

# MPV(MultiPhoto/Video) 적용에 있어서의 식별자 복구 알고리즘에 대한 연구

김두일\*

\*삼성전자 디지털 미디어 연구소

e-mail : [duilkim@samsung.com](mailto:duilkim@samsung.com)

## A Study for identifier (lastURL) fix-up algorithm in MPV's implementation

Du-Il Kim\*

\*Digital Media R&D Center, Samsung Electronics Co., LTD

### 요 약

MPV(MultiPhoto/Video)는 디지털 기기 간의 상호-연동성(Inter-operability)를 향상시키고, 사용자의 콘텐츠 관리를 용이하게 하기 위해서 제안되고 있는 표준안이다. MPV에서는 사용자의 편의를 위해서 데이터 부분에 해당하는 콘텐츠를 직접 관리하지 않고 XML 포맷으로 형성되는 메타데이터(Metadata)를 통하여 관리를 하게 되는데, MPV 와 관계 없는 소프트웨어에 의해서 콘텐츠의 정보가 변경되면, 이를 바로 잡을 수 있도록 복구 알고리즘을 제안하고 있다. 하지만, 제안된 알고리즘은 속도가 매우 느리다는 단점을 가지고 있어, 본 논문에서 이의 속도 및 효율을 향상시킬 수 있는 두 가지 알고리즘을 제안한다.

### 1. 서론

최근 들어 개인용 컴퓨터(Personal computer, 이하 PC 라 칭함)를 이용하여 디지털 사진(Digital picture), 비디오(Video), 디지털 오디오(Digital audio), 텍스트 등의 미디어를 처리하고, 재생하여 왔다. 그리고, 상기 기술한 미디어를 생성하는 장치, 예를 들어 디지털 카메라, 디지털 캠코더(Digital camcorder), 디지털 오디오 재생기(Digital audio player : MP3, WMA 등의 디지털 오디오 데이터를 재생하는 장치) 등의 장치 사용이 많아짐으로 인하여 다양한 종류의 콘텐츠가 대량으로 생성되고 있다.

하지만, 이렇게 대량으로 생성되고 있는 콘텐츠 - 멀티미디어 데이터 - 를 관리를 하기 위하여 개인용 컴퓨터를 주로 사용하여 왔고, 이로 인하여 사용자의 파일 기반 경험(File-based user experience)을 요구하게 되는 어려움이 있어왔다. 또한, 특정 제품에서 콘텐츠 생성 시 속성 데이터 - 데이터 재생 순서, 재생 방법 등 - 역시 각각의 콘텐츠 별로 생성되는 경우가 있는데, 이를 개인용 컴퓨터와 연결하면 속성 데이터를 잃어버리고, 원본 데이터만 이동되어버린다. 즉, 가전 제

품, PC, 및 디지털 콘텐츠 생성 장치간 데이터와 데이터의 속성에 대한 상호-연동성(Inter-operability)이 현재 매우 취약하다.

상호 연동성의 취약에 대해서 예를 들어 본다. 디지털 카메라를 사용하여 사진을 캡처하고, 캡처된 사진을 디지털 카메라에서 확인하기 위하여 슬라이드쇼(Slideshow) 기능을 사용하면서 결정된 속성 - 슬라이드쇼 순서, 사진간 시간 간격 등 -, 파노라마(Panorama) 기능을 사용하면서 결정된 속성 - 촬영된 사진 간의 관계 -, 연사 기능을 사용하면서 결정된 속성 등의 데이터가 원본 데이터 - 실제 화상 데이터 - 와 함께 저장되게 된다. 이러한 디지털 카메라를 에이브이(AV) 케이블을 사용하여 텔레비전으로 화상을 전송하게 되면 각각의 속성이 표현된 콘텐츠를 사용자가 볼 수 있게 된다. 그러나, 유에스비(USB : Universal Serial Bus) 케이블을 통하여 개인용 컴퓨터에 연결하게 되면 원본 데이터만 컴퓨터로 전송되며, 각각의 속성을 모두 상실한다.

상기 예에 나타난 바와 같이 디지털 카메라에서 저장된 속성 데이터, 즉 메타데이터에 대한 개인용 컴퓨터의 상호-연동성이 매우 취약하다는 결론을 내릴 수

있다. 또는, 디지털 카메라의 기능에 대한 개인용 컴퓨터의 상호-연동성이 없다고 결론을 내릴 수도 있다.

이러한 디지털 기기간 데이터에 대한 상호-연동성 취약을 보강하기 위하여 엠피브이(MPV : MultiPhoto/Video, 이하 MPV 라 칭함)라는 표준이 진행되고 있다.

본 논문에서는 MPV 표준 중에서 특정 식별자(lastURL)의 복구 알고리즘에 대한 연구 결과를 설명하고자 한다. 식별자 복구 알고리즘의 이해를 위해 먼저 MPV 에 대한 기본적인 내용에 대해서 설명하고, 이어서, 식별자 복구 알고리즘이 필요한 이유와, 제안하고자 하는 알고리즘에 대해서 차례로 설명하고, 마지막으로 후반부에서 실험 결과를 제시하도록 한다.

## 2. MPV(MultiPhotoVideo)

MPV 스펙(Specification, 이하 스펙이라 칭함)은 광학 디스크, 메모리 카드, 컴퓨터 하드 디스크 같은 저장 매체(또는 장치)에 저장되거나, 인터넷 프로토콜(Internet protocol)에 의해 교환되는 디지털 사진, 비디오, 오디오 등의 콘텐츠(Content)의 집합을 처리하고, 재생(Playback)하기 위한 매니페스트(Manifest), 메타데이터(Metadata) 및 프랙티스(Practice)를 정의하고 있다.

MPV 는 현재 OSTA(Optical Storage Technology Association)와 I3A(International Imaging Industry Association)에서 표준화를 진행하고 있으며, 오픈 스펙(Open specification)이며, 디지털 사진이나 비디오, 디지털 오디오, 텍스트 등의 집합을 처리하고, 교환하고, 재생하는 것을 용이하게 하는 것을 주 목적으로 한다.

먼저, MPV 에 대해서 간단하게 살펴보도록 한다.

MPV 는 크게 MPV 코어 스펙(MPV Core-Spec. 0.90WD)과 프로파일(Profile) 두 가지 분류로 나뉘어 있다.

먼저, 코어에 대해서 살펴보도록 한다.

코어는 3 가지 기본 요소로 이루어져 있다. 3 가지 요소는 컬렉션(Collection), 메타데이터, 식별(Identification)이다.

컬렉션으로는 루트 멤버(Root member)로 매니페스트를 가지고 있으며, 메타데이터, 앨범(Album), 표시 자산(MarkedAsset), 자산 리스트(AssetList) 등이 있다. 여기서, 자산은 2 가지 종류가 있다. 먼저, 심플 미디어 자산(Simple media asset)이 있으며, 디지털 사진, 비디오, 디지털 오디오, 문서 등이 그 예에 해당한다. 그리고, 복합 미디어 자산(Composite media asset)이 있으며, 디지털 사진+디지털 오디오(StillWithAudio), 연속 촬영 디지털 사진(StillMultishotSequence), 파노라마 촬영 디지털 사진(StillPanoramaSequence) 등이 그 예에 해당한다.

메타데이터는 XML 포맷을 따르며, 식별을 위해 모두 하기와 같이 5 개 종류의 식별자가 있으며 각각의 내용은 다음과 같다.

- 라스트 유알엘(LastURL): 해당 자산의 패스이름과 파일 이름(Path to the object)
- 인스턴스 아이디(InstanceID): 각 자산 마다 유일한 아이디(unique per object : 예, Exif 2.2)

- 문서 아이디(Document ID): 원본 데이터와 변경된 데이터에 모두 동일 (Same for all renditions)
- 콘텐츠 아이디(Content ID): 해당 자신이 무엇을 위해 사용되어 질 때마다 생성
- 아이디(id): 메타데이터 내에서의 로컬 변수

프로파일은 모두 7 개로 하기와 같이 예견되어 있다.

- 기본 프로파일(Basic profile)
- 프리젠테이션 프로파일(Presentation profile)
- 캡처/편집 프로파일(Capture/Edit profile)
- 기록 프로파일(Archive profile)
- 인터넷 프로파일(Internet profile)
- 인쇄 프로파일(Printing profile)
- 컨테이너 프로파일(Container profile)

이중 현재 발표된 것은 기본 프로파일과 프리젠테이션 프로파일이며, 모두 0.9 버전이다. 프로파일은 본 논문의 내용과 관련이 크지 않은 관계로 본 논문에서 설명은 피하기로 한다.

## 3. 콘텐츠 기반 파일 시스템(Contents based file system)

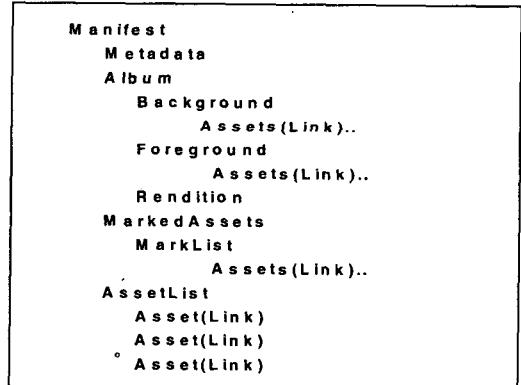


그림 1. MPV 파일의 구조를 설명하기 위한 예시도

그림 1 은 MPV 파일(메타데이터)의 구조에 대한 한 예들 든 것으로, 매니페스트(Manifest) 루트 멤버에 메타데이터, 앨범, 표시 자산, 자산 리스트가 XML 포맷으로 기록되어 있다. MPV 소프트웨어는 상기 MPV 파일에 기록되어 있는 상기 내용을 바탕으로 자산을 읽고, 재생하게 된다. 즉, MPV 파일은 MPV 소프트웨어와 자산이라는 데이터 사이에 위치하여 연결 역할을 하게 된다. 이는 기존 파일 시스템과 유사한 동작으로 상위 레벨의 파일 시스템이라 볼 수 있다. 따라서, 본 논문에서 이를 콘텐츠 기반 파일 시스템이라 칭하기로 한다.

상기 콘텐츠 기반 파일 시스템은 XML 포맷의 메타데이터로 존재하여 하위 파일 시스템에 독립적이다.

## 4. 복구 알고리즘에 대한 이해 및 연구

먼저, 복구 알고리즘의 이해를 위하여 MPV 스펙에

서 제안하고 있는 복구 알고리즘의 일부에 대해 설명 하도록 한다.

MPV 는 상기 기술한 바와 같이 컨텐츠 기반 파일 시스템이다. 하지만, 개인용 컴퓨터인 경우 MPV 관련 소프트웨어 이외의 소프트웨어는 특정 자산을 직접 제어할 수 있도록 노출되어 있다. 이로 인하여 자산의 이름 변경, 이동, 삭제 등의 제어에 의하여 MPV 파일 시스템의 내용과 실제 자산과 일치하지 않게 되는 문제가 발생한다. 특히, 파일 이름과 경로 이름에 해당하는 라스트 유알엘이 파손되게 된다. 이렇게 파손된 라스트 유알엘을 복구하기 위한 방법을 MPV 코어 스펙에 간단히 제안하고 있으며, 이 중 간단한 예를 그림 2 에 도시하였다.

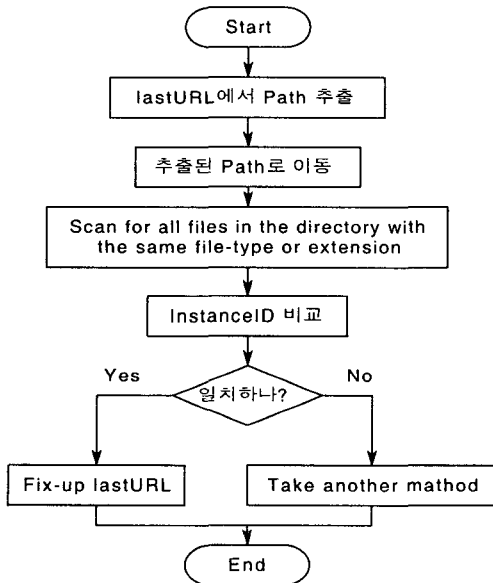


그림 2. MPV 스펙에서 제안하고 있는 라스트 유알엘(lastURL) 복구 알고리즘

그림 2 은 MPV 코어 스펙에서 제안하고 있는 라스트 유알엘 복구 알고리즘 중 간단한 예를 든 것으로, 그림에 표시된 바와 같이 현재 라스트 유알엘에서 경로 이름을 추출하고, 추출된 경로로 이동해서 확장자가 동일한 파일을 검색하고, 검색된 파일과 해당 자산의 인스턴스 아이디를 비교해서 일치하는 파일이 있으면, 일치된 파일 이름과 경로 이름으로 라스트 유알엘을 대체한다.

그림 2 에서 제시된 알고리즘에서 추출한 경로에서 인스턴스 아이디가 일치하는 파일을 찾지 못한 경우 하드 디스크 드라이브(HDD : Hard Disk Drive)전체를 검색해야 한다. 이럴 경우 라스트 유알엘 복구에 많은 시간을 요구하게 되어 MPV 소프트웨어의 자산 사용에 지연이 발생되는 문제가 발생한다.

이를 해결하기 위하여 본 논문은 다음 2 가지 중요 알고리즘을 제안한다.

먼저, 본 논문의 이해를 돕기 위해서 MPV 코어 스펙

에 기재되어 있는 연출(Rendition) 멤버에 대해서 간략히 살펴본다.

연출 멤버에서 정의 할 수 있는 것은 다음과 같다.

- 마스터(Master) : 원본 자산
- 썸네일(Thumbnail) : 마스터의 축소본(프리뷰에 사용)
- 스크린(Screen) : 재생할 스크린(주로 텔레비전)의 해상도로 변환(마스터를 축소 또는 확대함)
- 기타, 하이레스(highRes), 로우레스(lowRes), 프린트, 쇼(Show) 등이 있다.

대표적인 연출로 상기 기재한 바와 같이 스크린과 썸네일이 있다.

```

...
<mpv:AssetList>
...
<mpv:Still mpv:id="ID000500" mpv:lastURL="DSC09345.JPG">
  <mpv:Rendition mpv:renditionType="thumbnail">
    <mpv:StillRef mpv:idRef="ID000600"/>
  <mpv:Rendition>
  <mpv:Rendition mpv:renditionType="screen">
    <mpv:StillRef mpv:idRef="ID000700"/>
  <mpv:Rendition>
</mpv:Still>
<mpv:Still mpv:id="ID000600" mpv:lastURL="thumbs/DSC09345.JPG"/>
<mpv:Still mpv:id="ID000700" mpv:lastURL="screen/DSC09345.JPG"/>
...
</mpv:AssetList>
    
```

그림 3. 연출 멤버의 일 예

그림 3 에 연출 멤버의 일 예를 도시하였다. 도면에 도시된 바와 같이 인스턴스 아이디는 비록 원본 데이터(마스터)와 다르고, 해상도 역시 다르지만 동일한 내용을 담고 있는 자산(썸네일과 스크린으로 연출된 자산)이 2 개의 파일로 존재하게 된다.

본 논문에서는 이와 같이 연출된 자산을 이용하여 임시적으로 라스트 유알엘을 수정(복구)하고, 시스템의 리소스 사용 상태에 따라 그림 2 에 도시한 바와 같은 실제 복구 알고리즘을 백그라운드로 실행하는 것을 첫번째 알고리즘(이하, 임시 복구 알고리즘이라 칭함)으로 제안한다.

상기 제안한 방법을 사용하면 MPV 소프트웨어에서 임시적으로 복구된 라스트 유알엘을 사용하여 사용자의 명령에 따라 즉각 반응을 하게 되며, 추후 실제 복구 알고리즘을 백그라운드(Background)로 수행하여 라스트 유알엘을 수정하면 보다 완벽한 복구가 가능해진다.

이때, 임시적으로 복구된 라스트 유알엘이 실제 복구된 것이 아니라, 임시적임을 나타내기 위하여 메타 데이터 멤버에 기록하여야 한다.

개인용 컴퓨터인 경우 하드 디스크 드라이브에 많은 경로와 파일들이 존재한다. 이렇게 많은 경로와 파일을 검색해서 일일이 인스턴스 아이디를 비교하는 것은 많은 시간을 필요로 한다. 복구 시간을 단축하기 위한 복구 알고리즘 두 번째 제안으로 제시한다.

먼저, 경로 각각에 우선순위(Priority)를 부여한다.

이어서, 상기 우선순위에 따라 파일 검색/인스턴스 아이디 비교/라스트 유알엘 복구를 수행한다.

마이크로 소프트웨어의 윈도우 익스피를 예로 경로의

우선순위를 하기 표 1 을 참조하여 설명한다.

표 1. 경로 우선 순위 일 예를 나타낸 테이블

우선 순위	경로특징	경로(예)
0	라스트 유알엘에서 추출한 경로	C:\...
1	사용자 문서	C:\Documents and Settings\KilDongHong\My Documents
2	공유 문서	C:\Documents and Settings\All Users\Documents
3	다른 사용자 문서	C:\Documents and Settings\User1\My Documents C:\Documents and Settings\User2\My Documents ...
4	1, 2, 3, 5, 6 을 제외한 경로	C:\HunData C:\My Music ...
5	프로그램(SW)	C:\Program Files
6	임시 파일	C:\WINDOWS\TEMP C:\Documents and Settings\User\Local Settings ...
7	시스템 경로	C:\WINDOWS

표 1 에 기재된 우선순위는 낮은 수일수록 높은 우선순위를 가진다.

우선순위에 따른 라스트 유알엘 복구는 다음과 같이 수행한다.

- 우선순위 '0'의 경로로 이동한다.
- 경로 내에서 복구 대상 자산과 동일한 확장자의 파일을 검색하고 인스턴스 아이디를 비교한다.
- 상기 비교 단계에서 일치하는 파일이 발견되면 라스트 유알엘을 수정하고 복구 알고리즘(루틴)을 종료한다.
- 상기 비교 단계에서 일치하는 파일이 발견되지 않으면 다음 우선순위의 경로로 이동한다.
- 상기 2 번째 단계부터 4 번째 단계를 반복적으로 수행한다. (단, 검색 및 비교를 끝낸 경로는 모두 기억-저장-하여 하위 경로로서 동일한 경로를 다시 검색/비교하게 되는 일을 방지한다)

실제 우선순위 '6' 및 '7'의 경로에 대한 검색은 선택적으로 수행하지 않도록 할 수도 있다. 실제 우선순위 '6' 및 '7'의 경로에서 손실된 자산을 찾을 경우는 크지 않기 때문이다. 단, 인터넷 프로파일의 결정되면, 개인용 컴퓨터에 구현되는 경우 우선순위 '6'의 경로 중에 인터넷 임시 경로에서 손실된 자산을 찾을 확률이 있다. 이런 경우 인터넷 임시 경로를 우선순위 '6'으로 두고, 나머지 임시 경로들을 우선순위 '7'로 설정하여 복구 알고리즘을 수행 할 수도 있다. (이때 기존 우선순위 '7'이하의 경로는 한단계씩 우선순위가 밀릴 수 있다)

### 5. 실험 데이터

상기와 같이 제안한 알고리즘의 효과를 검증하기 위하여 간단한 실험을 수행하였다. 실험 장치로 실제 MPV 를 적용할 수 있는 장치 중 하나인 개인용 컴퓨터를 선택하였으며, 그 스펙은 다음과 같다.

- CPU : Pentium 4 1.6G
- HDD : 80GB
- OS : Windows XP Professional
- RAM : 512MB(SDRAM)

테스트 결과를 다음 표 2 를 참조하여 설명한다.

표 2. 테스트 결과 테이블

우선 순위	경로(예)	전체 크기	전체 파일 수	검색된 그 림 파일 수	경과 시간 (sec)
5	C:\Program Files	4.74 GB	63,071	1074	3 분 23 초
6	C:\Documents and Settings\User\Local Settings	434 MB	3882	396	54 초
7	C:\WINDOWS	2.15 GB	14,143	141	51 초

사용자의 수가 4 명이라고 가정하면 표 1 의 결과에 따라 단축되는 전체 검색 시간은 7 분 50 초(3 분 23 초 + 3 분 36 초 + 51 초)가 된다. MPV 에 적용되는 경우 파일을 비교하는 시간 까지 포함해야 하므로, 전체 1611 개의 파일에 대한 비교 시간이 단축된다. 따라서, 7 분 50 초 이상의 시간을 절약할 수 있게 된다.

테스트 결과는 사용하는 개인용 컴퓨터에 따라, 사용자의 습관에 따라 많은 차이를 보일 것이며, 표 1 은 단지 하나의 참조일 뿐이다.

### 6. 결론

본 논문에서는 디지털 제품간의 상호-연동성을 강화시키고, 사용자의 편의를 위한 스펙인 MPV 의 디지털 제품에 적용함에 있어서, 라스트 유알엘을 복구하는 효율적인 알고리즘에 대해 제안을 하였다. 제안된 알고리즘 중 임시 복구 알고리즘은 화질 면에서 다소 단점이 있을 수 있으나, 실시간(Real time)에 가까운 복구 속도를 보여 줄 수 있으며, 두 번째 알고리즘은 기존과 동일한 품질을 보여주되, 속도면에서 기존에 비하여 좋은 결과를 가져올 수 있다.

본 논문에서 제안한 알고리즘은 개인용 컴퓨터 뿐만 아니라, 개인용 비디오 리코더(PVR : Personal Video Recorder), 디브이디 레코더(DVD Recorder) 등의 여러 가지 디지털 제품의 구현에 사용이 가능하다.

### 참고문헌

[1] <http://www.osta.org/mpv/mpvmbms/specs.htm/MPVCore-Spec-0.90WD.pdf>, "MPVCore-Spec", Ver. 0.9, OSTA, Jun. 28. 2002