다양한 무선단말을 위한 다운로더블 디코더 시스템 구현

김진렬° 서덕영
경희대학교 전자공학과
jinreol@empal.com°, suh@khu.ac.kr

Implementation Downloadable Decoder System For Variable Mobile Terminal

Jin Reol Kim° Doug Young Suh

요 약

다양한 무선단말에서는 멀티미디어 서비스를 적용하기 위해서는 데스크탑과 달리 다양한 시스템에 따라서 다양한 컨텐츠에 대한 디코더가 필요하게 된다. 디코더만 보더라도 표준화에 따라 여러 종류에 따라 디코더들이 있고, 각 업체따라 독립적인 디코더들이 있다. 이런 디코더들을 모두 무선단말을 처리하기 위해선 시스템의 너무 커지게 된다. 특히 표준화에 정의하는 디코더는 여러 가지 기능을 갖추기 때문에, 가 디코더 크기가 커지기 때문에, 시스템 역시 따라서 커지게 된다.

본 논문에선 위에 문제점들을 해결하기 위해 각 컨텐츠에 대해 특성화하여 크기가 작은 디코더를 개발하여 디코더를 컨텐츠와 함께 전송하는 효율적인 시스템을 제안하고자 한다.

1. 서 론

현재 무선단말은 멀티미디어에 대한 시스템에 대해 각각 다른 디코더를 사용하고 있다. 이에 따라 시스템에 제각기 다른 시스템에 맞도록 설계해야 하기 한다. 이에 따라 생기는 개발비용 커져가게 된다. 이런 문제에도 불과하고 시스템에 독립적인 자바는 느리다는 관념 때문에, Java 쪽에 멀티미디어 개발이 미흡했다. 하지만, 자바 VM이 나날이 발전되어 갔고 현재는 C에 비해 차이가 날 정도로 많이 느리지 않게 되었다. Java World에서의 Mark Roulo의 논문[1]을 보면, 현재 자바는 C에 비교했을 때 많은 차이가 나타나지 않는다는 것을 알 수 있다.

아래의 표들은 실제로 본 논문을 위해 테스트 한 결과이다.

CPU : Pentimum-4 2.0GHz
RAM : 256Mbyte
C Complier : Visual Studio 6.0
Java VM : Java Standard Edition 1.4.0_03

표1. 테스트 환경

	C			Java		
	10^9회	10^{10}회	10^{11}회	10^9회	10^{10}회	10^{11}회
int + int	3	34	343	2	25	254
int * int	2	20	180	17	171	1714
float+float	15	147	1534	15	151	1528
float*float	15	154	1538	15	155	1548
int>>int	4	43	442	7	67	677
(int)float	21	224	2092	13	144	1471
(float)int	3	31	314	3	28	284
*method_calls	43	424	4243	3	28	277
**object creation				21	227	2178

*method calls : int add(int, int);

**object creation : Object obj = new Object (Java에서 Object class 호출)

표2. C와 Java의 연산속도 비교(단위 : 초)

	C			Java		
	10^4회	10^5회	10^6회	10^4회	10^5회	10^6회
table(o)	1	5	55	1	9	86
table(x)	12	123	1230	18	181	1812
* IDFT				0	0	2

*IDFT[2] - 여기서 제시하는 방법을 이용하면 상당히 덧셈 연산과 곱셈 연산이 줄어 들게 된다.

표3. DCT 연산 속도 비교

위의 표에서 나타난 결과에서 말하듯이 현재의 자바는 C에 대해 느리지 않다. 위의 결과를 바탕으로 자바의 특성에 맞도록 Decoder가 제작된다면, 연산속도가 그렇게 느리지 않다.

역시 초창기 인베디드 시스템의 목적으로 만들어졌기 때문에 다른 언어보다 실행 파일 사이즈가 작다. Kim의[3] 논문에서 보았을 때, 본문에서 제시한 자바 디코더의 크기는 60kbyte다. MPEG-1의 컨텐츠가 5분 분량에 20Mbyte라도 보았을 때 상당히 작은 크기이다.

이를 바탕으로 디코더를 단말로 전송해서 단말에서 다운받아 처리한다면, 기존에 디코더는 물론, 특정 컨텐츠에 대해 특성화된 디코더까지 처리하는 데 문제가 되지 않을 것이다.

2. 제안하는 환경

자바 표준(Java Standards)[4] 에서는 CLDC[5], MIDP[6]란 표준을 통하여 무선 단말을 정의하고 있지만, 현 추세로 보았을 때 무선 단말의 환경은 1~2년 안에 지금보다 발전하여 PDA급으로 올라 갈 것이라고 보고 있다. 그래서 CLDC의 확장된 표준인 CDC[7] 기반에 iPAQ PDA를 이용해서 테스트를 하였다.

테스트한 동작 환경은 다음과 같다.

PDA : iPAG H3950 process : 400 MHz Intel PXA250 memory : 64MB RAM, 32MB ROM Java VM : StrongArm Java VM[8]

표1. 동작환경

3. 디코더를 다운받아 처리하기 위한 시스템 설계 및 구현

본 논문에서는 디코더가 내장되어 처리되는 방식이 아닌 디코더를 다운받아 처리하는 방식을 제안하도록 한다. 그리고 제안하는 방식을 다운로더블 디코더라고 정의하였다.

최적화된 Java MPEG Player를 보면 93.1kByte인 경우 CDMA 1x 망에서 디코더를 다운받는데 걸리는 1~2초 정도밖에 걸리지 않는다. 즉 메모리량의 한계성이 큰 무선통신단말에서는 메모리에서 각각에 컨텐츠 타입을 처리하기 여러 종류에서 디코더를 저장하는 방식보다 디코더를 재생할 때 다운받고, 끝난 뒤에 지우는 방식을 적용하면, 더 효율적이다.

이 논문에서는 실제 다운로더블 디코더 시스템을 구현하기 위해 설계한 과정을 제시하고, 구현에 관한 내용은 다음 논문에 나타내도록 하겠다.

그림 1은 다운로더블 디코더 시스템의 개략적인 구조도를 나타내 보인다.

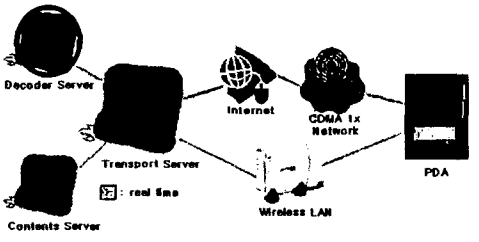

그림 1. 다운로더블 디코더 시스템의 개략적인 구조도

위의 구조도를 보면 시스템은 PDA를 통하여 Decoder와 Contents를 전송한다. 이 시스템이 다른 시스템과 다른 큰 차이점은 Decoder Server가 있어서 Transport Server를 통해 Decoder를 전송할 수 있다는 것이다. 이를 통하여, 다양한 포맷의 멀티미디어 서비스도 가능해졌다.

그림 2는 서버와 클라이언트간의 데이터처리 과정을 나타내 보인다.

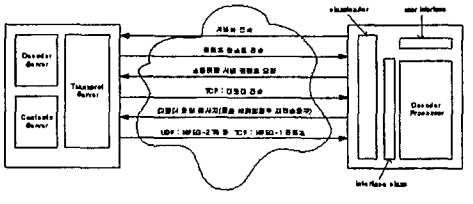

그림 2. 서버와 클라이언트간의 데이터처리 과정

위의 그림을 보면 서버에서 클라이언트로 접속을 하면 서버는 클라이언트에게 컨텐츠 리스트를 전송한다. 그러면, 클라이언트는 컨텐츠 리스트 중에서 하나를 선택하여 서버로 전송한다. 서버는 컨텐츠의 타입을 판단하여 디코더를 TCP망을 통해 전송하고 디코더가 다 전송되어 처리되었다는 클라이언트의 메시지를 받고, 컨텐츠를 전송하게 된다.

그림 3는 단말의 시스템 구조을 나타낸 것이다.

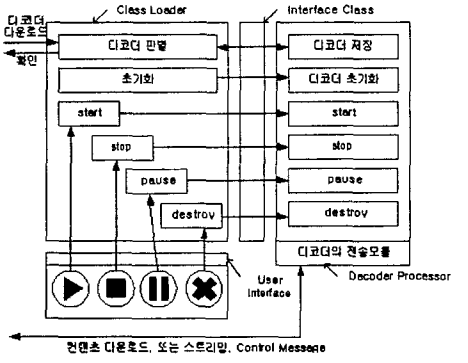

그림 3. 단말의 시스템 구조

Class Loader는 디코더를 받아서 저장하고 처리하는 역할을 하고, User Interface와 Interface Class와의 연동하여 디코더를 콘트롤 한다.

Decoder Processor는 디코더의 처리를 모두 관한한다.

Interface Class는 ClassLoader에서 디코더를 콘트롤 할 수 있도록 Interface methods를 정의하고, 이를 통해서 ClassLoader가 디코더를 제어할 수 있도록 한다.

User Interface는 내부모듈와 사용자를 연결하는 기능을 한다.

그림 4은 동작흐름도이다.

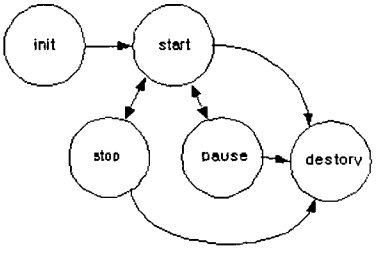

그림 4. 동작흐름도

위의 그림과 같이 실제 동작에 따라 Interface Method 를 정의하여, 모든 Decoder를 Interface Method로 제어할 수 있도록 하였다.

그림 5은 서버 시스템 구조이다.

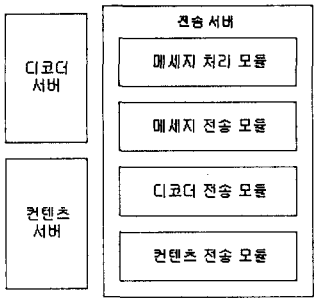

그림 5. 서버 시스템 구조

서버 시스템이 다른 시스템과 다른 구조를 가지는 것은 디코더 서버와 디코더 전송 모듈이다. 이를 이용하여 디코더를 전송하게 된다.

4. 결론

위의 설계 내용을 바탕으로 하여 구현하기 위하여 우선적으로 PDA에서 자바 멀티미디어 플래이어가 어떤 특징을 알아 보기 위하여 CDC포멧에 맞도록 수정하고 올려 보았다.

실제 가능성을 테스트하기 위해 CDMA 1x망이라고 가정하고, 144kbps에서 가상으로 구현하여 그 위로 80kbyte정도의 MPEG-1 플래이어를 전송하고 플래이 해 보았다.

실제 디코더를 전송하고 처리하는 데 걸리는 시간은 1~2초 사이였다.

그림6, 그림7은 그에 따른 실제 멀티미디어 MPEG-1 플래이어를 옮겨서 테스트 한 그림이다.

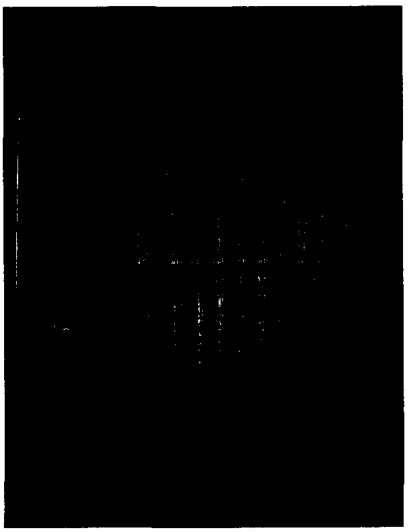

그림 6. Java MPEG Player의 실행 전 화면

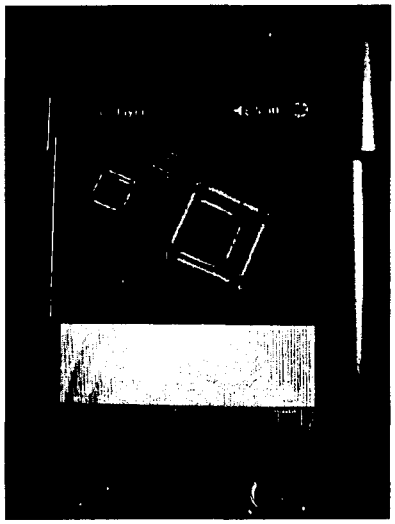

그림 7. Java MPEG Player의 실행 화면

다음 논문에서는 이 테스트를 바탕으로 실제 Downloadable Decoder를 구현하도록 하겠다.

5. 참고문헌

[1] "Accelerate your Java apps!", http://www.javaworld.com/javaworld/jw-09-1998/jw-09-speed.html

[2] "The Implementation of the 2D-IDCT", http://rnvs.informartik.tu-chemnitz.de/~ja/MPEG/HTML/IDCT.html

[3] Dohyung Kim, Jihong Kim, "Design and Implementation of a Java-Based MPEG-1 Video Deocder", IEEE Transactions on Consumer Electronics, Vol. 45, No. 4, Novemver 1999.

[4] "Java Standards", http://java.sun.com/j2se/index.html

[5] "CLDC", http://java.sun.com/products/cldc/index.html

[6] "MIDP", http://java.sun.com/products/midp/index.html

[7] "CDC", http://java.sun.com/products/cdc/index.html

[8] "Strong Arm Java VM", http://www.skelmir.com/ceej/ceej.html