] 또 다른 관점에서 디지털 권한 관리(DRM) 구조는 데이터 흐름에 적용될 수 있다. 예를 들어 디지털 권한 관리(DRM) 보호 에셋은 본 발명을 통해 공표될 수 있다. 그러므로 사용자가 에셋을 "플레이" 할 때 콘텐츠 "사용권리" 와 같은 적합한 자격을 충족하도록 자극한다.

[0065] 또 다른 관점에서 게임에셋은 클라이언트에 이용될 수 있는 리얼 또는 가상 ISO 이미지에 의한 형태로 분배된다.

[0066] B. 셋톱박스 실시예

[0067] 상기 클라이언트는 TV 셋톱박스과 같은 서로 다른 소스로부터 미디어 콘텐츠와 관련된 미디어 에셋을 다운로드 하고 관리하며 에셋은 케이블 또는 위성네트워크 또는 인터넷 또는 인트라넷과 같은 데이터 네트워크를 통해 또는 반소파 신호의 측과대 상에 전달된다. 예를 들어 실시예에서 가정에서의 관람자는 영화라이브러리를 액세스 하거나 소정 시간에 관람을 위해 오디오/비디오 콘텐츠를 액세스한다. 특히 방법이 누적된 영화를 허용하고 가정에서의 셋톱박스에서의 영화와 같은 전송미디어 에셋을 포함한다.

[0068] 셋톱박스 내에서의 하드디스크 드라이브는 영화를 저장하는데 이용된다. 상기 영화는 표준 방송 텔레비전 신호를 통해 다량의 데이터를 전달하는 데이터 캐스팅 기술을 이용하여 전송된다. 데이터캐스팅 기술에 관계되는 정보는 1999년 10월 28일 공개되고 1999년 4월 16일 출원된 Hartson 의 출원번호 W09955087 에서 알 수 있다. 예를 들어 24 시간 동안에 데이터캐스팅 기술은 20 고품질 특징을 갖는 영화를 발송할 수 있다. 셋톱박스는 VCR 또는 DVD 플레이어와 같은 다른 외부장치처럼 사용자의 텔레비전에 용이하게 연결될 수 있다. 구체적인 실

시에에서 셋톱박스는 개인 가정에 구비되고 표준비디오 케이블을 이용하여 텔레비전에 연결된다. 상기 셋톱박스는 방송텔레비전신호를 통해 데이터를 수신하는 안테나를 갖는다. 상기 셋톱박스 다수의 영화를 저장하기 위한 80 기가바이트의 하드 드라이브를 갖는다. 셋톱박스는 서비스 제공자를 주기적으로 접촉하기 위해 셋톱박스가 사용하는 모뎀을 갖는다. 정보는 서비스제공자와 셋톱박스 사이에 통과된다: 목적하는 바를 넣기 위해 이용되는 사용자"관람/렌탈이력"을 포함한다. 셋톱은 시스템 성능을 모니터링하는데 이용되도록 로그와 영화를 해독하는데 이용되는 "영화키"를 수행한다.

[0069] 셋톱박스는 방송신호, 공통데이터 및 쓰여진 데이터로부터 하드 드라이브에 까지 데이터 스트림을 수신할 수 있는 프로세서를 갖는다. 상기 프로세서는 원거리제어 및 모뎀 활성으로부터 적외선신호에 반응하고 영화를 플레이 할 수 있다. 상기 하드드라이브는 제거되고 이용되지 않는다면 보안 목적을 위해 셋톱박스와 결합된다. 상기 셋톱박스는 매우 조용하게 설계된 팬을 갖지 않는다. 상기 셋톱박스는 보안 시스템의 부분으로서 보안 프로세서를 포함한다. 상기 보안프로세서는 쉽게 변경되거나 위조되지 않게 예폭시로 PCB에 물리적으로 결합되며 스마트카드의 능동적인 구성요소이다. 영화는 리얼타임 내에서 셋톱박스에 "스트림" 되지 않고 콘텐츠파일은 "패킷화"되고 상기와 같은 "패킷" 들은 점차 유사하게 되어 셋톱박스에 연속적으로 전달된다. 사용자는 데이터가 셋톱박스로 보내지는 것을 알지 못하고 있다. 상기 영화는 유한시간 동안에 박스 내에서 수동적으로 구비된 제공자에 의해 제공된다. 데이터의 전송은 콘텐츠 또는 서비스 제공자에 의해 제어될 수 있다. 전체적으로 수신된 영화를 확보하기 위해서 동일한 영화가 몇회에 걸쳐 셋톱박스로 방송될 수 있다. 제1 전송시에 수신되지 않은 데이터 패킷은 이후 방송에서 수신될 수 있을 것이다. 본 발명의 상관관계에서 클라이언트프로세스는 셋톱박스에 구비하고 미디어 에셋은 예정된 제한(예., 네트워크, 대역폭, QOS)이 만족될 때 원거리 사이트로부터 얻어진다.

[0070] 셋톱박스에 전송되는 영화는 영화의 특성을 정의하는 관련된 정보를 가지고 있을 것이다. 예를 들어 영화는 영화가 관람될 수 있는 시간을 한정하는 시작과 종료와 관련된 날짜 또는 시간을 가지고 있을 수 있다. 예를 들어 영화는 셋톱박스 내에 저장되고 수신될 수 있으나 영화가 관련 날짜까지 관람되는 것을 허용하지 않는 관련 날짜를 가지고 있을 수 있다. 이것은 어느 한 위치에서 다른 위치까지 변화하는 영화를 위해 전송시간에 불일치를 허용하고 그리고 공식 릴리지 날짜에서 즉시 이용가능하고 "프리-로드"되는 영화를 허용한다. 유사하게 에셋리스트는 영화와 관련된 종료날짜를 가질 수 있고 영화가 더 이상 보여지지 않는 경우의 날짜 이후에 셋톱박스로부터 자동적으로 삭제된다.

[0071] 셋톱박스에 저장된 영화는 암호화된다. 영업규칙(사용자는 충분한 신용이 있어야 한다)의 만족과 영화관람 선택에 따라 셋톱박스는 복호화되고 플레이되는 영화를 허용한다. 상기 셋톱박스는 관람되는 영화를 허용하기 이전에 서비스 공급자에게 연결되는 필요하고 영화를 복호화하기 위한 키들은 일반적으로 프리패치되고 현재 계정 상태를 갖는 셋톱박스에 구비된다. 암호화된 영화는 복호화없이 관람될 수 없다. 모든 복호화는 로그되고 상기 로그는 사용자의 지불을 결정하는데 필요하다. 이러한 복호화 로직과 사용자 계정상태는 보안 프로세서에 의해 조정된다. 본 발명의 바람직한 실시예에서 비용은 관람, "사용확인" 대화창 및 영화의 관람 이전의 창의 선택에 따라 사용자에게 청구된다.

[0072] 또 다른 실시예에서 사용자는 영화의 일부분이 보여지면 영화관람 비용을 지불한다. 영화가 선택되었다면 영화는 제한된 시간에 대한 또는 제한된 관람횟수에 대한 비용없이 반복해서 보여질 수 있다. 바람직한 실시예에서 영화에 대한 비용지불 대신에 사용자는 월마다 지불한다. 월 마다의 비용은 관람될 수 있는 영화의 수, 선택된 영화가 관람동안 이용될 수 시간 및 선택된 영화의 관람 허용 횟수와 같은 다양한 옵션에 의존될 수 있다. 전화라인은 키가 사용자에게 의해 팔릴 수 있고 비용이 만들어 질 수 있는 정보로부터 전달하는 효과에 이용된다. 그러므로 사용자는 사용자의 셋톱박스에 저장된 영화의 라이브러리를 구비하지만 영화가 실제로 관람되지 않으면 비용이 청구되지 않는다.

[0073] 바람직한 실시예에서 데이터는 케이블, 위성, 인터넷 등에 의한 셋톱박스에 전송될 수 있다. 무선방송이 바람직한 실시예에 개시되지만 본 발명은 무선전송에 한정되지 않는다.

[0074] C. 실시예 설명

[0075] 이하, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자가 본 발명을 용이하게 실시할 수 있을 정도로 상세히 설명하기 위하여, 본 발명의 가장 바람직한 실시예를 첨부된 도면을 참조로 하여 상세히 설명하기로 한다. 본 발명의 목적, 작용, 효과를 포함하여 기타 다른 목적들, 특징점들, 그리고 작동상의 이점들이 바람직한 실시

예의 설명에 의해 보다 명확해질 것이다.

- [0076] 도1은 사용자장치에서 인터럽트없이 미디어 스트림을 프리젠틱하고 전달하기 위해 이용되는 시스템(1)의 개략도이다. 특히, 사용자장치(20)는 도면부호(40)로 도시된 바와 같이 콘텐츠제공자의 사이트에서 클라이언트 에셋리스트(42)와 교차하는 클라이언트 프로세스(24)을 포함한다. 제시된 예에서와 같이 사용자장치(20)에서 정보를 저장하기 위한 로컬캐시(26)가 구비된다. 상기 캐시(26)는 RAM, ROM, 하드드라이브 중에 적어도 하나가 될 수 있다. 또한 프로그램 콘텐츠(스포츠 이벤트, 쿠싱쇼)와 같은 에셋(62)을 포함하는 원거리 사이트(60)를 구비할 수 있다. 에셋(62)은 콘텐츠 제공자 사이트(40) 또는 케이블/인터넷 서비스 공급자에 구비될 수 있다. 상기 에셋은, 클라이언트가 얻어진 에셋을 시작하기 전에 케이블/ISP 제공자에서 이용될 수 있다. 또한 클라이언트는 물리적으로 분리된 다수의 위치로부터 동시에 에셋을 구할 수 있다(예., 케이블/ISP 제공자, 콘텐츠 제공자 등).
- [0077] 상기 클라이언트 프로세스(24)는 소비자에 대한 미디어 에셋(26)의 캐시를 관리한다. 실시예에서 캐시(26) 내의 에셋(62)은 데이터 경로(80)을 통해 콘텐츠 제공자(40)에 의해 제공되는 에셋리스트(42)에 의해 지시된다. 실시예에서 클라이언트 에셋 매니저 프로세스(24)는 데이터 경로(80)를 통해 네트워크 위치(콘텐츠 제공자 사이트(40)와 같은)로부터 특정 시간간격에서 에셋리스트(42)를 구한다. 특히 클라이언트 프로세스(24)는 데이터 경로(80)를 통해 업데이트된 에셋리스트(42)를 전달하는 콘텐츠 제공자(40)에 제어신호경로(84)를 통해 수요를 수행한다.
- [0078] 마찬가지로 다른 실시예에서 에셋리스트(42)는 주기적 방법으로 콘텐츠 제공자의 서버에 의해 소비자 클라이언트(24)에 보내질 수 있다. 상기 에셋리스트(42)는 콘텐츠에셋의 리스트를 포함하는 데이터 파일이다. 에셋리스트(42) 내에서의 각 요소는 일반적으로 네트워크 위치를 식별하고 에셋을 얻기 위해 필요로 되는 프로토콜을 식별한다. 다른 실시예에서 네트워크 위치는 URL이고 프로토콜은 HTTP 이다. 에셋리스트(42)는 각 에셋과 연관되는 다른 속성을 포함할 수 있는 것이다.
- [0079] (i) 만료일, (ii) 삭제
- [0080] (ii) 콜백 URL - 바람직한 실시예에서 상기 콜백 URL은 특별한 에셋이 관람을 위해 준비되는 E-mail 또는 인스턴트 메시지를 통해 사용자에게 공지하는 어플리케이션과 트래킹 어플리케이션(에셋이 다운로드되고 종종 트래킹을 허용)과 같은 에셋의 다운로드에 기초하여 동작을 초기화하는데 이용된다.
- [0081] (iii) 클라이언트 측 토큰 - 바람직한 실시예에서 에셋의 이용성에 기반하여 프리젠틱이션을 적용하는데 이용된다.
- [0082] (iv) 처리량 - 바람직한 실시예에서 이러한 숫자는 이용가능한 클라이언트 처리량의 퍼센트이고 100% 보다는 적다. 클라이언트는 종종 데이터의 조각을 얻는 것을 시도하여 다운로드를 천천히 할 수 있다. 바람직한 실시예에서 숫자는 매초 마다의 바이트 내에서의 다운로드 데이터 레이트이다.
- [0083] 엠바고 날짜 - 콘텐츠가 이용할 수 있는 한정된 윈도우가 될 수 있다. 콘텐츠에 도달된 엠바고 데이터는 보다 길게 이용할 수 있도록 클라이언트에 의해 제거된다,
- [0084] 한편 에셋은 콘텐츠의 번들로 조합될 수 있다. 이러한 속성은 번들로 할당될 수 있다. 예컨대 이것은 미디어 에셋의 세트가 사용자의 장치에 전달될 때 생성될 수 있는 클라이언트 측 토큰을 허용한다.
- [0085] 몇몇 속성이 전체 에셋리스트(42)와 결합될 수 있다.
- [0086] 각 속성은 이하와 같을 수 있다.
- [0087] (a) 다시읽기 레이트 : 바람직한 실시예에서 이것은 서버측에서 제어될 수 있고(예, 콘텐츠 제공자 사이트 40), 데이터 경로(80)를 통해 클라이언트 매니저에 에셋리스트(42)를 입력한다.
- [0088] (b) 리소스 경로, (c) 미디어 에셋
- [0089] 도7은 에셋리스트에서 에셋정보와 관련된 정보의 계층적 성질을 도시한 것이다. 도시된 바와 같이 콘텐츠 제공자/ISP에 의해 클라이언트에 제공되는 에셋리스트는 에셋정보에 대한 속성을 갖는다.
- [0090] 실시예에서와 같이 속성(1,2)은 에셋(1)의 만료일을 참조할 것이다. 도면에서 속성은 각각의 에셋에 근접한다.
- [0091] 바람직한 실시예에서 속성은 도8에 도시된 바와 같이 에셋리스트에 최적화된다. 이러한 경우에서 클라이언트는

에셋의 모든 에셋에 적합한 것 만큼 최적의 속성에 기반하여 에셋리스트 내에서 다운로드된 에셋을 관리한다.

- [0092] 클라이언트(24)는 원거리사이트(60)로부터 다운로드한 에셋의 많은 파라미터들을 관리하는 것을 가능하게 한다. 예를 들어 실시예에서와 같이 클라이언트 에셋 매니저(24)는 신호(86)를 통해 원거리 사이트(60)에 요청을 초기화하고 데이터 경로(90)을 통해 로컬 캐시(26)에 미디어 에셋(62)을 전달하도록 초기화한다. 상기 클라이언트 프로세스(24)는 예정된 제한의 세트가 만족되는지에 기반하여 원거리 사이트(60)에 데이터 전달을 위한 요청을 보낼 수 있다. 예를 들어 다운로드를 사용자장치(20)가 아이들(IDLE)일 때, 네트워크 대역폭이 일정 레벨 이하 일 때, 사용자 장치에 있어서의 CPU 및/또는 메모리가 성능 레벨 이하 일 때 발생할 수 있다.
- [0093] 다운로드 용량은 사용자에게 의해 온/오프 스위치될 수 있다. 사용자는, 장치가 소정 네트워크 사에 연결되어 있을 때 다운로드가 일어나는 것을 명확히 할 수 있다(예, 다이얼을 통한 LAN).
- [0094] 미디어 에셋(62)은 클라이언트 프로세스(24)로부터 요청을 초기화하는 것이 없이 네트워크 상태(예., 트래픽, 이용가능한 대역폭, 하루시각 등)에 의존하여 원거리 사이트(60)에 의해 드로틀 되거나 푸쉬될 수 있다.
- [0095] 에셋리스트(42)는 클라이언트의 로컬 스토어/캐시(26) 상의 콘텐츠를 관리하고 공표하는 콘텐츠 생산자를 위한 수단으로서 이용된다. 예를 들어 이러한 기능은 몇몇 지점에서 웹 페이지 또는 플래시-베이스 프리젠테이션의 프리젠테이션으로 통합되는 미디어에셋(62)을 공표하는 콘텐츠 생산자를 허용하는 것이다. 따라서 생산자는 말미에 공표하는 시스템으로 에셋과 연합될 수 있는 위치, 프리젠테이션 날짜 및 텍스트& 관련정보를 포함하는 에셋에 대한 정보를 입력한다. 이러한 정보는 공표되고 관련 에셋리스트로 구성된다. 생산자는 저대역폭 미디어 에셋을 입력하고 대응문자정보를 입력하며 상기 정보는 두개의 다른 버전의 페이지를 생산하기 위한 풍부한 미디어 에셋에 정보와 결합될 수 있다.
- [0096] 상기 클라이언트 에셋 매니저 프로세스(24)는 다운로드와 캐싱을 제공하도록 사용자장치(20)에서 동작할 수 있는 어플리케이션이다. 사용자는 클라이언트 매니저/프로세스를 얻기 위한 콘텐츠 제공자 웹사이트를 통해 가입하도록 할 수 있다. 바람직한 실시예에서와 같이 클라이언트 매니저는 자동으로 시작하도록 인스톨될 수 있고 후순위에서 동작될 수 있다.
- [0097] 도2는 클라이언트 에셋 매니저 프로세스(102)가 사용자장치에서 구동할 때 시스템(100)에 대한 정보의 흐름을 도시한 것이다. 도시된 실시예에서 클라이언트 에셋리스트 제공자(120)로부터 주기적으로 다운로드되고 파일 내에 리스트된 에셋(130)은 원거리 사이트/콘텐츠 제공자(140)로부터 다운로드되고 캐시된다. 사용자는, 클라이언트 매니저 프로세스(160)와 통신할 수 있는 스크립트(150)를 포함하여 에셋을 디스플레이 할 수 있는 웹페이지를 네비게이트 할 수 있다. 클라이언트 매니저 프로세스(102)는 액티브 엑스서버가 될 수 있고 프록시 오브젝트는 웹페이지(180)에서 구동될 수 있고 능동적인 제어는 클라이언트 매니저프로세스(102)의 일부분이 될 수 있다. 다른 관점에서 클라이언트 프로세스(160)는 웹페이지(180) 또는 클라이언트 매니저 프로세스(102) 부분에 구비될 수 있다.
- [0098] C(i). 설계도법
- [0099] 입력은 이하가 될 수 있다:(i) 활성의 클라이언트 매니저 프로세스(102)를 통해 원거리제어를 허용하는 XML 파일과 같은 구조파일; (ii) 클라이언트에 캐시되는 원거리 미디어 파일. 출력은 미디어파일로의 로컬 경로가 될 수 있다.
- [0100] 어떤 웹 페이지는 클라이언트(102)에 의해 캐시되는 에셋을 디스플레이한다. 클라이언트의 액티브 엑스 인터페이스를 이용하여 페이지는, 클라이언트 프로세스(102)가 인스톨된다면 볼 수 있는 자바스크립트 코드(150)를 포함하고 미디어 파일을 위한 로컬경로에 관한 정보에 대하여 클라이언트에 요청한다. 경로가 리턴된다면 스크립트는 파라미터를 받아들인 미디어 플레이어에 기반한 소정 웹에 경로정보를 통과 시킬 수 있다.
- [0101] 구조서버(120)는 클라이언트 구조파일(110)을 호스트하는 소정 웹서버가 될 수 있다. 상기 미디어 서버(140)는 미디어 파일 또는 에셋(130)을 호스트하는 소정 웹서버가 될 수 있다.
- [0102] C(ii). 실행흐름
- [0103] 상기 클라이언트 프로세스(102)는 a.cab 파일로 패키지화 될 수 있고 프로그램의 생성자를 식별하는 디지털 증명서로 서명된다(예., 월트디즈니 인터넷 그룹). 분리 .cab, xyzvideo.cab 는 서로 다른 콘텐츠 제공자로서 생성될 수 있다. 예를 들어 파일이름은 ESPN에 대해 espnvideo.cab 가 될 수 있고 ESPN 디지털 증명서로 서명될 수 있다. 인터넷 익스플로러는 어플리케이션을 다운로드하고 트랙하는 인터넷 컴포넌트 다운로드 특징을 이용한다. .INF 파일은 어떻게 클라이언트 프로세스(102)을 인스톨하는지에 관하여 컴포넌트 다운로드를 갖는 .cab

내에 포함될 수 있다.

- [0104] .cab 파일은 웹서버 내에 구비되고 브라우저는 적합한 액티브 엑스 컨트롤의 클래스 id를 포함하는 오브젝트(OBJECT) 태그와 같이 웹페이지를 로딩하고 코드베이스(CODEBASE) 속성에 의해 상세화된 URL을 이용하여 .cab 파일을 다운로드 할 것이다. 디지털 증명서는 우리의 제어가 다운로드되고 실행을 허용하는 프롬프트하는 것으로 사용자에게 알려주도록 디스플레이된다. 액셉트되었을 때 이러한 윈도우는 업데이트된 버전이 다운로드되지 않으면 다시 디스플레이되지 않을 것이다.
- [0105] 모듈, clientprocess.exe 는 다운로드된 프로그램 파일 폴더로 더해지고 클라이언트는 스타트업 폴더 내에 쏿킷을 생성할 것이다. 쏿킷의 이름은 파일을 만드는 것에 의존된다(예, 디즈니, ESPN 등). 상기 클라이언트 프로세스(102)는 클라이언트 프로세스가 인스톨된다면 참으로 세트되는 클라이언트 프로세스라 불리는 go.com 도메인 상의 쿠키를 생성한다. 상기 쿠키는 사용자에게 의해 일시적으로 삭제되는 것으로부터 방지하도록 규칙적으로 체크될 수 있다.
- [0106] 상기 프로그램은 사용자장치가 부트업되는 때에 구동하는 것으로부터 구동시스템에 의해 실행된다.
- [0107] 상술된 디지털 증명서는 대부분의 사용자가 인터넷으로부터 신뢰할 수 없는 어플리케이션의 인스톨을 허용하지 않을 만큼브라우저 보안세팅을 갖는다. 증명서는 소프트웨어의 출처를 식별하는 브라우저를 허용하고 회사 또는 개인이 참여되어야만 하는 것을 사용자에게 요청한다.
- [0108] 오브젝트 태그(OBJECT TAG)의 코드베이스(CODEBASE) 속성은 인스톨되는 소프트웨어의 바람직한 버전의 선택적인 부가를 지지한다. 상기 버전은 분리자로서 사용된 '#' 을 갖는 캡의 위치를 URL에 부가된다. 예를 들어 필드 내에서 클라이언트 프로세스의 현재 버전과 새로운 버전1,0,0,2 은 웹서버에 입력되고 코드베이스(CODEBASE) URL 의 말단부에 부가한 '#1,0,0,2'는 다운로드할 컴포넌트 다운로드를 관리하고 새로운 버전을 인스톨한다.
- [0109] .INF 파일은 업데이트가 리부트(REBOOT)를 필요로 하지 않도록 clientprocess.exe 에 /셋다운 파라미터를 갖는 clientprocess.exe 를 구동하도록 먼저 후크를 포함한다.
- [0110] 컴포넌트 인스톨러는 미리 정의된 키를 이용한 레지스트리에 정보를 부가할 것이다.
- [0111] 클라이언트 프로세스 시스템 트레이 아이콘에서 메뉴는 언인스톨옵션을 포함할 것이다. 클릭되었을 때 클라이언트 프로세스는 언레그서버 파라미터(/UNREGSERVER PARAMETER)를 통과하는 clientprocess.exe의 다른 예를 구동하여 언인스톨 될 것이다. 언인스톨프로세스는 (i) 컨트롤 언레지스터 (ii) 클라이언트 프로세스에 의해 더해진 레지스트리 엔트리 클린업 (iii) 스타트업 폴더로부터 클라이언트 공정 아이템을 제거한다.
- [0112] 구동시스템 시작과 익스플로러가 스타트업 폴더에서 아이템을 구동할 때 클라이언트 프로세스(102)가 시작할 것이다. 프로세스는 사용자가 시스템 트레이 아이콘 또는 구동시스템 셋다운으로부터 프로세스를 닫지 않으면 중지되는 않는다.
- [0113] 클라이언트 프로세스(102)는 몇몇 메뉴 아이템을 포함하는 시스템 트레이 아이콘을 제외하고 사용자에게 보이지 않는다. ESPN BottomLine 어플리케이션이 구동된다면 클라이언트 프로세스의 ESPN 버전은 BottomLine이 메뉴 내의 클라이언트 몇몇 옵션을 진행할 것이기 때문에 트레이아이콘을 디스플레이 하지 않을 것이다. 이것은 프로세스가 클라이언트 상에서 트레이 내에 구비될 있는 다수의 아이콘이 컷다운 되는 것과 같이 유용하다. BottomLine과 클라이언트 프로세스(102)와 같은 프로그램은 다른 하나가 셋다운된다면 또 다른 하나가 알려지는 것과 메뉴아이템 중의 하나가 선택된다면 클라이언트 프로세스가 알려지는 것을 확실하게 하기 위해 서로 주기적으로 통신되는 것이 필요할 수 있다. 상기 아이콘은 제작에 기초하여 ESPN 아이콘 또는 다른 아이콘이 될 수 있다.
- [0114] 이하의 아이템은 클라이언트 프로세스 트레이 아이콘 메뉴에 구비될 것이다:(i)도움말 - 클라이언트 프로세스(102)에 대한 도움말을 디스플레이함 (ii) XML 그룹파일은 선택적인 helpUrl 아이템을 갖을 수 있다. 사용자가 도움말을 선택한다면 브라우저는 URL 로 오픈될 수 있을 것이다 (iii) ~대하여 - 프로그램저작권/정보 대화창도시 (iv) 종료 - 프로그램정지(스타트업 폴더로부터 제거하지 않는다) (v) 언인스톨 - 프로그램을 언인스톨하고 COM 레지스트리 엔트리를 제거.
- [0115] 클라이언트 프로세스(102)의 적어도 하나의 예가 구동될 것이다. clientprocess.exe가 /언레그서버(UNREGSERVER) 또는 /셋다운 파라미터와 구동된다면 하나의 예는 이미 구동되는 것을 중단할 것이라는 예외가 있다. Clientprocess.exe 가 일부 프로그램에 의해 체크되는 분산된 뮤텍스(MUTEX)를 생성하여 단일 인스턴스

기능을 달성한다. 그러므로 몇몇 경우에 있어서 존재하는 인스턴스가 발견된다면 제2는 즉시 중단될 것이다.

- [0116] 클라이언트 프로세스에 대한 트래딩(THREADING) 모델은 아파트먼트 모델이 될 수 있다. 이것은 클라이언트 프로세스 액티브 엑스 컨트롤이 단일 트레드로부터 액세스되도록 설계되는 것을 의미한다. 그러나 전체로 클라이언트 프로세스는 멀티트레드되고 단일 에셋 다운로드를 수행하도록 트레드를 시작할 것이고 다음 다운로드가 일어날 때 트레드와 시작 트레드를 중지할 것이다. 다운로드에는 한번에 일어날 것이다.
- [0117] 규칙적인 동작모드 동안에 클라이언트 공정은 파라미터 없이 구동할 것이다. 그러나 이하의 커맨드 라인 파라미터가 지지될 수 있을 것이다 (i) /레그서버, (ii)/셋다운 - clientprocess.exe의 인스턴트를 구동하는 것을 정지한다, (iii) /언레그서버.
- [0118] 상기 클라이언트 프로세스는 구조서버로부터 다운로드되는 XML 그룹 파일을 이용하여 구성된다. 그룹파일의 URL은 웹페이지 상의 클라이언트 프로세스 그룹 액티브 엑스 컨트롤에 대한 오브젝트 태그 내에 xml PARAM 태그를 통해 얻어진다. 매회 클라이언트 프로세스 그룹 컨트롤은 xml 파라미터로 초기화되고 상기 클라이언트 프로세스는 그룹이 이미 추가되면 체크한다. 그렇지 않으면 프로세스는 레지스트리에 더해지고 파일은 다운로드되고 분석된다. 그룹파일에 대한 레지스트리 엔트리는 클라이언트 프로세스가 언인스톨될 때 까지 제거되지 않는다.
- [0119] 그룹파일은 설정가능한 리프레쉬 레이트에 따라 규칙적으로 다운로드된다. 서버가 If-Modified-Since header를 지지하면 클라이언트 프로세스가 마지막 다운로드 이래로 변경되지 않으면 클라이언트 프로세스는 파일을 다운로드하지 않을 것이다.
- [0120] 그룹파일 내에서 XML 은 몇몇 부가적인 클라이언트 프로세스 특정 요소로 RSS 포맷을 이용한다. RSS는 다수의 디즈니의 피드로서 용되는 배급을 위한 오픈 포맷이다. 아이디어는 그룹파일이 다른 시스템에 의해 이용될 수 있도록 가능한 표준만큼 이용하는 시도를 할 것이다.
- [0121] 도3은 그룹파일에 대한 실시예의 DTD를 도시한 것이다.
- [0122] 스타트업에 따라 클라이언트 프로세스는 이하의 액티브 엑스 컨트롤에 대한 COM 로컬 서버와 같이 시스템으로 레지스터한다. 클라이언트 프로세스 그룹은 초기화에 대하여 표시되고 클라이언트 프로세스 로케이터는 스크립팅에 대하여 표시된다. 이것은 고의적인 목적에 대한 제어의 이용을 방지하는 것이다. 클라이언트프로세스의 초기의 프로토타입은 상태를 표시하기 위한 웹페이지를 허용하는 각 주요한 요소에 대한 액티브 엑스 인터페이스를 갖도록 설계된다. 이러한 인터페이스는 발달/테스팅 현상 동안에 트러블슈팅에 대해 매우 유용한 것이다.
- [0123] ClientProcessSvc:
- [0124] 이것은 어플리케이션의 주요 클래스이고 다운로드와 어플리케이션의 인스톨을 가하는 오브젝트 태그를 갖는 인스톨 웹 페이지에 구비된다. 모든 에셋과 생성된 그룹 오브젝트를 트랙하는 공정에 대해 생성된 CClientProcessSvc의 단일 인스턴스가 있다. IClientProcessSvc 인터페이스는 프로그램의 상태를 리포팅하기 위한 방법을 제공한다.
- [0125] ClientProcessGroup:
- [0126] ClientProcessGroup Active X control 은 클라이언트 프로세스에 구조파일을 할당하는 자바스크립트 (JavaScript)에 의해 이용된다. IPropertyBag 인터페이스를 통하여 ClientProcessGroup xml 또는 소스라 불리는 오브젝트 태그 파라미터(OBJECT TAG PARAMETER)를 액셉트하고 파라미터는 XML 파일의 URL로 세트된다. 단일 XML 구조파일은 ClientProcessGroup 오브젝트에 의해 리프리젠티된다. IClientProcessGroup 인터페이스는 기능적 상태를 제공한다.
- [0127] ClientProcessLocator:
- [0128] ClientProcessLocator는 스크립팅을 위한 안전한 제어이고 GetLocalPath 방법을 갖는다. GetLocalPath 방법이 다운로드된다면 로컬파일의 경로를 결정하는 미디어 파일을 디스플레이하는 웹페이지 내의 자바스크립트에 의해 이용된다.
- [0129] 또한 사용자가 클라이언트 프로세스를 정지하고 미디어 페이지에 네비게이트하면 ClientProcessLocator 클래스 id 를 포함하는 오브젝트태그(OBJECT TAG)는 다시 실행할 clientprocess.exe 를 일으킬 것이다. 사용자가 제품을 언인스톨하면 이러한 태그는 코드베이스(CODEBASE) 파라미터가 특정되면 리인스톨 또는 업데이트를 가할 것이다.

- [0130] ClientProcessAsset :
- [0131] ClientProcessAsset(클라이언트 프로세스 에셋)은 부분적으로 캐시된 오브젝트를 나타낸다. 상기 COM 오브젝트는 ClientProcessSvc 내의 get_asset 또는 ClientProcessDownloader 내의 get_queuedAsset 에 의해 리턴된다.
- [0132] ClientProcessDownloader ;
- [0133] 단일 ClientProcessDownloader(클라이언트 프로세스 다운로더)는 어플리케이션이 시작할 때 생성된다. ClientProcessDownloader는 에셋의 다운로드 큐를 조정한다. 에셋은 큐다운로드(QueueDownload) 방법을 갖는 큐에 추가되고 에셋은 다운로드 큐의 상부에 가까워 질 때 에셋은 프로세싱을 위한 CAsyncCacheDownloader 인스턴스와 이격된다.
- [0134] ClientProcessReference :
- [0135] ClientProcessReference 오브젝트는 미디어파일을 리프리젠티한다. 각 그룹은 하나 또는 그이상의 리퍼런스를 갖는다. 이러한 COM 오브젝트는 ClientProcessGroup 에 있는 get_reference method에 의해 리턴된다.
- [0136] 다른 C++ 클래스
- [0137] 이 부분은 클라이언트 프로세스 어플리케이션 내의 실시예의 C++ 클래스를 정의하고클래스의 활성을 요약하고 서로 교차한다. 클래스이 몇몇은 다른 어플리케이션과 통신하기 위한 COM 래퍼(WRAPPER) 오브젝트를 갖는다. 도4는 다양한 클래스의 테이블을 제공한다.
- [0138] CAsyncCacheDownloader - asynccachedownloader.h/.cpp:
- [0139] 상기 클래스는 HTTP 를 통해 로컬 장치에 다운로드한 에셋을 캡슐화한다. CAsyncCacheDownloader 는 대부분 동작이 분리트레드에서 일어나기 때문에 메시지를 수신하는 것을 허용한 CWindowImpl을 세분한다. 상기 클라이언트 프로세스는 인터-트레드(INTER-THREAD) 커뮤니케이션에 대한 윈도우 메시지를 이용한다.
- [0140] 대부분의 방법은 메인 트레드 상에서 구동되지만, StartDownload()는 새로운 트레드를 생성하고 DoDownload() 방법을 호출한다. DoDownload()는 단이 에셋을 다운로드하는 WinInetAIOs를 이용하고 인터넷 익스플로러 캐시 엔트리를 생성한다.
- [0141] DoDownload 방법은 한번에 작은 조각에서 파일을 다운로드 할 것이다. 상기방법은 처리량이 100% 이하라면 각 조각 후에 슬리프(sleep) 될 것이다. 예를 들어 처리량 50% 이라면 다운로드 트레드는 평균 50% 처리량을 달성 하는 조각을 다운로드하는 동일한 시간 동안 슬리프할 것이다.
- [0142] CCacheAsset - cacheasset.h/.cpp
- [0143] CCacheAsset 인스턴트는 로컬시스템에서 캐시되는데 필요한 단일 파일을 리프리젠티한다. 이것은 그룹파일 또는 미디어파일이 될 수 있을 것이다. CCacheGroup 과 CCacheReference 인스턴스는 파일의 다운로드와 캐싱을 조절하는 CCacheAsset 인스턴스를 예로 든다. GetInterface 방법은 각각의 COM 래퍼(WRAPPER) 오브젝트의 ClientProcessAsset 인터페이스 포인터를 리턴한다.
- [0144] 다운로드가 성공적이지 않다면 CCacheAsset 는 도5의 표에 따라 다시 시도하는 타이머를 세트할 것이다. 다운로드가 성공적이라면 CCacheAsset 은 캐시 아이템이 캐시로부터 삭제되지 않는 것을 확실하게 하는 파일시스템을 기다린다. CCacheAsset 이 있다면 아이템은 다시 다운로드된다.
- [0145] CCacheGroup - cachegroup.h/.cpp:
- [0146] 상기 클래스는 XML 그룹 파일을 캡슐한다. 각 그룹 XML 파일은 CCacheGroup를 줄이고 각 미디어 파일 아이템에 대한 CCacheReference 인스턴스를 생성하는 CCacheGroup의 인스턴스로 통과된다. 상기 GetInterface 방법은 각각의 COM 래퍼(WRAPPER) 오브젝트의 ClientProcessGroup 인터페이스 포인터를 리턴한다. CCacheAsset 인스턴스는 그룹파일을 리프리젠티하도록 생성되고 타이머는 XML 내의 상세화된 리프레시 레이트에 간격세트로 세트된다. 타이머가 오프되면 그룹 파일에 대한 CCacheAsset 은 다운로드큐로 추가된다.
- [0147] CCacheReference - cachereference.h/.cp:
- [0148] XML 그룹 파일은 CCacheGroup에 의해 줄여지기 때문에 CCacheReference 인스턴스는 각 미디어 파일 아이템에 대해 생성된다. CCacheReference 는 다운로드/캐시 기능을 조정하는 CCacheAsset 인스턴스를 생성한다. 이러한 클래스는 만료된 디스크로부터 아이템 만료시간을 체크와 아이템들을 제거의 원인이 된다.

- [0149] CCacheTime - cachetime.h/.cpp:
- [0150] CCacheTime은 시변환 기능을 제공하는 ATL CTime 클래스를 세분한다. 클라이언트 공정에서의 모든 날짜/시간 멤버 변수가 CCacheTime을 이용하여 저장된다.
- [0151] CClientProcessModule - clientprocess.cpp:
- [0152] 상기 클래스는 실행가능한 프로세스를 리프리젠티 한다. CClientProcessModule은 표준 ATL CAtIExeModule T class를 세분한다. 이러한 이유는 레지스터 및 언레지스터 뿐만 아니라 프로세스의 스타트업에 기능을 추가하는 것이다. 스타트업에서 분할된 뮤텍스는 이름으로서 ClientProcessSvcGUID 를 이용해서 생성된다. 뮤텍스가 이미 존재한다면 어플리케이션은 이미 구동하고 따라서 현재 인스턴스가 종료된다. 모든 인스톨과 언인스톨 기능은 상기 클래스에서 이루어진다.
- [0153] CException - exception.h/.cpp:
- [0154] CException의 인스턴스는 에러가 발생할 때 생성되고 던져진다.
- [0155] CException가 오차의 형태를 조절한다.
- [0156] CRegistryVirtualDeviceX- registryvirtualdevicex.h:
- [0157] ATL 내의 CRegistryVirtualDevice 의 세분. 이러한 클래스는 우리가 레지스트리 내에서 InProcServer32 또는 LocalServer32를 입력할지를 결정하는 모듈의 확장을 체크해서 ATL 를 정지하는 AddStandardReplacement 를 오버로드한다. 클라이언트 프로세스는 항상 .exe 이고 LocalServer 호출이 불필요하게 되고 어플리케이션 상에서 외부요구사항을 추가한다.
- [0158] CTimer 및 CTimeable - timer.h/.cpp:
- [0159] CTimer 는 타이머 기능을 캡슐화한다. 소정 오브젝트는 CTimeable, 수행 OnTimer, CTimer의 인스턴스 생성 및 시작방법의 호출로부터 상속하도록 필요한 타이머를 이용하는 것이 바람직하다. Stop()는 타이머를 멈춘다.
- [0160] CUrlMap - urlmap.h/.cpp:
- [0161] CUrlMap 은 맵이 URL에 오브젝트하는 해시(HASH) 맵 클래스를 생성하는 것을 허용하는 CAtIMap 템플레이트 클래스를 상세화한다.
- [0162] 더욱 상세하게는, 시스템은 이하에 의해 보다 확장된다: (i) 다운로드 코드에 다운로드 서포트를 추가, (ii) 만료되었을 때 예셋 삭제, (iii) ClientProcessLocator control을 수행하고 쿠키 로직을 제거,(iv)시스템 트레이 아이콘 및 메뉴 이용, (v) HitCountURL 서포트 제공, (vii)프로그램 정보제공
- [0163] 관리는 안전하게 이루어진다: (i) 적합한 액티브 X 안전세팅은 각 제어에 적용된다, (ii) COM 방법과 가치는 알려져 있지 않다(상기 방법과 가치는 비주얼 베이직과 같은 사용자 인터페이스를 제어설계하면서 보여지지 않는 방법이다), (iii) 모든 타이머가 이용되도록 파일시스템을 폴(POLL)하는데 이용된다, (iv) BottomLine 의 로직은 ClientProcess를 검출하고 시스템 트레이 아이콘 서브메뉴를 제공하는데 추가된다, (v) 스타트업 메뉴 슛킷트는 레지스트리 엔트리를 구동하는 윈도우 대신에 이용된다, (vi) 규칙적으로 ClientProcess 가 필요할 때 인스톨되거나 재생성되는지를 지시하는 쿠키를 체크한다.
- [0164] C(iii). BottomLine 개선
- [0165] BottomLine(ESP의 상표)이 존재한다면 클라이언트 프로세스를 검출하여 변화되는 것과 시스템 트레이 메뉴에서 클라이언트 프로세스 아이템을 디스플레이 하는 것을 변경하는 것이 필요할 것이다. 클라이언트 프로세스는 CLIENTPROCESS_MENU_START 및 CLIENTPROCESS_MENU_END 범위 내의 메뉴아이템 아이디(ID)를 정의한다. 이러한 상수들은 헤더가 만들어 질 때 BottomLine에 받아들여 질 수 있는 헤더파일 내에서 정의된다.
- [0166] BottomLine과 클라이언트 프로세스는 BottomLine과 클라이언트 프로세스가 이하의 단계들을 수행하는 것이 필요하도록 각 프로세스가 구동하는지를 인식하는 것이 필요하다:
- [0167] (a) BottomLine이 스타트업 할 때 BottomLine은 클라이언트 공정이 구동되면 볼 수 있는 클라이언트 프로세스 뮤텍스(MUTEX)를 생성하는 것이 필요하다,(b) 클라이언트 프로세스가 스타트업 하거나 닫혀질 때 클라이언트 프로세스는 BottomLine 이 보고 BottomLine 이 승인에 반응하는 레지스터된 윈도우 메시지를 내보낸다.
- [0168] 레지스터된 윈도우 메시지는 동일한 메시지를 레지스트하는 다른 윈도우와 통신하기 위한 유일한 메시지를 정의

하는 어플리케이션이 허용된다. 클라이언트 프로세스 또는 BottomLine이 어플리케이션 또는 승인에 고지할 때 클라이언트 프로세스 또는 BottomLine 미래 통신에 있어서 윈도우 핸들을 따라 통과될 것이다. 클라이언트 프로세스는 메뉴에서 디스플레이되는 정보를 BottomLine에 전송하는 WML_COPYDATA를 이용할 것이다. 클라이언트 프로세스 메뉴 아이템이 선택될 때 BottomLine은 클라이언트 프로세스 메뉴 아이템 앞에 있을 것이다.

- [0169] 몇몇 특징은 클라이언트 프로세스를 수행하는 시스템으로 포함된다: (i) 인스톨레이션 및 업데이트의 용이성, (ii) 클라이언트측 사용자 인터페이스가 적거나 없음, (iii) 클라이언트 프로그램의 서버측 구조, (iii) 파일이 다운로드될 때에 제어 오버 레이트, (v) 자바스크립트를 통해 다운로드된 미디어 파일의 상태를 결정하는 능력, (vi) 콘텐츠 제공자 스펙이 되는 제어버전을 이용하는 능력.
- [0170] C(iv). 시스템 특징
- [0171] 클라이언트 프로세스 프로그램:
- [0172] 클라이언트 프로세스 프로그램은 프로그램이 웹 페이지 상에 디스플레이되기 전에 미디어 콘텐츠 시간을 다운로드 하기 위한 원인이 된다. 사용자가 플레이백(PLAYBACK) 품질이 현저하게 개선된 파일을 디스플레이 하기 위한 HTML을 포함하는 페이지를 내비게이션 할 때 파일의 로컬 카피가 이용된다.
- [0173] 인스톨레이션:
- [0174] 클라이언트 프로그램은 자동적으로 인스톨하도록 허용될 것이다. 소정 웹 페이지는 필요하다면 브라우저가 자동으로 인스톨하거나 업데이트하는 태크를 포함할 수 있다.
- [0175] 사용자 인터페이스:
- [0176] 클라이언트 프로그램에서 사용자 인터페이스는 턴오프하거나 프로덕트를 언인스톨하는 능력을 제공하는 메뉴를 갖는 시스템 트레이 아이콘과 통합된다. BottomLine 어플리케이션이 존재한다면 BottomLine의 시스템 트레이 메뉴가 대신에 이용될 수 있을 것이다.
- [0177] 실행:
- [0178] 클라이언트 프로세스는 구동될 수 있도록 사용자장치의 스타트업 폴더 내에 인스톨된다. 상기 프로그램은 에셋들이 완료하고 에셋을 다운로드하는 레이트에서 규칙적으로 다운로드되는 에셋에 대한 정보를 포함하는 WDIG 서버로부터 XML 파일을 다운로드할 것이다. 로컬 미디어 파일의 위치를 결정하기 위한 클라이언트 프로세스와 통신하는 콘텐츠 제공자 웹사이트에 대해 인터페이스 될 수 있을 것이다.
- [0179] 호스팅:
- [0180] 인스톨레이션 .cab 파일은 이러한 제품을 이용하여 수직으로 제공된 하나 또는 그 이상의 헤드-엔드에 의해 호스트 될 것이다. 클라이언트 프로세스의 새로운 버전이 릴리즈 됨에 따라 이러한 파일은 서버상에서 업데이트 될 것이다.
- [0181] 콘텐츠 제공자 BottomLine 변화:
- [0182] 적합한 변화가 클라이언트에 대한 시스템 트레이 인터페이스를 제공하도록 BottomLine 어플리케이션으로 될 것이다.
- [0183] 샘플 HTML 및 스크립트:
- [0184] 샘플 HTML과 자바스크립트는 어떻게 호스트하고 클라이언트 프로세스와 통신하는지를 보여주는 디벨로퍼로 제공 될 수 있다.
- [0185] 외부인터페이스 필요사항
- [0186] 상기 부분은 인터페이스가 생성되거나 변화될 필요가 있도록 외부시스템을 묘사한다.
- [0187] ESPN BottomLine:
- [0188] 몇몇 약간의 변화는 클라이언트 프로세스와 클라이언트를 정지하고 언인스톨하기 위한 부가메뉴 아이템과 클라이언트 존재를 검출하는 BottomLine 으로 만들어진다.
- [0189] 클라이언트 프로세스 검출:
- [0190] 각 시간마다 BottomLine 메뉴가 디스플레이되고 BottomLine이 클라이언트 프로세스의 존재를 검출해야하고, 메

뉴가 존재하거나 구동하면 클라이언트 프로세스 메뉴 아이템을 포함하는 서브메뉴를 디스플레이한다. 클라이언트 프로세스의 검출은 클라이언트 프로세스가 스타트업에 의존하여 생성하는 존재를 체크하거나 클라이언트 프로세스 윈도우 클래스를 갖는 윈도우를 조사하여 용이하게 이루어진다.

- [0191] 클라이언트 프로세스 커뮤니케이션:
- [0192] 클라이언트 프로세스가 구동하면 BottomLine이 디스플레이하는 메뉴아이템을 질의하는 윈도우 메시지를 이용할 수 있다. 클라이언트 프로세스 메뉴 아이템이 선택될 때 BottomLine은 아이템이 선택되는 것에 관한 클라이언트 프로세스를 언급하는 메시지를 이용할 것이다.
- [0193] 미디어 콘텐츠 서버:
- [0194] 미디어 파일을 호스트 하는 서버는 다운로드 드로틀링을 가능하도록 다운로드하는 HTTP 바이트 범위를 유지하는 HTTP1.1 웹서버로 될 것이다. 서버가 바이트 범위를 유지하지 않으면 전체 파일은 구조적 세팅에도 불구하고 한번에 다운로드 될 수 있다.
- [0195] 구조서버 인터페이스
- [0196] XML 구조 파일을 호스트하는 서버가 될 수 있다. XML은 스펙을 설계해서 공표되는 형태가 될 것이다.
- [0197] 기능적 요구사항
- [0198] 1. 프로그램을 지시하는 사용자의 장치에서 쿠키의 세팅이 인스톨된다. 이러한 쿠키는 사용자가 제거하지 않는 것을 명확히 하도록 주기적으로 체크된다.
- [0199] 2. 인터넷 익스플로러의 컴포넌트 다운로드 장치를 통해 업데이트하는 지지소프트웨어는, 새로운 버전이 구비되고 HTML 페이지가 최근 버전을 요청하여 변화된다면 클라이언트코드가 적절하게 업데이트되는 것을 확실히 할 것이다. 또한 재부팅이 리프레스가 필요로 하지 않은 것이 확실히 될 것이다.
- [0200] 3. 프로그램은 사용자가 멈추지 않으면 항상 구동될 것이고 스타트업 폴더 내에 인스톨 될 것이다.
- [0201] 4. 프로그램의 인스턴스는 한번에 구동될 것이다.
- [0202] 5. 언인스톨이 시스템 트레이 아이콘 메뉴를 통해 제공될 것이다. 언인스톨은 초기화된 아웃사이드 프로그램으로부터 될 것이다.
- [0203] 6. 메뉴아이템은 시스템 트레이 아이콘으로 제공된다.
- [0204] 7. 도움말 - 디즈니 캐시로서 도움말을 디스플레이한다.
- [0205] 8. 구조파일은 선택적인 도움말 URL 아이템을 갖을 것이다. 사용자는 도움말을 선택할 것이고 브라우저는 이러한 URL과 오픈될 것이다.
- [0206] 9.~ 대하여 - 프로그램 저작권/정보 다이얼로그를 도시한다.
- [0207] 10. 종료 - 프로그램을 정지한다(스타트업 폴더로부터 제거하지 않는다)
- [0208] 11. 언인스톨 - 프로그램을 언인스톨하고 레지스트리 전체를 제거한다.
- [0209] 12. 구조파일:(i) XML 포맷이 될 것이다, (ii) 얼마나 자주 구조파일을 다운로드하는지를 결정하기 위한 리프레스 레이트를 제공할 것이다, (iii) URL을 위한 퍼-아이템(PER-ITEM), 만료일, 히트 트래킹을 위한 URL 및 다운로드 레이트를 제공할 것이다.
- [0210] 파일 다운로드
- [0211] 브라우저(예., 인터넷 익스플로러) 캐시:
- [0212] 클라이언트 프로세스는 미디어 파일의 클라이언트 측 캐싱을 제공하기 위해 인터넷 익스플로러 캐시를 이용한다. 캐시내에 저장된 파일들은 필요하다면 다른 아이템들을 위한 공간을 만들기 위해 인터넷 익스플로러에 의해 클린업 될 수 있다. 그러나 XML 구조파일은 소정 주기(예., 30일) 동안에 하드 드라이브상에 남겨지도록 고정으로 마크될 수 있다. 인터넷 익스플로러 캐시가 윈도우 로그인 기초에 있기 때문에 각 사용자는 분리 캐시를 갖도록 허용된다.

- [0213] 클라이언트 공정은, 아이템이 캐시로부터 제거될 때 주기적으로 검출하기 위해 체크가 필요하다. 아직 완료되지 않았다면 다시 다운로드 될 수 있을 것이다.
- [0214] 아이템 존속기한
- [0215] 각 아이템은 만료일과 시간을 갖을 것이다. 주기적으로 클라이언트 프로세스는 만료일과 같은 시간을 체크할 것이고 만료된 캐시로부터 아이템을 삭제할 것이다. 클라이언트 프로세스는, 클라이언트가 에셋을 만료하는 기회를 갖기 전에 에셋이 XML 구조로부터 제거되면 장치상에 존재하는 에셋을 트랙하는 것이 필요하고 에셋은 사용자의 하드 드라이브로부터 삭제될 수 있을 것이다(이것은 사용자가 연장된 주기 동안 장치오프(DEVICE OFF)를 갖으면 발생할 수 있을 것이다).
- [0216] 아이템이 다운로드되었을 때 인터넷 익스플로러가 캐시로부터 아이템을 제거하지 않으면 다시 다운로드 되지 않을 것이다.
- [0217] 구조 리프레시:
- [0218] 클라이언트는 리프레시 레이트 XML 요소에 의해 상세화된 레이트에서 서버로부터 다운로드하여 XML 구조 파일을 리프레시한다.
- [0219] 다운로드 에러:
- [0220] 아이템이 다운로드 될 때 에러가 발생한다면 타임아웃 된 후에 클라이언트 프로세스는 다시 시도할 것이다. 기다린 시간은 도6에 도시된 차트에 따라 에러 형태에 의존한다.
- [0221] 또한 클라이언트 프로세스는 네트워크 연결이 이용가능하다면 다운로드를 수행할 것이다. 필요하다면 프로세스는 다운로드 한 후에 다이얼 업을 수행하지 않을 것이다.
- [0222] 드로틀링:
- [0223] 구조파일에서 각 아이템은 선택적으로 소정 다운로드 양을 갖는 구조가 될 수 있다. 이것은 클라이언트 프로세스가 피스(PIECE) 내에서 파일을 다운로드하고 이러한 것을 달 성하는 각 피스들 사이에서 중지를 하는 응답이다. 파일을 호스트하는 웹서버는 HTTP 바이트 범위 특징을 유지할 수 있다.
- [0224] 웹 인터페이스
- [0225] 구조파일에 대한 URL은 액티브 엑스 컨트롤 프로퍼티(PROPERTY)를 통해 클라이언트 프로세스로 통과된다. 액티브 엑스 컨트롤은 스크립팅과 초기화로 인해 안전하게 될 것이다. 상기 클라이언트 프로세스는 go.com 도메인으로부터 변환된 XML 구조 URL의 서버부분을 체크할 수 있다. 이것은 잠재적으로 위험한 데이터를 갖는 클라이언트 프로세스를 초기화하는 것으로부터 적절하지 않은 웹페이지를 방지한다(큰 파일들은 작은 리플레시 레이트로부터 종종 다운로드된다). 오브젝트 태그는 클라이언트 프로세스가 적절하지 않은 누군가를 위해 인스톨되지 않도록 코드베이스(CODEBASE) 파라미터를 갖는 것이 필요하지 않다. 자바스크립트는 인스톨된 쿠키를 체크하고 호출하는 API 를 만들기 전에 오브젝트가 존재하는지를 체크하도록 코드를 갖을 것이다. 적어도 클라이언트 프로세스의 버전은 클라이언트 장치에 인스톨 될 수 있다.
- [0226] 삭제
- [0227] D. 개요
- [0228] 본 발명의 바람직한 실시예와 실시예의 설명은 설명과 묘사를 목적으로 기술되고 있다. 실시예들은 본 발명을 한정하거나 제한하는 것이 아니다.
- [0229] 실시예들은 본 발명의 기술적 사상을 벗어나지 않는 범위내에서 다양한 변화와 부가 및 변경이 가능하다.
- [0230] 예를 들면 이하와 같이 될 수 있다.
- [0231] (a) 인스톨된 클라이언트 APP 들은 부가 메뉴 아이템에 레지스터 할 수 있으며 일반 시스템 트레이 아이콘유지를 제공하는 새로운 컴포넌트를 생성하는 것이 가능하게 될 수 있다.
- [0232] (b) 서버측 사용자는 클라이언트의 활성을 결정하도록 세팅한다. 클라이언트 프로세스는 인스톨하는 사용자에게

게 속해있는 유일한 식별자를 필요로 한다. 이 때 서버는 더욱 복잡하게 될 수 있고 다른 사용자들을 위해 서로 다른 XML 구조 파일을 전송한다.

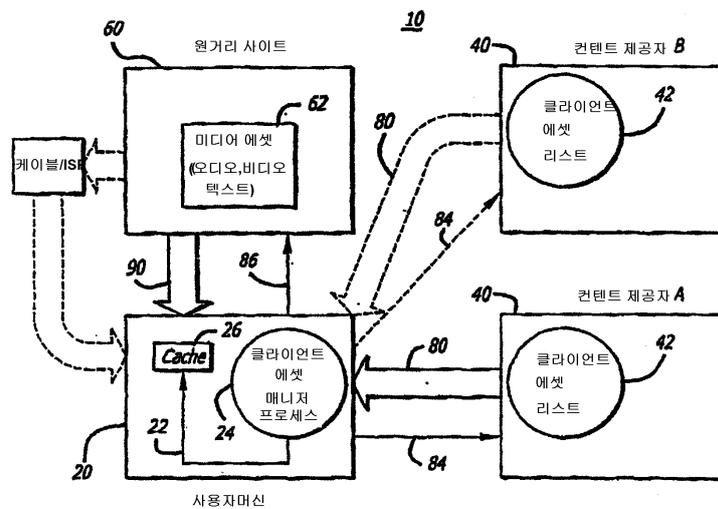
- [0233] (c) 연속적인 다운로드가 완전하지 않다.
- [0234] (d) 더욱 복잡한 다운로드 드로틀링을 부가한다(예를 들어 베이스라인 클라이언트 대역폭 측정을 취하고 처리량의 제어를 계산하기 위해 이용)
- [0235] (e) MRTG 데이터에 기반하는 서버부하를 결정하기 위한 서버측 어려움과 리턴(503) 하도록 결정하는 이용은 에셋 다운로드 시에 무거운 부하를 방지하는 클라이언트 프로세스에 응답한다.
- [0236] 여기에서 개시되는 실시예는 여러가지 실시 가능한 예 중에서 당업자의 이해를 돕기 위하여 가장 바람직한 실시예를 선정하여 제시한 것일 뿐, 본 발명의 기술적 사상이 반드시 이 실시예에만 의해서 한정되거나 제한되는 것은 아니고, 본 발명의 기술적 사상을 벗어나지 않는 범위내에서 다양한 변화와 부가 및 변경이 가능함은 물론, 균등한 타의 실시예가 가능함을 밝혀 둔다.

도면의 간단한 설명

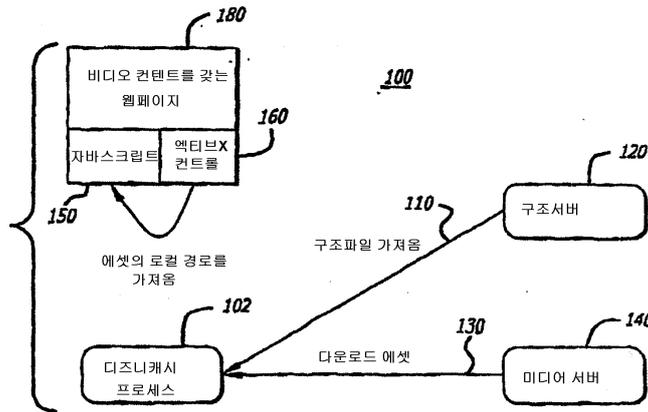
- [0037] 도1은 본 발명에 의한 바람직한 실시예로서 클라이언트 프로세스의 개략도를 나타낸 것으로 콘텐츠제공자 사이트와 원거리 미디어 서버와 교차하고 있는 것을 도시한 것이고,
- [0038] 도2는 클라이언트 프로세스의 다른 도면으로서 웹브라우저로 통합스트리밍과 캐시된 미디어로 이용되는 스크립트를 이용하는 것을 도시한 것이고,
- [0039] 도3은 파일을 위한 프로그램코드의 실시예를 도시한 것이고,
- [0040] 도4는 클라이언트 프로세스내에서 C++ 클래스의 실시예를 도시한 것이고,
- [0041] 도5는 클라이언트에 의한 미디어 에셋을 다운로드하기 위해 이용되는 타이밍의 특징의 실시예를 도시한 것이고,
- [0042] 도6은 미디어에셋 다운로드를 위해 기다림 시간의 양에 영향을 주는 오차차트를 도시한 것이고,
- [0043] 도7은 에셋리스트 내의 에셋정보에 관계되는 계층적 속성의 일면을 도시한 것이고,
- [0044] 도8은 에셋리스트 내의 에셋정보에 관계되는 계층적 속성의 다른 일면을 도시한 것이다.

도면

도면1



도면2



도면3

```

<|ELEMENT channel (item+ | refreshRate | helpUrl? | cookieDomain? |
cookieName?)>
<|ELEMENT item (link | expires | hitCountUrl? | throughput? | trackWithCookie?
)*>

```

The following is an example XML group file:

```

<?xml version="1.0" ?>
<channel>
  <item>
    <link>http://myserver/myfile.wmv</link>
    <expires>2002-22-03T00:17:15z</expires>
    <hitCountUrl>http://hitserver/hiturl</hitCountUrl>
    <throughput>75</throughput>
    <trackWithCookie/>
  </item>
  <refreshRate>60</refreshRate>
  <helpUrl>http://myserver/help.html</helpURL>
  <cookieName>adAssets</cookieName>
  <cookieDomain>.go.com</cookieDomain>
</channel>

```

도면4

Internal Class Name	COM/ActiveX Class Name
CCacheAsset	CDisCacheAsset
CCacheGroup	CDisCacheGroup
CCacheReference	CDisCacheReference

도면5

Error Type	Retry Attempted	Timeout Before Retry
Invalid item URL	no	N/A
Server Connection	yes	5 minutes
Server not found	yes	5 minutes
File not on server	yes	15 minutes
Unknown server error	yes	15 minutes
Internal Error	yes	1 hour

도면6

Error Type	Retry Attempted	Timeout Before Retry
Invalid item URL	no	N/A
Server Connection	yes	5 minutes
Server not found	yes	1 hour
File not on server	yes	1 hour
Unknown server error	yes	15 minutes
Internal Error	yes	1 hour

도면7

Asset List

Asset 1: {Attribute(1,1); Attribute (1,2);...;Attribute (1,M)}
Asset 2: {Attribute(2,1); Attribute (2,2);...;Attribute (2,M)}
⋮
Asset N: {Attribute(N,1); Attribute (N,2);...;Attribute (N,M)}

도면8

Asset List {Attribute 1; Attribute 2;...;Attribue N}

Asset 1
Asset 2
⋮
Asset N