



Technical integration into mobile phone  
New concept of mobile phone  
New service on the mobile phone  
New business generation

# Mobile 3D Game

김성재 / (주) 넥서스칩스  
[www.nexuschips.com](http://www.nexuschips.com)

**Nexus  
Chips**

# CONTENTS

- **국내 모바일 3D 게임**
  - 모바일 게임 시장 분석
  - 게임폰 및 서비스
- **국내 WIPI-C 환경**
  - SKT WIPI-C
  - KTF WIPI-C
- **WIPI기반의 3D 솔루션**
  - 개발 환경 및 툴
  - 개발 방법
- **OpenGL|ES 표준화 방향**
  - OpenGL|ES 1.1
  - OpenGL|ES 2.0
- **하드웨어 가속칩 현황**

# 국내 모바일 3D 게임

## ▶ 모바일 게임 시장 분석

### • 2D 게임

- 특정 개발 플랫폼을 사업자가 제조사에 요청
- 게임관련 제조사의 영역 미미 : 플랫폼 탑재 & 2D 임베디드 게임
- 다수의 다운로드 게임 제작 : 비용↓, 개발 기간 ↓, 가격 ↓, 서비스 폰↑

### • 2D 게임 및 Simple 3D 게임(S/W 렌더러)

- WIPI 플랫폼으로 통일
- 개발 플랫폼과는 별도로 3D 엔진 필요 (콘텐츠 호환 이슈)
- 제조사에서 차별화 전략으로 3D 임베디드 게임의 중요도 증가
- 기존 다운로드 서비스와 동일 (게임 용량 상향 조정, 최대 1 Mbyte)
- 사업자별 공모전을 통한 3D 콘텐츠 확보 전략

## ▶ 모바일 게임 시장 분석

- **고용량 2D 게임 및 3D 게임 (H/W 가속칩)**
  - 제조사 + 사업자 논의로 게임폰 제작
  - **WIPI**와 더불어 **3D low-level API**는 **OpenGLES** 로 표준화
  - 게임 포털 사이트 오픈 : 대작 **3D** 게임 / 퍼블리셔 등장
  - 온라인/**PC** 게임의 모바일화
  - **PC** 다운로드 및 무선망 인증 방식으로 서비스 형태 변환 (고용량 콘텐츠)
  - 향후 일반 폰에 가속칩이 내장되어 **3D** 게임의 보편화 가능

# 국내 모바일 3D 게임

## ▶ 게임폰 및 서비스



PH-S6000



LG-SV360 / LG-KV3600



2,2인치 QVGA 액정으로 3D 게임을 즐긴다  
GPANG Game Phone SPH-G1000

SPH-G100 / SCH-G100



카메라 MP3는 기본! 이젠 3D게임으로 승부한다  
GXG 3D Game Phone IM-8300

IM-8300

# 국내 모바일 3D 게임

## ▶ 게임폰 및 서비스

- 고용량 **2D/3D** 게임
- **GXG (www.gxg.com)**
  - 게임폰 (삼성 **SCH-G100**, LG **SV360**, SKTT **IM-8300**, P&C **PH-S600**)
  - **WIPI + GIGA** 표준 스펙 정의 → 콘텐츠 호환성 확보
  - 온라인/PC 게임의 모바일화
- **GPANG (www.gpang.com)**
  - 게임폰 (삼성 **SPH-G1000**, LG **KV3600**)
  - 제조사별 스펙 정의 → 콘텐츠 호환성이 해결 과제
    - **OpenGLES**로 표준 가능성
  - 게임 퍼블리셔들과 파트너형태로 **3D** 게임 소싱 다각화

# CONTENTS

- 국내 모바일 3D 게임
  - 모바일 게임 시장 분석
  - 게임폰 및 서비스
- **국내 WIPI-C 환경**
  - **SKT WIPI-C**
  - **KTF WIPI-C**
- **WIPI기반의 3D 솔루션**
  - 개발 환경 및 툴
  - 개발 방법
- **OpenGL|ES 표준화 방향**
  - **OpenGL|ES 1.1**
  - **OpenGL|ES 2.0**
- 하드웨어 가속칩 현황

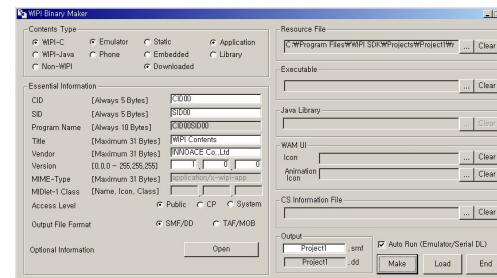
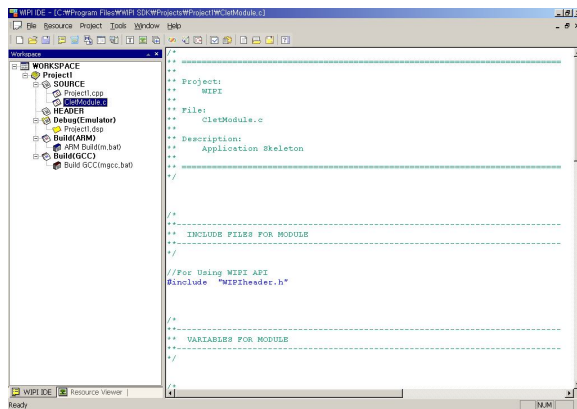
## 국내 WIPI 환경

- **국내 WIPI-C 환경**
  - **SKT와 KTF/LGT WIPI** 두 가지
  - 콘텐츠 바이너리 호환 안됨
  - 개발 환경 및 에뮬레이터가 다름
  - 이동통신사별 추가 **API** 존재
- **공통점**
  - 동일한 **WIPI-C** 스펙 지원
  - 소스 레벨에서의 호환이 가능 함

# 국내 WIPI 환경

## ▶ SKT WIPI-C 환경

- **ARM** 컴파일러나 **GCC** 컴파일로 모두 사용 가능
- 잘 정리된 **IDE** 제공
- IDE에서 **VC6** 과 연결 가능
- 배치 파일과 **Makefile**을 통한 쉬운 컴파일 환경 제공
- 개인 컴퓨터에서 컴파일 하여 실행 파일 제작 가능



# 국내 WIPI 환경

## ▶ KTF/LGT WIPI-C 환경

- **GCC** 컴파일만 사용 가능
- 별도 **IDE** 환경 없음
- **VC6**를 기준으로 라이브러리 및 에뮬레이터 제공
- 배치 파일 제공 및 초기 컴파일 셋팅 필요
- 외부 컴파일 서버를 이용하여 최종 실행 파일 제작



# CONTENTS

- 국내 모바일 3D 게임
  - 모바일 게임 시장 분석
  - 게임폰 및 서비스
- 국내 WIPI-C 환경
  - SKT WIPI-C
  - KTF WIPI-C
- **WIPI기반의 3D 솔루션**
  - 개발 환경 및 툴
  - 개발 방법
- OpenGL|ES 표준화 방향
  - OpenGL|ES 1.1
  - OpenGL|ES 2.0
- 하드웨어 가속칩 현황

## WIPI 기반의 3D 솔루션

- ▶ **모바일 개발 플랫폼과 3D 표준**
  - **low-level or high-level**
  - **BREW → OpenGL|ES**
  - **Symbian (native) → OpenGL|ES**
  - **Java for GSM → JSR-184 / JSR-239**
  - **WinCE → D3DM, OpenGL|ES?**
  - **WIPI → OpenGL|ES ? JSR-184 ? D3DM?**

## WIPI 기반의 3D 솔루션

### ▶ 가바플러스 - NF3D™

- 특징

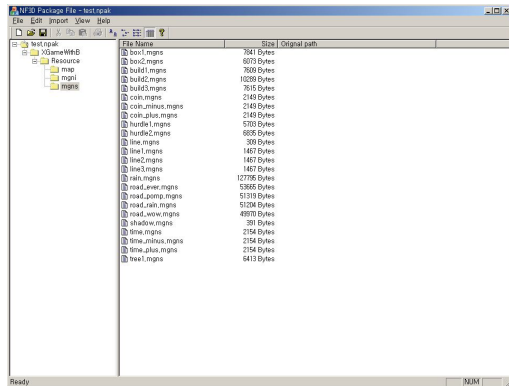
- 콘텐츠 호환성 고려
- **SKT, KTF, LGT WIPI용 SDK** 제공 및 서비스
- **JSR-184**에 그래픽 리소스 적용 가능
- **OpenGL|ES + high-level** 엔진
- **OpenGL|ES 1.0** 적합성 테스트 통과

- 개발툴

- **3dsmax® Exporter** 플러그인
- 애니메이션 툴 / 패키지 툴 / 맵 툴

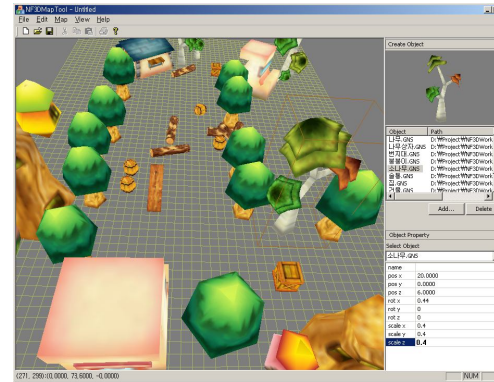
# WIPI 기반의 3D 솔루션

## ▶ 가바플러스 - NF3D™



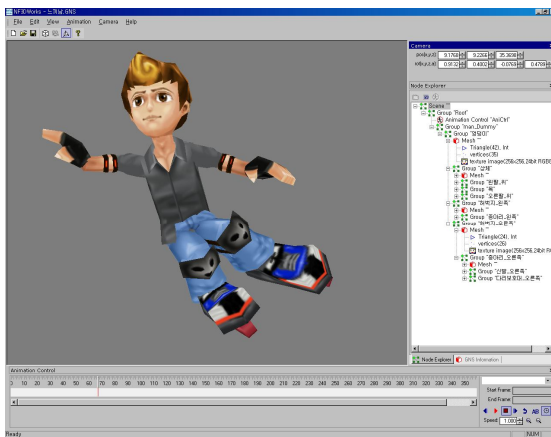
### 패키지 툴

여러 파일을 하나의 파일로 압축/저장

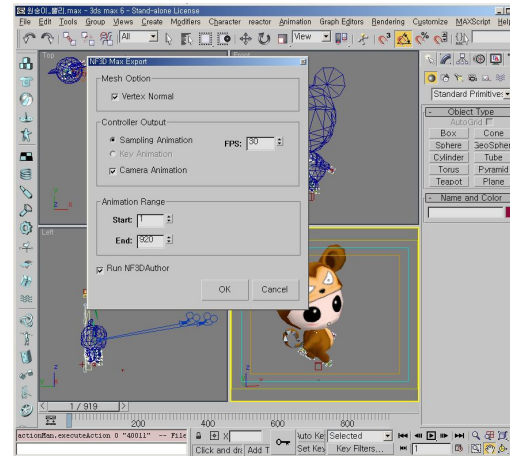


### 맵 툴

맵 정보 생성 및 관리



### 애니메이션 툴 애니메이션 최적화



### Exporter 플러그인 3D 데이터 추출

## WIPI 기반의 3D 솔루션

### ▶ 고미드 - G3D

- 특징

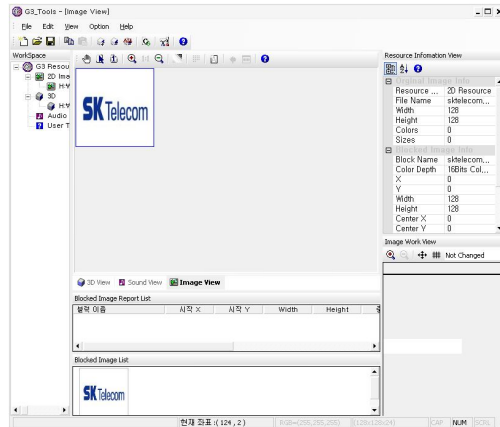
- 개발 플랫폼에 최적화
- **SKT WIPI용 SDK** 및 서비스
- **OpenGL|ES + high-level 엔진**

- 개발툴

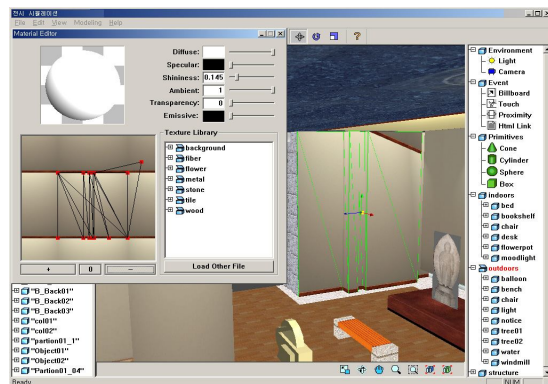
- **3dsmax® Exporter** 플러그인
- **Resource Editing Tool / Map Tool**

# WIPI 기반의 3D 솔루션

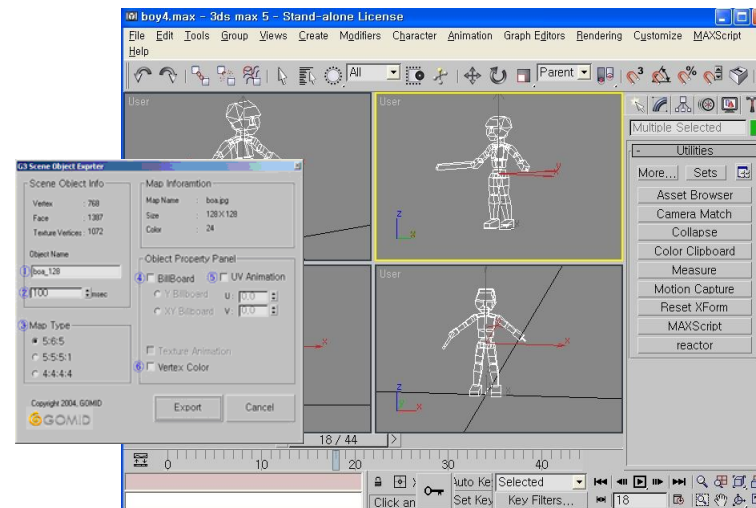
## ▶ 고미드 - G3D



Resource Editing Tool  
Sound / Bitmap / 3D 데이터



Map Tool  
World 관리 / 이벤트 편집



Exporter 플러그인  
3D 데이터 추출

## WIPI 기반의 3D 솔루션

### ▶ 리코시스 - M3D™

#### • 특징

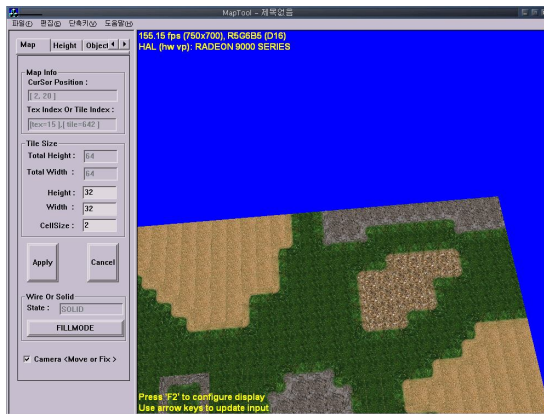
- **2.5D Indoor 엔진 및 3D 엔진**
- **SKT WIPI용 SDK 및 서비스**
- **OpenGL|ES + high-level 엔진**
- **OpenGL|ES 1.0 적합성 테스트 통과**

#### • 개발툴

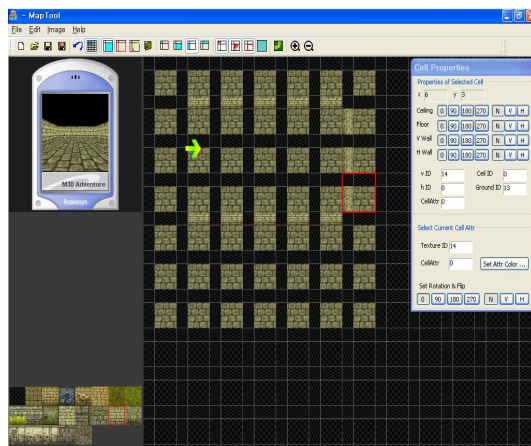
- **3dsmax® Exporter 플러그인**
- **World Editor / MMZ Tool(2.5D Indoor)**

# WIPI 기반의 3D 솔루션

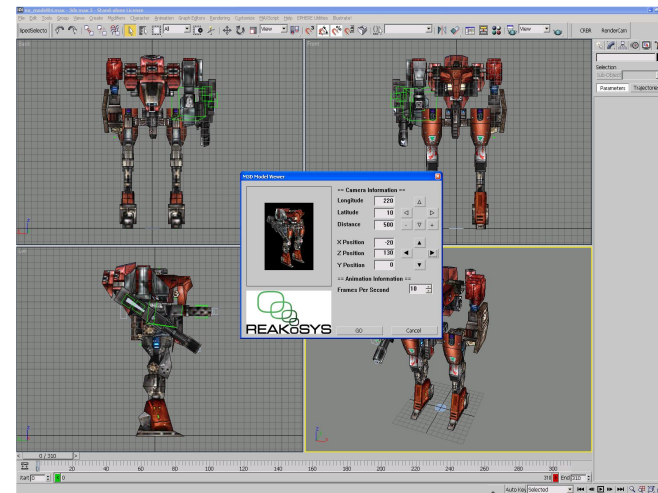
## ▶ 리코시스 - M3D™



World Editor  
3D 지형 제작



MMZ Tool  
Indoor Engine을 위한 맵툴



Exporter 플러그인  
3D 데이터 추출

## WIPI 기반의 3D 솔루션

### ▶ 신지소프트 – Swerve (Superscape)

- 특징

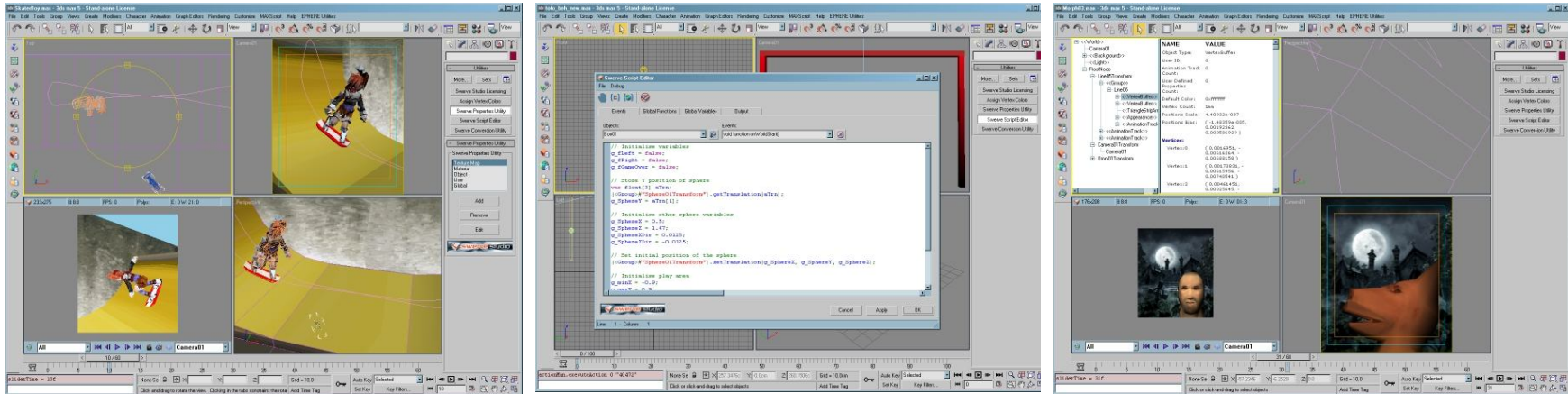
- JSR-184의 구조 적용
- GNEX SDK 를 이용하여 콘텐츠 제작
- SKT 콘텐츠 서비스
- OpenGL|ES + high-level 엔진

- 개발툴

- GNEX 3D Studio : 3dsmax® 통합 플러그인

# WIPI 기반의 3D 솔루션

## ▶ 신지소프트 – Swerve (Superscape)



### Gnex 3D Studio

Previewing, Behavior Scripting, Spy, Exporter,  
Plasma Developer, Stage Set Functionality

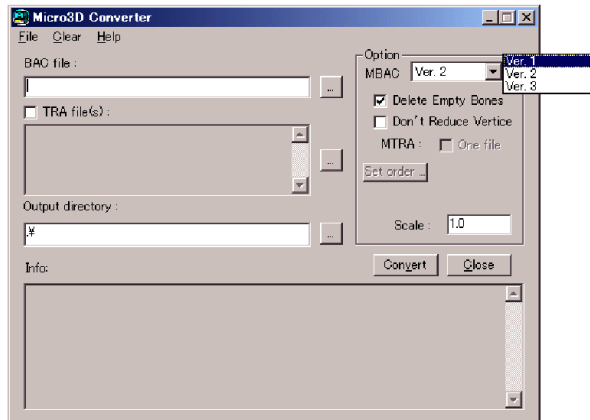
## WIPI 기반의 3D 솔루션

### ▶ HI Corp. – Mascot Capsule

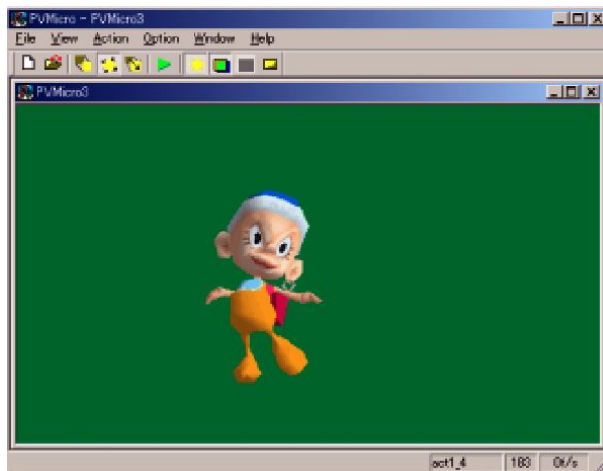
- 특징
  - 일본 콘텐츠 확보
  - **SKT, KTF WIPI용 SDK** 및 서비스
  - **OpenGL|ES + high-level 엔진**
- 개발툴
  - **3dsmax® Exporter** 플러그인
  - **PAC / Micro3DConv / PVMicro**

# WIPI 기반의 3D 솔루션

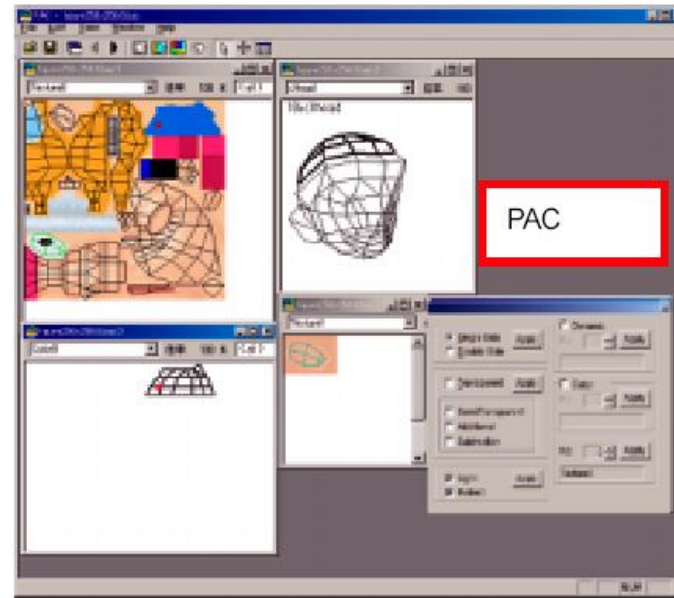
## ▶ HI Corp. – Mascot Capsule



Micro3DConv  
모바일용 데이터로 변환



PVMicro  
모바일 데이터 뷰어



PAC  
Polygon Authoring

## WIPI 기반의 3D 솔루션

### ▶ Fathammer – X-Forge

- 특징

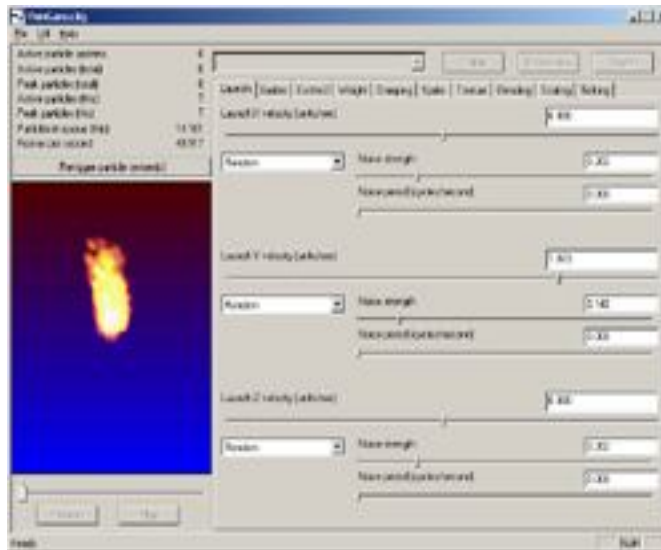
- 여러 컴포넌트 / C++ 언어만 지원
- 대작 게임 개발 지원
- **SKT WIPI용 SDK** 및 서비스
- **OpenGL|ES + high-level** 엔진

- 개발툴

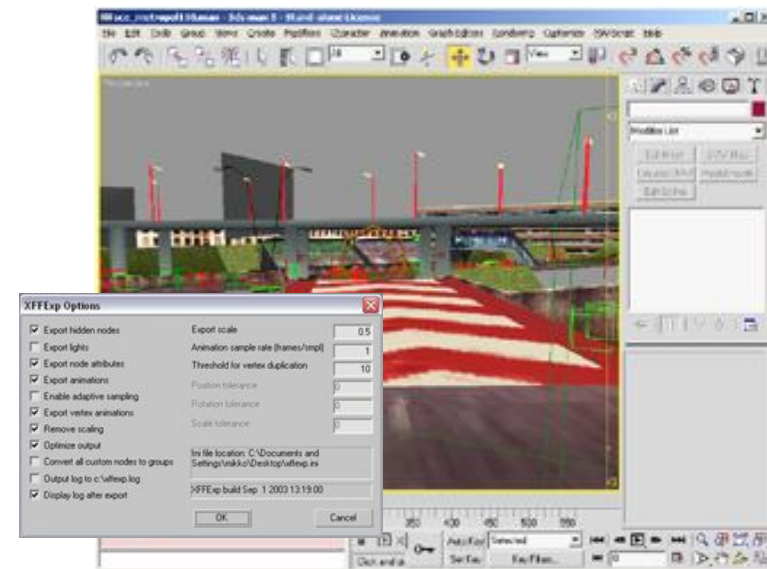
- **3dsmax® Exporter** 플러그인
- **Packager / Profile Analyzer / Particle FX Editor**

# WIPI 기반의 3D 솔루션

## ▶ Fathommer – X-Forge



Particle FX Editor

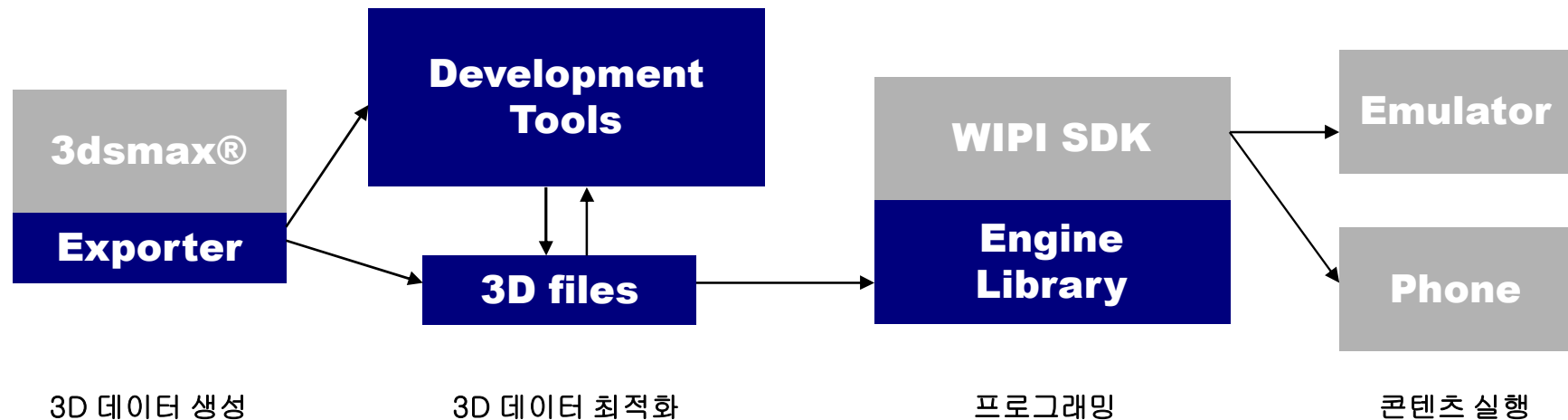


Exporter 플러그인  
3D 데이터 추출

## WIPI 기반의 3D 솔루션

### ▶ 3D 솔루션을 이용한 콘텐츠 개발 방법

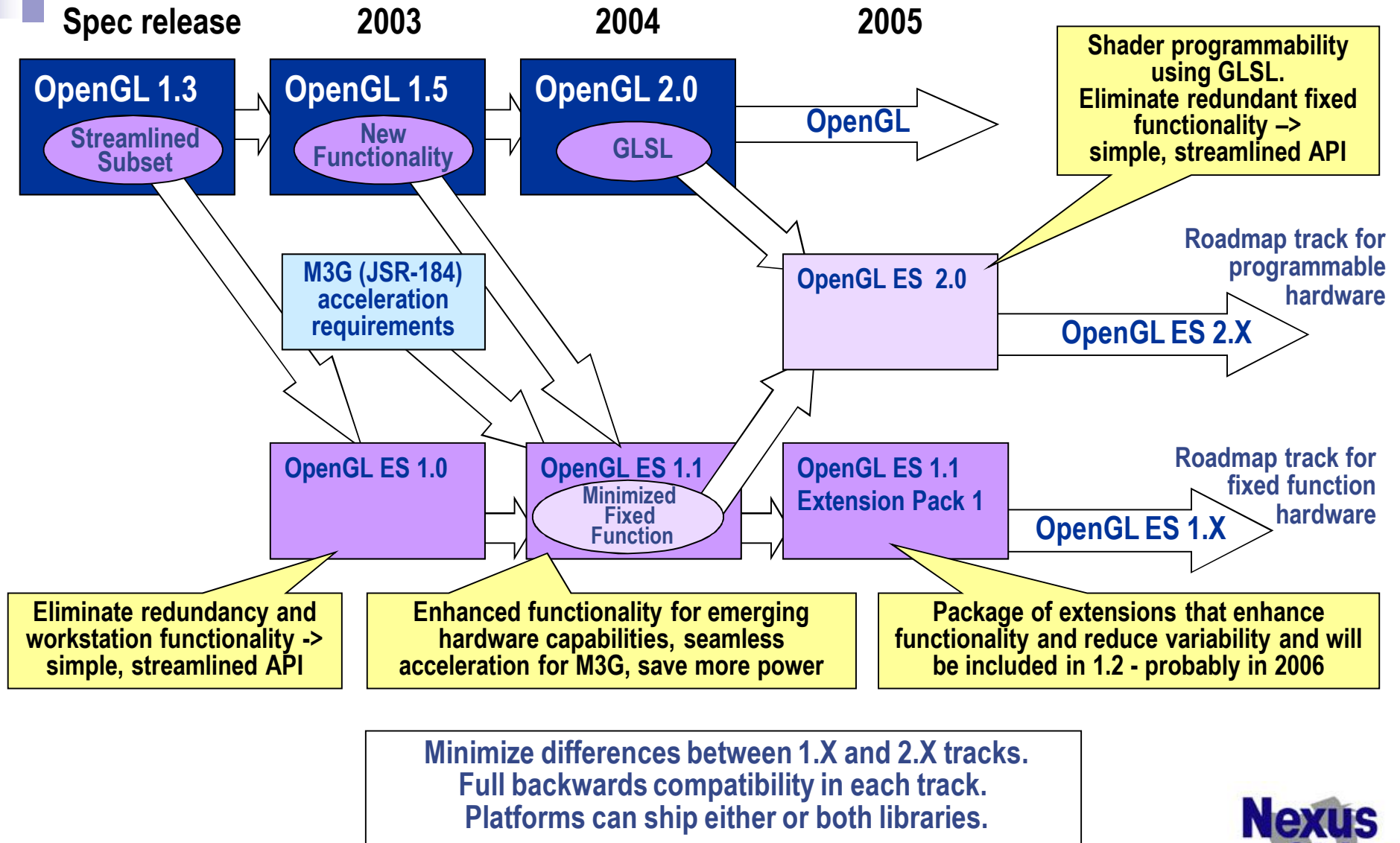
- 개발 환경을 위한 3D 솔루션 SDK 설치
  - SKT / KTF / LGT WIPI-C
  - 해당 WIPI-C 플랫폼용 3D 솔루션 설치



# CONTENTS

- 국내 모바일 3D 게임
  - 모바일 게임 시장 분석
  - 게임폰 및 서비스
- 국내 WIPI-C 환경
  - SKT WIPI-C
  - KTF WIPI-C
- WIPI기반의 3D 솔루션
  - 개발 환경 및 툴
  - 개발 방법
- **OpenGL|ES 표준화 방향**
  - **OpenGL|ES 1.1**
  - **OpenGL|ES 2.0**
- 하드웨어 가속칩 현황

# OpenGL/ES 표준화 방향



## OpenGL ES 1.1

- **Based on OpenGL 1.5 specification**
- **Buffer Objects**
  - Use to allocate, initialize and render from memory
  - Store vertex array and element index data
- **Auto mipmap Generation**
  - potentially accelerate auto mip-level generation
- **Enhanced texture processing**
  - Multi-textures and texture combiner functionality for effects
- **Vertex skinning functionality**
  - Allows smooth animation of complex figures and geometries

## OpenGL ES 1.1

- **User-defined clip planes**
  - Efficient early culling of non-visible polygons
- **Enhanced point sprites and point sprite arrays**
  - Efficient and realistic particle effects
- **Static and Dynamic state queries**
  - Layered software environment
- **Draw texture**
  - Fast rendering of background paintings, bitmapped font glyphs
  - 2D framing elements in games
- **New Core Additions and Profile Extensions**

# OpenGL ES 2.0 Philosophy

- **Based on GLSL as used in OpenGL 2.0**
  - Open standard
  - Proven on desktop
- **Optimized for use in Embedded devices**
  - Reduced shader program sizes
  - Reduced register usage
  - Reduced numeric precision
- **Pure programmable model**
  - Most fixed functionality removed.
- **Not 100% backward compatible with OpenGL ES 1.x**

# OpenGL Shading Language (GLSL)

- 'C' – like language
- **Types:**
  - void float vec2 vec3 vec4 mat2 mat3 mat4 int ivec2 ivec3 ivec4  
bool bvec2 bvec3 bvec4 samplerND
  - Structs, Arrays
- **Type Qualifiers**
  - uniform
    - 'Constants' (light position/direction, texture units, ...)
    - Does not change per vertex.
  - attribute
    - Per-vertex values (position, normal,...)
  - varying
    - Generated by vertex shader
    - Interpolated by the rasterizer to generate per pixel values
    - Used as inputs to Fragment Shader
    - e.g. texture coordinates

# Vertex Shader Functions

- **The vertex shader can do:**
  - Transformation of position using model-view and projection matrices
  - Transformation of normals, including renormalization
  - Texture coordinate generation and transformation
  - Per-vertex lighting
  - Calculation of values for lighting per pixel
- **The vertex shader cannot do:**
  - Anything that requires information from more than one vertex
  - Anything that depends on connectivity.
  - Any triangle operations (e.g. clipping, culling)
  - Access colour buffer

# Fragment Shader Functions

- **The fragment shader can do:**
  - Texture blending
  - Fog
  - Alpha testing
  - Dependent textures
  - Pixel discard
  - Bump and environment mapping
- **The fragment shader cannot do:**
  - Blending with colour buffer
  - ROP operations
  - Depth or stencil tests
  - Write depth

# CONTENTS

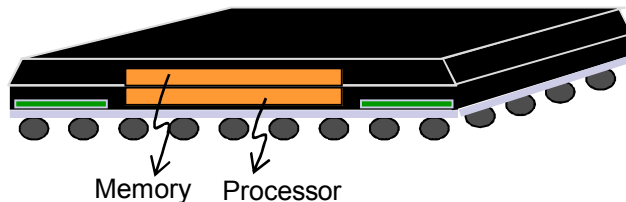
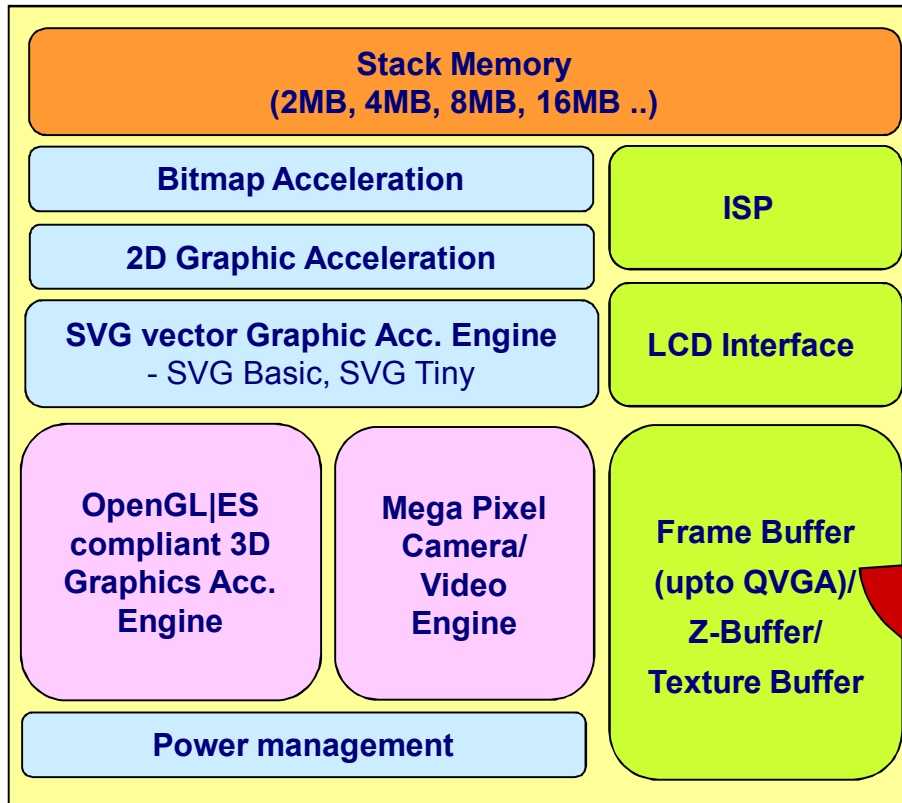
- **국내 모바일 3D 게임**
  - 모바일 게임 시장 분석
  - 게임폰 및 서비스
- **국내 WIPI-C 환경**
  - **SKT WIPI-C**
  - **KTF WIPI-C**
- **WIPI기반의 3D 솔루션**
  - 개발 환경 및 툴
  - 개발 방법
- **OpenGL|ES 표준화 방향**
  - **OpenGL|ES 1.1**
  - **OpenGL|ES 2.0**
- **하드웨어 가속칩 현황**

# 하드웨어 가속칩 현황

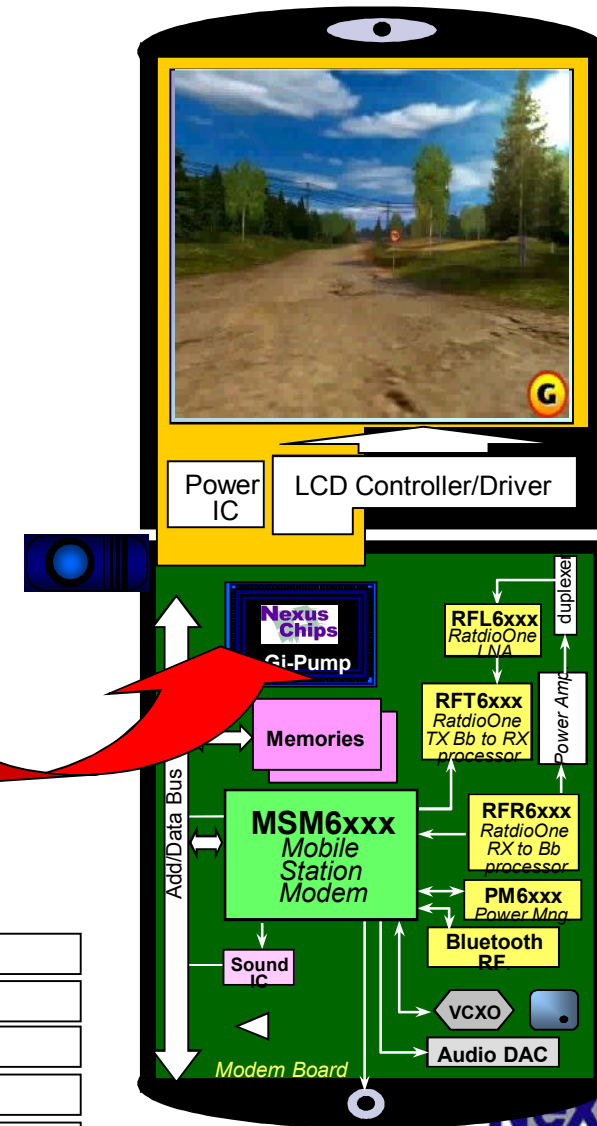
		<i>Solution 1</i>	<i>Solution 2</i>	<i>Solution 3</i>	<i>Solution 4</i>
		RISC CPU	DSP	Application Processor	H/W Graphics Chip
Vendor		3D S/W Engine Solution provider	ARM9, TI-DSP Nazomi	TI-OMAP SH-Mobile	Gi-Pump™
Architecture		ARM CPU	DSP	ARM CPU + DSP	Full H/W R/G Graphic engine
Performance	polygons / second	1,000 ~ 2,000	10,000 ~ 30,000	100,000 ~ 200,000	NX1004 : 1.3M ~ NX1005 : 4M~
	Frames / second	3 ~ 5	8 ~ 12	12 ~ 15	25 ~ 30
Operating Frequency		High	High	High	< 100MHz
Power Consumption		High	High	High	0.8mA/MHz
Price		N/A	Low	High	Low
Standard Graphics Format	OpenGL ES acceleration	No	Software Engine Required	Software Engine Required	Yes
	SVG acceleration	No			Yes
Local Memory(Stack)		No	No	No	Yes

# 하드웨어 가속칩 현황

- 3D Graphics Acceleration Engine embedded
- Mega pixel camera / Video Engine embedded



Package Type	TABGA/SCSP
Pin Number	100 ~200
Body Size	8x8 ~ 12x12
Height	1.15 ~ 1.5
Memory	Stacked/Extensible



Nexus Chips

**감사합니다.**

**김성재**  
[mid@nexuschips.com](mailto:mid@nexuschips.com)

**Nexus  
Chips**