

# Interactive TV 3D User Interface

1, 2, 3, 4  
1 2 3 4

{ninei<sup>1</sup>, sunny<sup>2</sup>, jjsky33<sup>3</sup>, moon<sup>4</sup>}@sogang.ac.kr

## An Implementation of 3D User Interface for Interactive TV Application

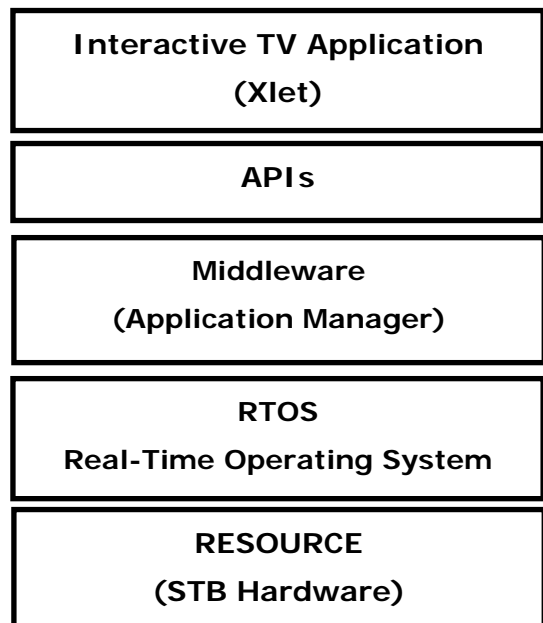
Jung Hwan Kim<sup>1</sup>, Youn Sun Park<sup>2</sup>, Chan Goo Jung<sup>3</sup>, Moon Ryul Jung<sup>4</sup>  
GSMC of Sogang Univ.<sup>1 2 3 4</sup>

(Xlet)  
UI  
2  
3 가  
UI  
가 3D System, Virtual Timeline, Animation Model  
3D System UI 가 3 가  
Timeline Animation Model STB  
UI 가  
3D UI TV 가

Keyword : 3D, UI, Interactive TV, Application

interactive TV 2  
UI 가 2  
STB(Set-Top Box) UI (depth),  
(Xlet), 3 UI  
MPEG-2 TS[7] 3  
STB STB , STB JVM,  
EPG(Electronic Program Guide), T-Game, T-Banking 3 UI  
가

DVB-MHP[3], OCAP[4], ACAP[5] Service MPEG-2  
 , 3 가 , Xlet  
 STB JVM(Java Virtual Machine) Java 3D [1]  
 API OpenGL, DirectX STB 가 .  
 RTOS(Real-Time OS), (Middleware) [3] Service 가  
 RTOS , (Virtual Channel) [4][5]  
 (Xlet) 가  
 가 . STB , 가 .  
 가 ,  
 3 2. STB Xlet  
 가 , interactive TV MPEG-2 TS Data/Object carousel [6][8][11]  
 3 User Interface Xlet STB  
 , 3 UI . Xlet STB  
 가 (Middleware : STB RTOS )  
 3D Graphic Engine [2] . 3D Engine )  
 3 UI 가 3 API ,  
 , Xlet (Middleware)  
 10kbyte 가 가 (Application Manager)  
 Downloadable 3D Graphic API .



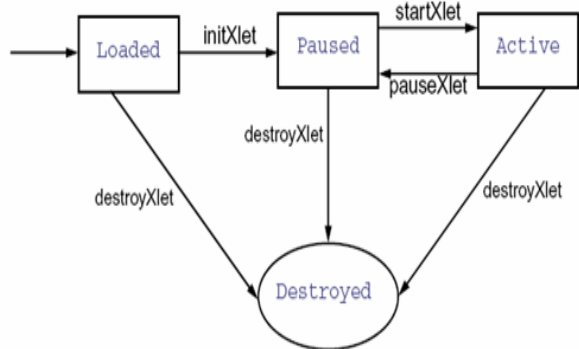
3D UI 가 (Xlet)  
 1. (Xlet)  
 MPEG-2 TS  
 STB(Set-Top Box)  
 Xlet . Xlet  
 Applet  
 .  
 Xlet (Service:  
 )

1. STB(Set-Top Box) Xlet

1 가 ,  
 Resource STB  
 , Tuner, De-Multiplexer,  
 Video/Audio Decoder . 가 3  
 RTOS Real-Time Operating System STB  
 가 . PC  
 (Windows) . Middleware  
 , Xlet ,

API .

3. Xlet (Life Cycle)  
 Xlet  
 ,  
 2. Xlet . Xlet  
 4 가 (State) 가 ,  
 2 Loaded, Paused, Active, Destroyed  
 가  
 initxlet(),  
 startXlet(), pauseXlet(), destroyXlet()  
 method Xlet .



2. Xlet life cycle [3]

Xlet  
 T-Learning, T-Government, T-Banking

2D

UI 가 .

**. Virtual 3D System**

3D UI TV

가 3 .  
 Graphics Reference  
 Model , 가 Z  
 3  
 . 3D Virtual System UI  
 3D Graphic Engine

1. Graphics Reference Model

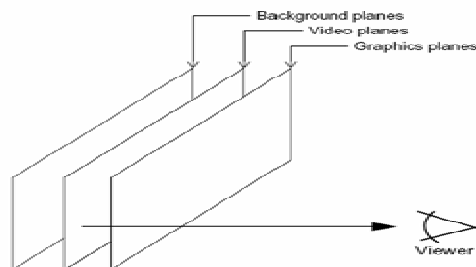
DVB-MHP 3 plane  
 Graphics Reference Model( 3)[3]

. Graphics Reference Model  
 Background planes, Video planes, Graphics planes  
 , plane  
 . plane layer

Background plane MPEG-2 TS[7] i-frame

가 가 .  
 Background plane i-frame STB  
 decoder .

가 ,  
 STB STB 가 가 .  
 i-frame 가  
 MPEG-2 TS i-frame  
 , Xlet  
 가 .



3. MHP Graphics Reference Model[3]

Video plane MPEG-2 TS 가 4 가 3D  
 decoding Background plane . 3D Graphics  
 plane . API Engine[2] Polygon(  
 가 , 가  
 . )  
 Graphics plane , Rendering . 3D Graphics  
 Xlet , Engine WCS(World Coordinate System),  
 , . 3D UI CCS(Camera Coordinate System), LCS(Local  
 Graphics planes Coordinate System) 가 3D  
 , i-frame, . 5 rendering .

2. 3 가

Graphics Reference Model

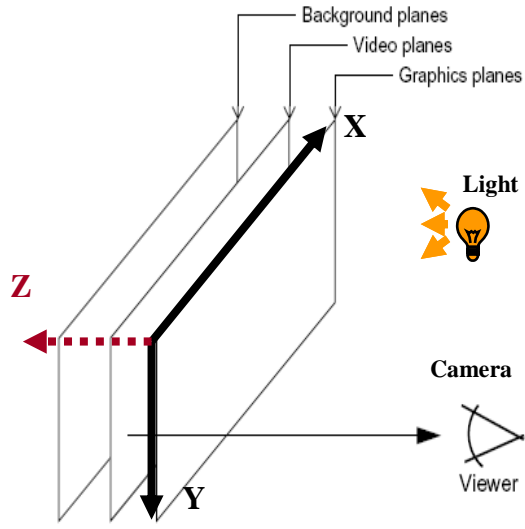
2 가 . 3 가  
 , , 가  
 z (X,  
 Y, Z)  
 4 3D UI 가  
 3 Graphics planes  
 . 4 3D UI 가  
 3D . Graphics planes Z ,  
 3 Light, Camera



5. 3D Rendering Object example

3D Graphic Engine 10kbyte  
 Xlet 가 , 가  
 Downloadable 3D Engine . Xlet  
 API

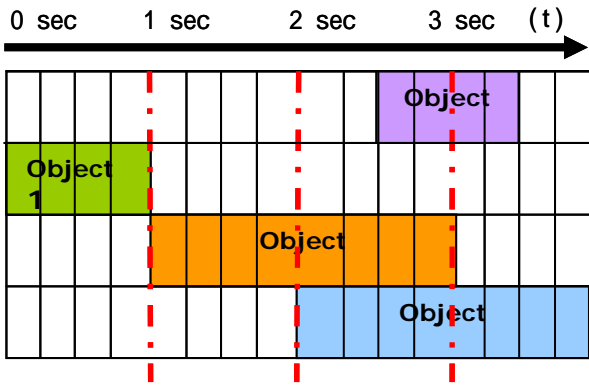
[2]



4. Virtual Coordinate System

. Virtual Timeline

1. 가  
 .  
 가 3  
 가 3D UI  
 3  
 2



6. Virtual Timeline

Virtual Timeline

가

Timeline API  
 getTimeLine() API  
 setFrameRate(),  
 가 가

2. 3D UI Timeline

3D UI 가 STB

Graphics Reference Model( 3)

Graphics planes 가

3 Video planes STB Decoder 가

MPEG-2 28~30

planes, Graphics

planes STB

Video planes

가 Virtual

Timeline 30

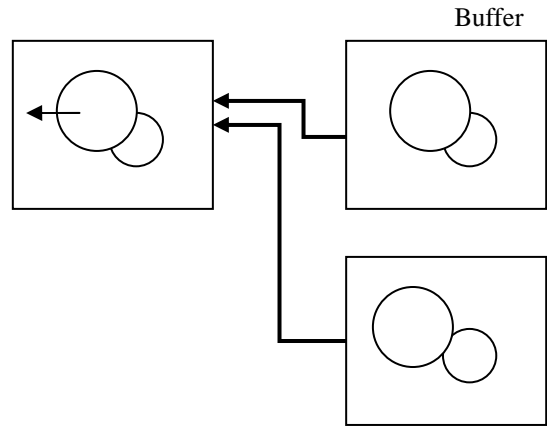
STB

가

STB (Flickering)

(Double Buffering)

7



7. Double Buffering Example

, STB

, Virtual Timeline

**. UI Animation Model**

Model

STB 가

UI STB

1. Smooth Moving Model

Smooth Moving Model 3 가

smooth

3 가

3D

3D

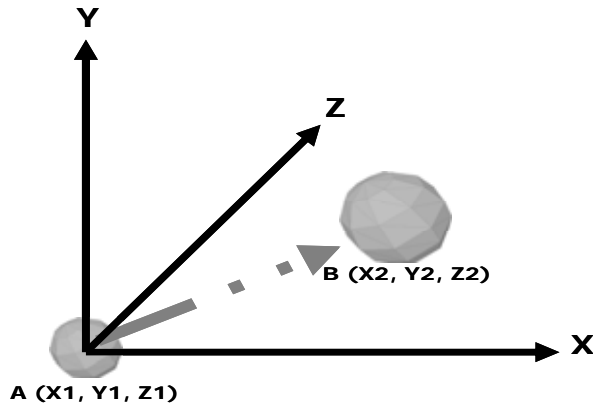
(Vertex) A(x1, y1, z1)

B(x2, y2, z2)

Vertex

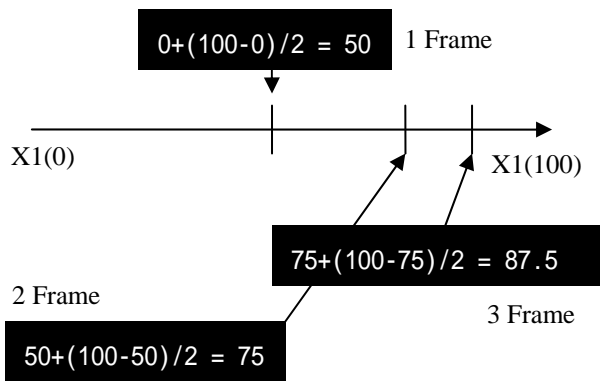
3D UI

Smooth Moving Model



### 8. 3D Object Moving Example

8 A 가 B A X1  
 X2-X1 , X1+X2-X1  
 B X2 , 3  
 X1=0, X2=100 가 , smooth 2  
 $X' = X1 + (X2 - X1) / 2$



### 9. smooth Moving

Y, Z , A  
 B ,  
 3D  
 UI 3 Smooth

### 2. Spring Model

Spring Model Smooth Moving Model  
 Spring 가  
 , X1 , X2 . X0  
 Spring Model

$$X1 = a * (X1 - X2) + b * (X0 - X2) + X2$$

(a: -2 < a < 2, b: -1 < b < 0)

```
int tempX = x1;
int tempY = y1;
x1 = (int)(0.8*(x1-x2)-0.7*(x0-x2)+x2);
y1 = (int)(0.8*(y1-x2)-0.7*(y0-y2)+y2);
x0 = tempX;
y0 = tempY;
```

### 3. Gravity Model

Smooth Moving Model Spring Model  
 가  
 가 Spring Model  
 , X2 , X1 가  
 (X1)  
 , Spring Model (a, b)

4. Vibration Model

Vibration Model

sign, cosign . sign  
 가 X, Y , Y  
 0 , -1 1  
 가 . Y  
 . 10 -10 10

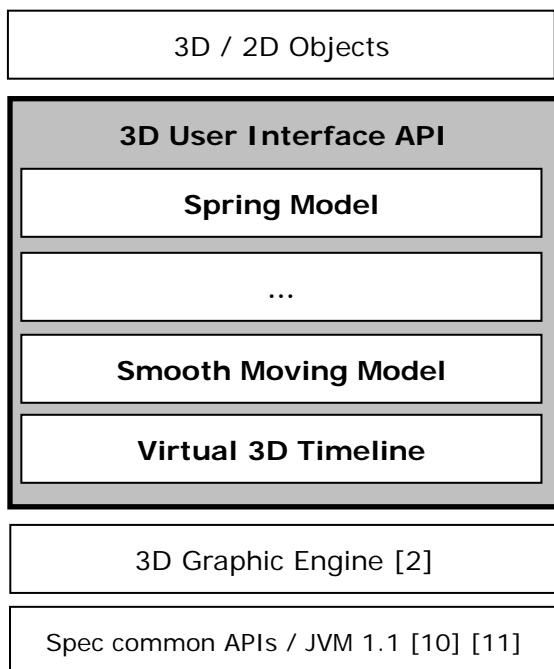
Vibration Model STB

JDK 1.1 Math  
 (Radian)

**3D User Interface API**

3D User Interface

10) STB APIs  
 API [2] JVM 1.1



10. 3D UI API

API 가 . 3D UI APIs Virtual 3D Timeline,

. Xlet 2D/3D Object UI .

3D UI STB

( . UI Animation Model)

Graphics drawing method(drawPolygon(), fillRect() )

, STB

Animation Model

(fps) Virtual Timeline

, 4-8 fps, drawing method 18-24 fps 가

(OCAP - HUMAX STB)

UI . Z Spring Model , 24 fps(frame per sec)



11. 3D UI Example (Z-Spring Model , 24 fps)

3D User Interface

2

Graphic Engine 3D user Interface 3D , 가

STB 3  
 UI . 3D  
 UI ,  
 가  
 .

[1] , . “ ” . 9  
 1 . 2004.  
 [2] , . “ Interactive TV  
 Polygon 3D Graphic Engine ” .  
 . 2005.  
 [3] ETSI TS 102 812 V1.2.1 : Digital Video  
 Broadcasting (DVB); Multimedia Home Platform (MHP)  
 Specification 1.1.1, 2003-06.  
 [4] OC-SP-OCAP1.0-I13-041215, OCAP 1.0 Profile,  
 2004.  
 [5] ATSC Candidate Standard: Advanced Common  
 Application Platform (ACAP); Advanced Television  
 Systems Committee, 2004-02.  
 [6] TR 101 202 Digital Video Broadcasting (DVB);  
 Implementation guidelines for Data Broadcasting. 1999.  
 [7] ISO/IEC 13818-1 Generic Coding of Moving Picture  
 and Associated Audio : Systems  
 [8] ISO/IEC 13818-6 Generic Coding of Moving Picture  
 and Associated Audio : Digital Storage Media Command  
 and Control. 1996.  
 [9] Steven Morris, Anthony Smith-Chaigneau.  
 “Interactive TV Standards.”Focal Press. 2005.  
 [10] BartCalder , Java TV API Technical Overview :  
 The JavaTV API Whitepaper ver 1.0, 2000.  
 [11] Richard S. Chernock, "Data Broadcasting:  
 Understanding the ATSC Data Broadcasting Standard",  
 McGrawHill, 2001.