



게임 프로그래밍 과정

afewhee@gmail.com





- Game Software Developing Flow

- ◆ V Model

- ◆ 계획/분석/설계/테스트/배포/유지보수

- ◆ Programming Skill

- Game Components

- ◆ Graphics/Input/Media/Font/Network/Data

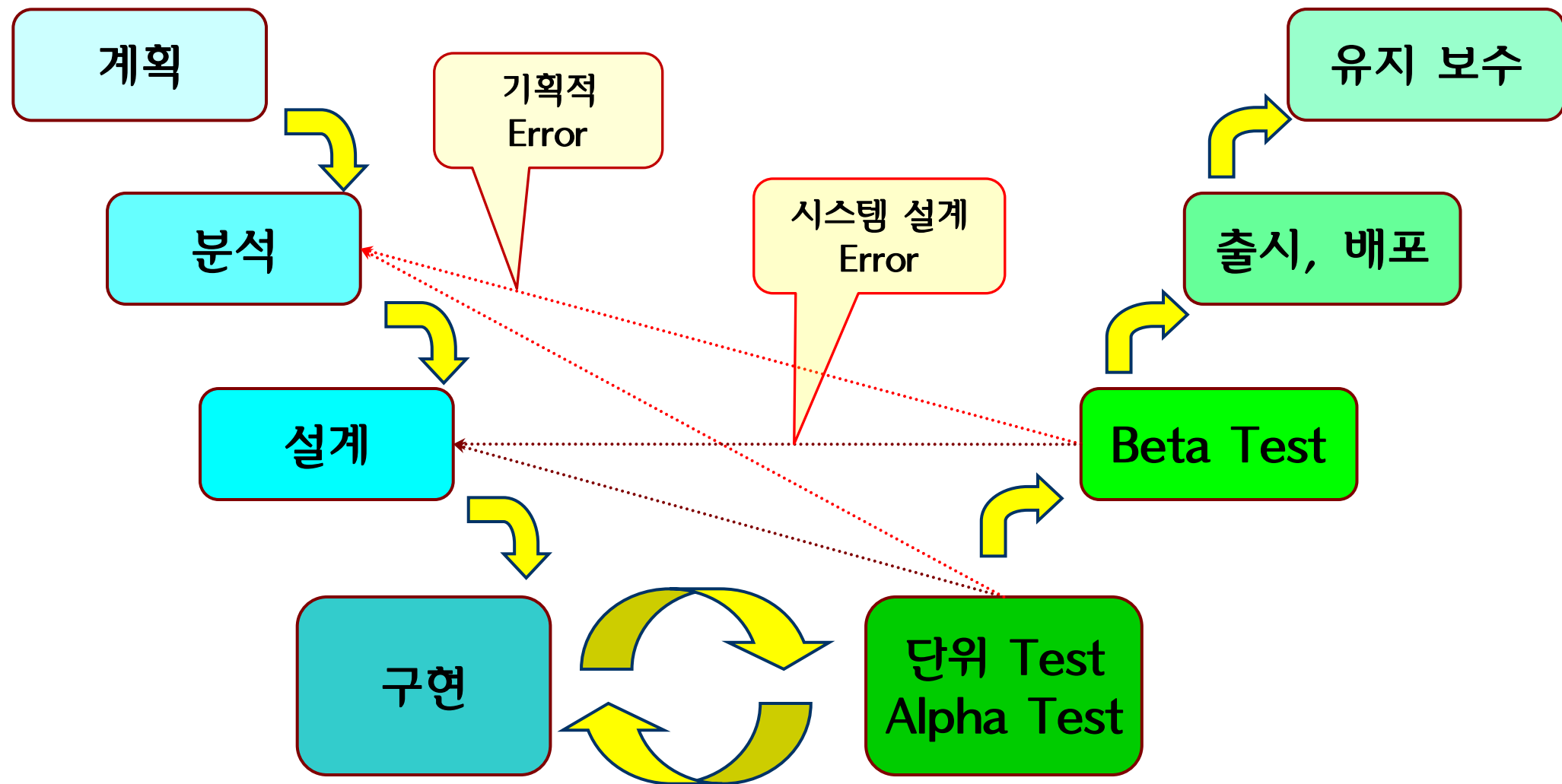




PART 1. Software Developing Flow



1.2 Software Developing Flow(V Model)





● 계획

- ◆ 개발 내용 계획: 기술적인 이슈와 개발 업무 계획
- ◆ 비용 계획: 시간, 인원 등의 소요 비용 산정
- ◆ 조직 계획: 개발 팀 구성 및 역할 배정

- ◆ 제안서 작성





● 분석

- ◆ Domain 분석: 객체 정의 → 관계 설정 → 추상화
- ◆ Play 분석: 실행 흐름 작성
- ◆ 그래픽 Concept → Game 데이터 설정

- ◆ 상세 기획 작성





● 설계

- ◆ 추상화 분석 → 구체화 설계

● 게임 시스템, 자료구조 설계

- ◆ 공통 모듈: 툴, 콘텐츠 등에서 공통으로 사용할 모듈

- ◆ 렌더링 머신

- ◆ 장면 관리자

- ◆ 네트워크 입출력

- ◆ Database 설계

- ◆ 입력장치, 미디어 플레이 인터페이스 설계





- 게임 엔진 설계

- ◆ 지형 엔진, 물리 엔진
- ◆ 그래픽 플러그인, 캐릭터 애니메이션
- ◆ GUI, 스크립트





- 툴 프로그램 구현

- ◆ 리소스 관리 툴
- ◆ 지형 툴
- ◆ 캐릭터 툴
- ◆ Effect, GUI 툴





● 콘텐츠 구현

- ◆ 네트워크 입출력 데이터 형식 결정
- ◆ 게임 플레이 위상 구현
- ◆ 렌더링 pass 구현
- ◆ AI 구현
- ◆ 엔진 모듈 통합 → 콘텐츠 구현





● 테스트

- ◆ 기능(Play) 중심 알파 테스트
- ◆ 그래픽 Quality, 성능 중심의 클라이언트 알파 테스트
- ◆ 네트워크 I/O 중심의 클라이언트/서버 베타 테스트





- 배포, 유지보수

- ◆ 버그 패치

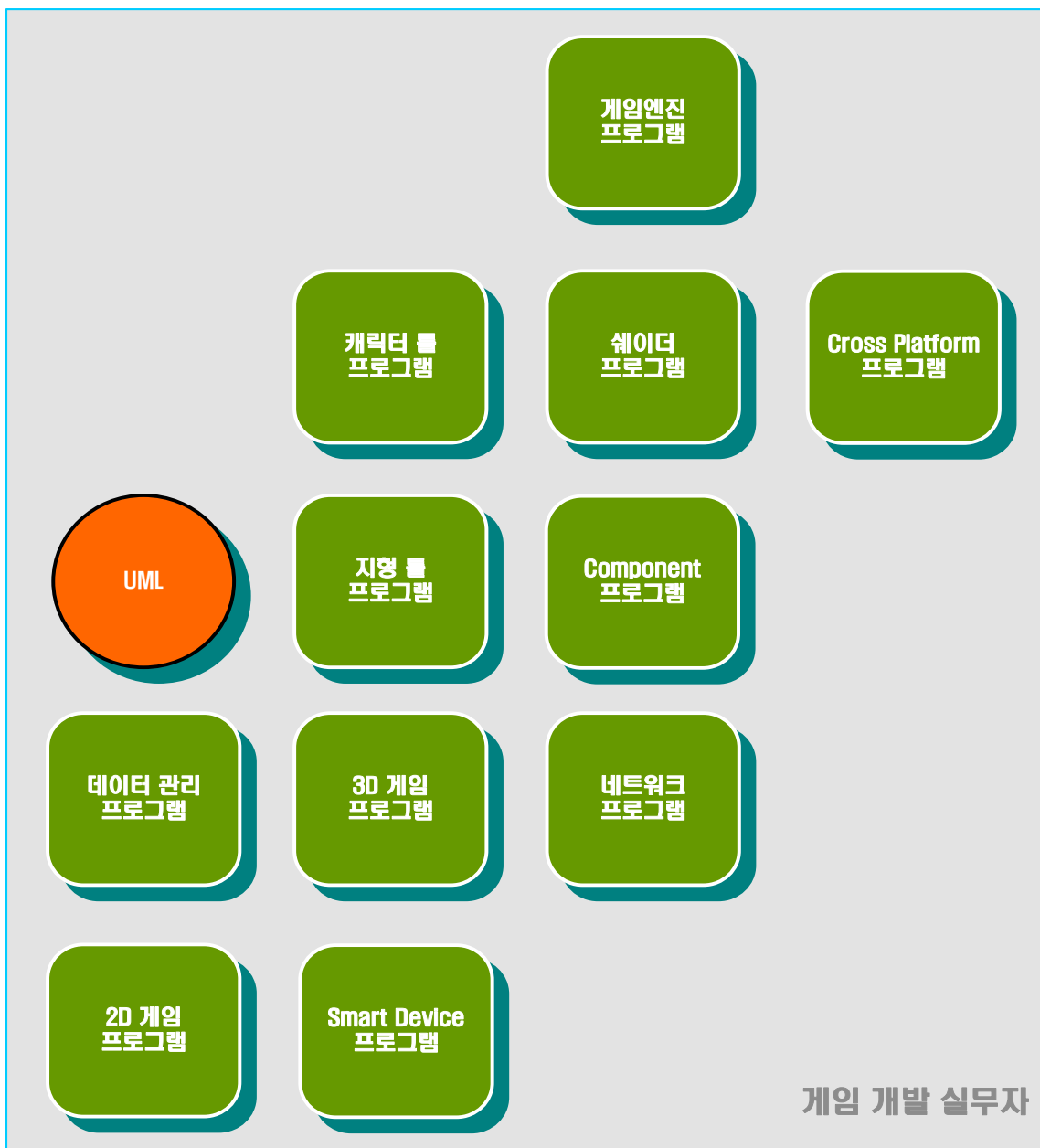
- ◆ 기능 강화

- ◆ 보안 강화



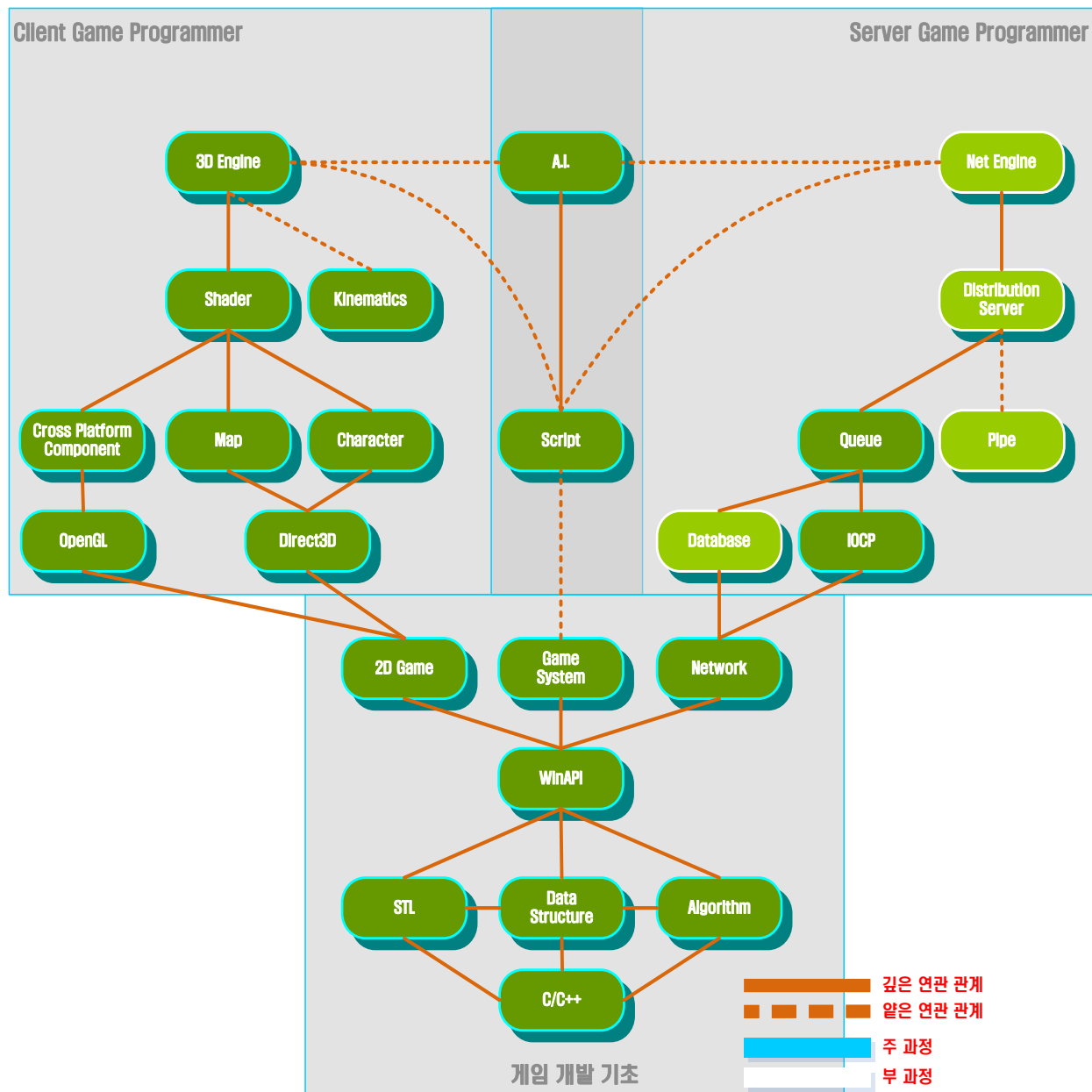


1.3 Programming Skill





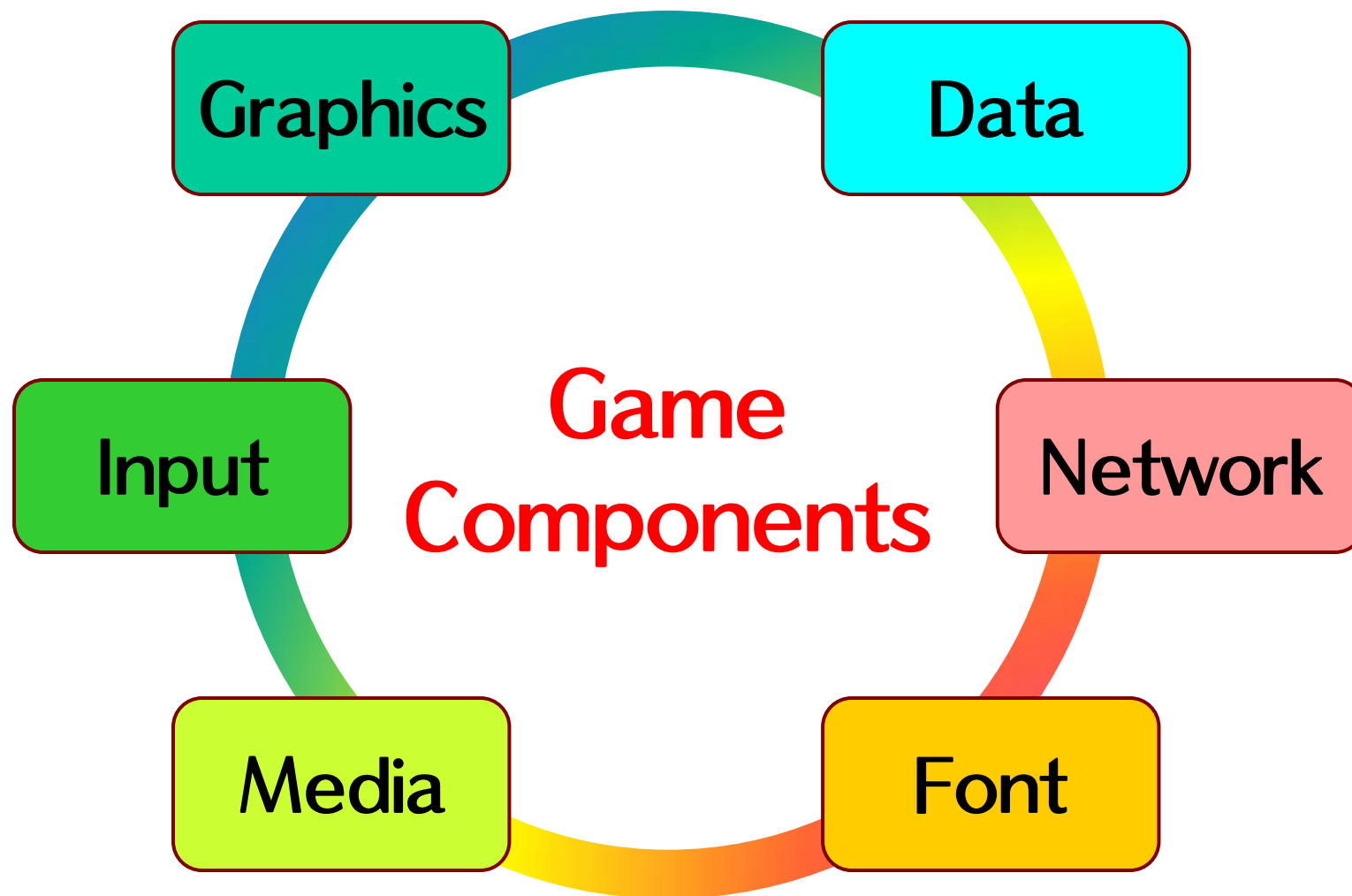
1.3 Programming Skill





PART 2. Game Components







● Graphic Library

◆ Soft or Direct device

- Video Buffer: 3D Rendering Pipe Line 이 없는 시스템

◆ OpenGL

- Workstation, PC, Embedded에 두루 사용
- 2D 렌더링 객체 없음 → 3D 사용

◆ Direct3D

- Microsoft 시스템에서만 가능
- 2D/3D 등 게임 제작에 필요한 대부분의 기능이 포함되어 있음





● 2D

◆ Image

- Digital Pixel Data

◆ Texture

- Image → Pixel Sampling

◆ 2D 모델

- Sprite 객체
- 관절체, 상/하 분리 모델

◆ GUI(Graphic User Interface)

- 시스템에서 지원되는 GUI는 범용이라 속도가 중요한 게임에 부적합 → 자체 또는 엔진에서 제공





● 3D

◆ 3D Mesh

- Map(Indoor/Outdoor)
- 3D Model
- Particle Effect ...

◆ Vertex Processing

- Transform & Lighting
- Fog
- Rasterization

◆ Pixel Processing

- Multi-Texturing
- Test: Alpha, Depth, Stencil
- Alpha blending





- Keyboard

- ◆ Keyboard
- ◆ Keypad, 버튼
- ◆ Library: DirectInput, API 함수

- Mouse

- ◆ Mouse
- ◆ Multi touch screen
- ◆ Library: DirectInput, API 함수

- Joystick

- ◆ Library: DirectInput, API 함수





● Sound

- ◆ Analog signal → Digitize
- ◆ Mono/Stereo
- ◆ 3D sound
- ◆ Library: DirectSound, OpenAL, OpenSL, OpenSSL

● Video

- ◆ Library: DirectShow, API 함수

● Music

- ◆ Direct Music: 연주를 위한 악보 개념의 모듈이었으나 구현이 Sound로 통합





- 2D Font

- ◆ 2D String 출력
- ◆ Library: D3DX, API, FreeTypeFont

- 3D Font

- ◆ String → Vertex Buffer 변환 → 3D 출력
- ◆ Library: D3DX, API 함수





● I/O Model

- ◆ Windows: AsyncSelect, EventSelect, IOCP
- ◆ Linux, Unix: Epoll, Kqueue
- ◆ BSD: Select
- ◆ Library: Winsock, API

● Database

- ◆ 관계형 데이터 베이스
- ◆ 객체지향형 데이터 베이스
- ◆ XML
- ◆ 종류: Oracle, MSSQL, MYSQL, Embedded용 SQL
- ◆ Library: ODBC, OLEDB, API





● Distributed Server Model

◆ 기능 분산

- 패치, 인증, 콘텐츠 서버

◆ 데이터 분산

- Zone, Room, DB 서버

◆ 공간 분산

- Server Group(서버 군) 구성





● Game Play Data

- ◆ Game Play와 직접 연관있는 데이터
- ◆ 반드시 실행
- ◆ 네트워크 입출력 데이터, Game Level, AI, UI

● Rendering Data

- ◆ 연출 데이터
- ◆ 필요에 따라 실행이 안될 수 있음
- ◆ Particle, Shader, ...

