

# WIPI 기반 3D API 설계

\*손민우, 허원, 신동규, 신동일

세종대학교대학원컴퓨터공학과 멀티미디어 인터넷 연구실

{minwoo15, heowon, shindk, dshin}@gce.sejong.ac.kr

## Design of 3D API based on WIPI

\*MinWoo Son, Won Heo, Dongkyoo Shin, Dongil Shin

Dept. of Computer Engineering, Sejong University

### 요약

무선인터넷을 중심으로 한 휴대폰 단말, 모바일 통합 플랫폼, IMT 2000 관련 기술의 끊임없는 발전으로 가입자들이 급속히 확산되어 가고 있다. 또한 수 년 전부터 무선인터넷 등장과 이로부터 파생되는 전체 산업의 구조 변화가 크게 일어나고 있다. 그리고 현재 IMT 2000은 국내 이동통신사들을 중심으로 본격적인 서비스 준비를 끝내고 도약을 위한 준비가 한창이고 2001년 중반 이후 개발이 시작된 모바일 표준 플랫폼 WIPI(WIPI:Wireless Internet Platform for Interoperability)가 국내의 통합 플랫폼으로 자리매김을 하고 있다. 따라서 모바일 표준 플랫폼 WIPI에서도 3D 컨텐츠가 활성화 될 것으로 예상된다. 이에 본 논문에서는 WIPI 기반의 3D API 개발을 위한 설계에 대하여 기술하였다.

### 1. 서 론

무선인터넷을 중심으로 한 휴대폰 단말, 모바일 통합 플랫폼 및 IMT 2000 관련 기술의 끊임없는 발전으로 가입자들이 급속히 확산되어 가고 있다. 또한 수 년 전부터 무선인터넷 등장과 이로부터 파생되는 전체 산업의 구조 변화가 크게 일어나고 있다. 그리고 현재 IMT 2000은 국내 이동통신사들을 중심으로 본격적인 서비스 준비를 끝내고 도약을 위한 준비가 한창이며, 그 중심에는 모바일 플랫폼이 자리 잡고 있다. 또한 모바일 플랫폼은 이미 J2ME · GVM · MAP 등 기존 방식에서 기술적인 점증이 끝난 상태이다. 최근 SK 텔레콤은 모바일 플랫폼인 SK-VM 서비스는 이미 서비스가 활성화 되어 국내 약 1,000만 대의 휴대폰에 포팅되어 있으며, 이스라엘 및 대만 등에 수출되고 있다. KTF 또한 이미 2001년 11월부터 월컴의 브루 플랫폼을 도입해 멀티팩 서비스를 시작한 이후 빠른 가입자 확보 증가율을 보이고 있다. 2001년 중반 이후 개발이 시작된 모바일 표준 플랫폼 WIPI(Wireless Internet Platform for Interoperability)가 현재 일반 업체나 이동통신사가 단독으로 개발한 것이 아니라 정부가 주도하고 투자해 국내의 통합 플랫폼으로서, 한국정보통신기술협회(TTA)의 인증까지 획득된 상태이다[6]. 따라서 모바일 표준 플랫폼 WIPI에서도 3D 컨텐츠가 활성화 될 것으로 예상된다. 또한 WIPI는 △응용 프로그램 및 API 관리기능과 안정화된 플랫폼 보안관리기능 △다국어 지원 및 C언어, 자바 언어로 작성된 응용 프로그램의 실행 환경 △다중 응용 프로그램의 동시실행과 프로그램간 통신기능 제공 △고효율의 메모리 관리기능 등을 제공한다[7]. 그러나 주요 API 중에서 그래픽 관련 API는 점, 선 및 사각형을 표현하고 이미지를 디스플레이 하는 단순한 기능에 머물고 있어서 좀 더 복잡한 그래픽이 요구되는 멀티미디어 컨텐츠(예: 비디오 플레이어, 게임 등)의 제작에 많은 시간이 필요하다. 따라서 본 논문에서는 미래 컨텐츠의 중심축인 이동 멀티미디어 컨텐츠를 WIPI 기반에서 손쉽게 제작할 수 있는 WIPI용 고성능 멀티미디어 라이브러리와 만들어진 컨텐츠를 손쉽게 테스팅 할 수 있는 PC 기반의 저작도구를 개발하기 위한 WIPI 기반의 3D API 개발 설계에 대하여 기술하였다.

### 2. WIPI (Wireless Internet Platform for Interoperability)

#### 2.1 개발 배경

이동통신3사를 중심으로 한국통신기술협회, 전파연구소, 전자통신연구원 등이 모여 2001년 8월부터 한 달 동안의 플랫폼에 필요한 요구사항을 만들어 9월초에 포럼을 통해 발표한 “표준 플랫폼에 대한 이동통신 3사의 요구 사항”을 만족하도록 만들어졌으며 이 요구 사항 문서는 기존에 서비스되고 있는 다양한 플랫폼들의 장점을 수용하며 차세대 서비스에 필요한 기능을 추가 하였고, 기존 플랫폼의 단점을 보완하도록 요구하고 있다. WIPI 규격은 이동통신 단말기(이하 단말기로 표기함)에 탑재되어 응용 프로그램을 수행할 수 있는 환경을 제공하는 모바일 표준 플랫폼 규격(이하 플랫폼 규격)으로 정의되고 WIPI 규격을 만족하는 모바일 플랫폼(이하 플랫폼으로 표기함)은 단말기용 응용 프로그램 개발자에게는 플랫폼 간 컨텐츠 호환성을 보장하고, 단말기 개발자에게는 플랫폼의 이식의 용이성을 제공하며, 일반 이용자에게 다양하고 풍부한 컨텐츠 서비스의 제공을 목적으로 한다.

#### 2.2 WIPI의 주요 기능 규격

WIPI가 제공하는 주요 기능 규격은 기본 API를 통해 지원될 수 있는 부분과 플랫폼 내부에서 처리해야 하는 부분으로 구성되어있으면 이러한 기능들은 기존의 플랫폼 기능을 모두 수용하고, 기존의 플랫폼에서 적용되지 않았던 새로운 기능을 적극 수용하여 차세대 단말기에서 손색이 없도록 고려되어 △C 및 자바 언어로 작성된 응용프로그램의 실행 환경을 제공 △Java언어로 작성된 프로그램에 대해 고속의 바이너리 코드 실행 환경을 제공 △다중 응용프로그램의 동시 실행 환경을 제공하며, 다중 응용프로그램간 통신 기능을 제공 △다운로드에 의한 동적 공유 라이브러리를 지원함에 따라, 동적으로 API를 추가/갱신하는 기능을 제공 △고효율적인 메모리 관리 △메모리 압축(Memory Compaction) 기능과 함께 자동 메모리 정리(Garbage Collection) 기능을 제공 △응용프로그램 종료 시 자동으로 메모리 해제 기능을 제공 등의 몇 가지 주요 특징을 갖는다[3].

### 2.3 WIPI 플랫폼의 개념적 구조 및 표준화 개요

WIPI 규격은 플랫폼 이식성을 높이기 위한 표준화된 하드웨어 추상화 계층인 HAL(Handset Adaptation Layer)과 표준화된 플랫폼 호환성을 제공하여 다양한 응용 프로그램 개발을 촉진하기 위한 기본 응용 프로그래밍 인터페이스(Basic API : Basic Application Programming Interface)로 구성되며 플랫폼 개발언어로는 C 언어 및 자바 언어를 모두 지원하도록 규격화 하여, 개발자의 참여 폭을 최대화하고 있으며 또한 이동통신사업자의 서비스 차원의 차별화를 위해 규격을 필수 기능과 선택 기능으로 분류하였으며, 특히 동적 API 추가/생신 기능을 통하여 차별화된 API를 동적으로 제공할 수 있도록 한다[4].



### 3. WIPI용 3D API의 필요성

이동통신 인프라의 급격한 발전과 단말기의 빠른 진화에 따라서 2005년 이후에는 대형 LCD, 대용량 메모리 및 고속 CPU를 탑재한 스마트 단말기가 주종을 이를 것으로 예상되어 멀티미디어 애플리케이션의 요구 사항 변화에 될 것이다. 이에 모바일 3D 컨텐츠는 선으로 이뤄진 다각형면(polygon)이 합쳐져 입체적인 캐릭터를 만들어내는 것으로, 2차원 캐릭터가 상하좌우로만 움직이는데 반해 원근감을 자유롭게 표현하고, 자연스럽게 회전하는 등 사실감을 크게 높일 수 있는 것이 장점이다. 또한 모바일 3D 컨텐츠가 일반화 되면 스트리트 파이터 등 모바일 게임 캐릭터 및 배경의 사실감이 크게 증대되고, 휴대전화에서 입체적인 사이버 애완동물도 키울 수 있다. 또 데이터 크기의 변동 없이 아바타 동작을 구현하고 웃을 갈아입힐 수 있는 등 다양한 서비스가 가능하고 3D 서비스는 또 평면 그림이 아닌 다각형 면으로 구성돼 자연스러운 동작을 구현하면서 2005년도의 휴대전화 수준에서도 자유로운 동작 구현이 가능하도록 개발되어야 한다.

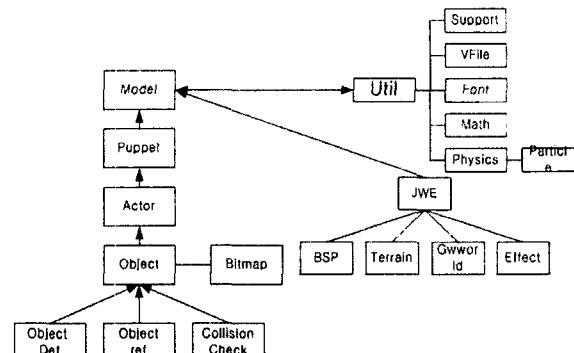
### 4. 모바일 3D API 설계

#### 4.1 모바일 3D API 개발 시의 고려 사항

- 게임을 위한 랜더링에 최적화 한다.
- Radio Lighting 처리를 한다.
- 광도와 게임 화면에서의 랜더링의 최적화를 위해 전처리 연산된 Light 효과를 지원한다[1].
- 적합한 게임 환경 및 빠른 랜더링을 위해 쿠팅 시스템으로 BSP 트리구조를 사용한다.
- 게임 Texture를 위해 morphing과 blending 등의 애니메이션 효과를 접목한다.
- 선택적인 World 기하구조를 위해 Portal 랜더링을 지원한다[1].
- 폴리곤 및 매쉬의 크랙을 방지하기 위해 Sky를 구형으로 처리한다.
- Sound 효과에 대해 3D 사운드의 위치에 따른 감쇄 효과를 지원한다.
- 게임 사용자를 위해 특수효과의 확장성을 갖고 있으며 저레벨의 시스템을 지원한다.

- 스크린에서 나오는 텍스처와 비트맵은 윈도우에서 지원하는 비트맵과 폰트를 레이블링하는 것을 지원한다.

### 4.2 모바일 3D API를 구성하는 주요 클래스들의 구성



[그림 2] 모바일 3D API 클래스 구성도

본 논문에서는 3D API 구성에서 가장 핵심적인 부분인 Puppet 클래스에 대해서 상세히 기술하였다. Puppet의 형태는 일반적인 3D 그래픽 디자인 도구에서의 Puppet의 의미와 비슷하며 Puppet에 필요한 정보는 크게 세 부분으로 정의할 수 있는데 텍스처와 Material에 대한 정보와 Light에 대한 정의 값을 그리고 Shadow에 관련된 정보들이 Puppet의 주요 구성부분이다[2].

#### 4.3 Puppet 구조체

구조체의 멤버들에 대한 정보는 다음과 같다.

[표 1] Puppet 구조체

구조체	설명
TextureImageContext	텍스처 정보를 가진 파일 정보
*Bodyinstance	Body 정보를 가진 구조체 대한 인스턴스
MaterialCount	Material에 매칭되는 정수 값
*MaterialArray	Puppet에서 사용하는 Material에 대한 정보
MaxDynamicLightToUse	최대 동적 광원의 정수 값
MaxStaticLightToUse	최대 정적 광원의 정수 값
LightReferenceBoneIndex	Bone들의 ROOT 조인트 값, 초기에는 -1로 정의
FillLightNormal	3차원 형태의 빛 정보
FillLightColor	RGB 형태의 광원 색상 값
UseFillLight	FillLight의 사용여부
AmbientLightIntensity	주변 광원의 색상
AmbientLightFromFloor	주변 광원 여부
PerBoneLighting	부분 라이트가 적용되는가 여부
Slights[MAX_STATIC_LIGHTS]	최대 정적 광원들에 대한 인덱스 배열
SlightCount	인덱스 숫자
*BoneLightArray	부위에 적용되는 동적 및 정적 광원에 대한 정보
BoneLightAxisSize	BoneLightArray의 크기
DoShadow	음영(그림자)처리 여부
ShadowScale	음영 크기를 정의

*ShadowMap	음영에 맵핑이 되는 텍스쳐
ShowBoneIndex	부위에 적용이 되는 음영의 수
pEngine	Puppet에 적용하기 위한 비트맵에 대한 정보
fOverallAlpha	알파값

## 5. Puppet 클래스 내부 메소드

### 5.1 Create Puppet

모델 공간에서의 Puppet을 생성하는 기능을 한다. 내부적인 동작으로 Puppet을 생성하기 위해 Body의 인스턴스 정보를 얻어오는 것과 Puppet에 대한 메모리를 확보하는 것, Puppet에 대한 비트맵 정보를 생성하는 것이 주요 동작이며 Puppet에 관련되어 최초로 호출되는 함수이다.

### 5.2 Destroy Puppet

모델 공간에 생성된 Puppet을 소멸하는 기능을 한다. Puppet을 생성하기 위해 호출했던 Create()에서의 정보를 소멸하는 역할을 한다. 특정적인 동작은 없으며 내부적으로 참조하는 함수들의 소멸 함수들을 호출하는 역할을 할 뿐이다.

### 5.3 Render Puppet

생성된 Puppet을 렌더링 하는 역할을 한다. 카메라에 대한 클리핑을 수행하기 위해 클리핑 화면의 크기를 설정한다. 오브젝트에 카메라를 적용하기 위해 카메라의 상태를 체크한 이후에는 카메라의 현재 좌표를 얻는다. 이 좌표는 클리핑 이후의 좌표로 변환이 되며 프로젝션을 적용한다.

### 5.4 Render Through Frustum Puppet

위에서 언급한 Render Puppet 함수는 카메라에 Frustum에 대한 정보를 가지고 있다는 조건에 의해 호출이 되어 동작하는 함수이다. 만약 Frustum에 대한 정보를 주지 않았을 경우에는 Render Through Frustum Puppet을 호출하여 Frustum에 대한 정보를 설정하여 준다.

### 5.5 Get&SetMaterial Puppet

Puppet 데이터에 재질을 입히기 위해 사용되는 함수들이다. 이 함수들도 여타 다른 함수들과 마찬가지로 Get&SetMaterial Actor의 하부구조를 이룬다.

### 5.6 Light Puppet

Puppet에 라이트 정보를 주는 방식은 일반적인 렌더링 파이프라인에서 설정을 따라 Geometry 스테이지에서 결정을하게 되며 이는 크게 세 가지로 구분이 된다. 첫 번째는 DynamicLight로 동적 라이트 환경을 매핑하기 위해 사용한다. 이 데이터들은 StaticLight 의 DynamicLight 배열에 저장된다. 두 번째는 StaticLight로 정적 라이트 환경을 매핑하기 위해 사용한다. 세 번째는 AmbientLight로 주변광 환경을 매핑하기 위해 사용한다. 이와는 별도로 Light에 대한 옵션을 설정하여 Actor의 하부구조를 구성하는 SetLightOption과 GetLightOption이 있다.

### 5.7 Vertex Color

일반적인 렌더링 파이프라인 구조에서 오브젝트의 형태를 그려낼 때 꽂지점을 중심으로 그려낸다. 각 꽂지점의 색상을 기준으로 하여 그 색상의 값을 뒤섞는 형식으로 오브젝트 전체를 그려내는 것이다. SetVertexColor는 이 기능을 담당한다.

### 5.8 SetShadow Puppet

Puppet\_SetShadow는 명암을 주기 위한 함수이다. 기존의 ShadowMap이 존재하면 기존의 ShadowMap을 지우고 새로운 설정을 대입한다. 여기서의 새로운 설정은 오브젝트에 추가되는 비트맵에 대한 카운터 정보를 말한다. 이 함수는 이후에 ActorSetShadow에 의해 호출이 되어 사용된다.

## 6. 결 론

본 논문에서는 미래 컨텐츠의 중심축인 이동 멀티미디어 컨텐츠를 위한 WIPI 기반의 3D API 설계에 대하여 기술하였다.

WIPI 기반의 모바일 3D API 설계시 문제점으로 개발시에 발생할 수 있는 이벤트 처리, Timer 세팅 및 콜백 함수 처리, 메모리 관리, 표준 함수 사용 등을 고려하는 부분이 어려웠으며 실 개발 시 적용하고 보완해야 할 것이다.

## 7. 참고문헌

- [1]Moller, Tomas, Eric Haines, "Real-Time Rendering", A K Peters Ltd., 1999
- [2]Kelly Dempski, "Real-Time Rendering tricks and techniques in DirectX", Premier, 2002
- [3]김경민, "무선인터넷 표준 플랫폼(WIPI)상의 어플리케이션 개발", JAVA Unlimited 2003
- [4]한순희, "Wireless Internet Platform", 여수대학교
- [5]앵구리닷컴 <http://www.ddanguri.com>
- [6]배석희, "모바일 플랫폼 표준화 동향 및 향후발전방향", TTA 저널 제82호
- [7](주)아로마소프트 "제1차 WIPI ION 상호운용성 시험"