



3D게임프로그래밍

-CHAPTER4-

SOULSEEK



목차

1. 계층 구조(Hierarchy)



계층 구조(**HIERARCHY**)

1. 계층 구조(HIERARCHY)

트리

- 자료구조 중에서 Tree로 구현 가능하다.
- 3D 그래픽스에서 가장 애용되는 자료구조이다.
- 내부 지형 처리 - 이진 트리
- 거대 지형 처리 - 쿼드 트리
- 거대 공간 처리 - 옥 트리

계층 구조의 구현

변환을 나타내는 공식

- $V_{world} = V_{local} * TM$

정점들로 이루어진 육면체가 계층 구조로 이루어져 있다면

- $V_{world} = V_{local} * TM_{child} * TM_{parent}$

1. 계층 구조(HIERARCHY)

부모 메시는 원점에 있고 **Y**축 회전, 자식 메시는 **Z**축 회전을 하고 부모로부터 **(3, 3, 3)**의 거리에 있다고 했을 때,

부모 메시

- 기본 변환 : TM_{parent} , **Y**축 회전 : R_{parent}

자식 메시

- 기본 변환 : TM_{child} , **Y**축 회전 : R_{child} , 이동 변환 : R_{child}

부모의 최종 행렬

- $M_{parent} = R_{parent} * TM_{parent}$

자식의 최종 행렬

- $M_{child} = R_{child} * TM_{child} * M_{parent}(R_{parent} * TM_{parent})$

1. 계층 구조(HIERARCHY)

변환 준비

```
D3DXMATRIXA16 g_matTMParent; //부모의 기본 행렬  
D3DXMATRIXA16 g_matRParent; //부모의 회전 행렬
```

```
D3DXMATRIXA16 g_matTMChild; //자식의 기본 행렬  
D3DXMATRIXA16 g_matRChild; //자식의 회전 행렬
```

변환을 적용

```
D3DXMatrixIdentity(&g_matTMParent); //부모의 메시는 원점에 있으므로 TM은 단위 행렬
```

```
D3DXMatrixRotationY(&g_matRParent, GetTickCount() / 500.0f); //부모 메시의 Y축 회전 행렬
```

```
D3DXMatrixRotationZ(&g_matRChild, GetTickCount() / 500.0f); //자식 메시의 z축 회전 행렬
```

```
D3DXMatrixTranslation(&g_matTMChild, 3, 3, 3); //자식 메시는 원점으로 부터 (3, 3, 3)의 거리에 있다.
```

```
//특정 vector를 축으로 하는 행렬
```

```
D3DXMatrixRotationAxis(&g_matRChild, nomalVec, Angle);
```

1. 계층 구조(HIERARCHY)

준비된 변환을 부모, 자식 객체에 적용해서 그린다

//부모의 변환을 만든다 - 회전 * 기본 변환

matWorld = g_matRParent * g_matTMParent;

//적용된 변환을 기반으로 부모 객체를 그린다.

DrawMesh(&matWorld);

//자식의 변환을 만든다 - 자신의 회전 * 자신의 기본 변환 * 부모의 회전 * 부모의 기본변환

matWorld = g_matRChild * g_matTMChild * g_matRParent * g_matTMParent;

//matWorld = g_matRChild * g_matTMChild * matWorld;

//적용된 변환을 기반으로 자식 객체를 그린다.

DrawMesh(&matWorld);

1. 계층 구조(HIERARCHY)