Dual Scripting in a Virtual Reality Engine
Embedding Python in XVR

Emanuele Ruffaldi

pit@sssup.it

PERCRO

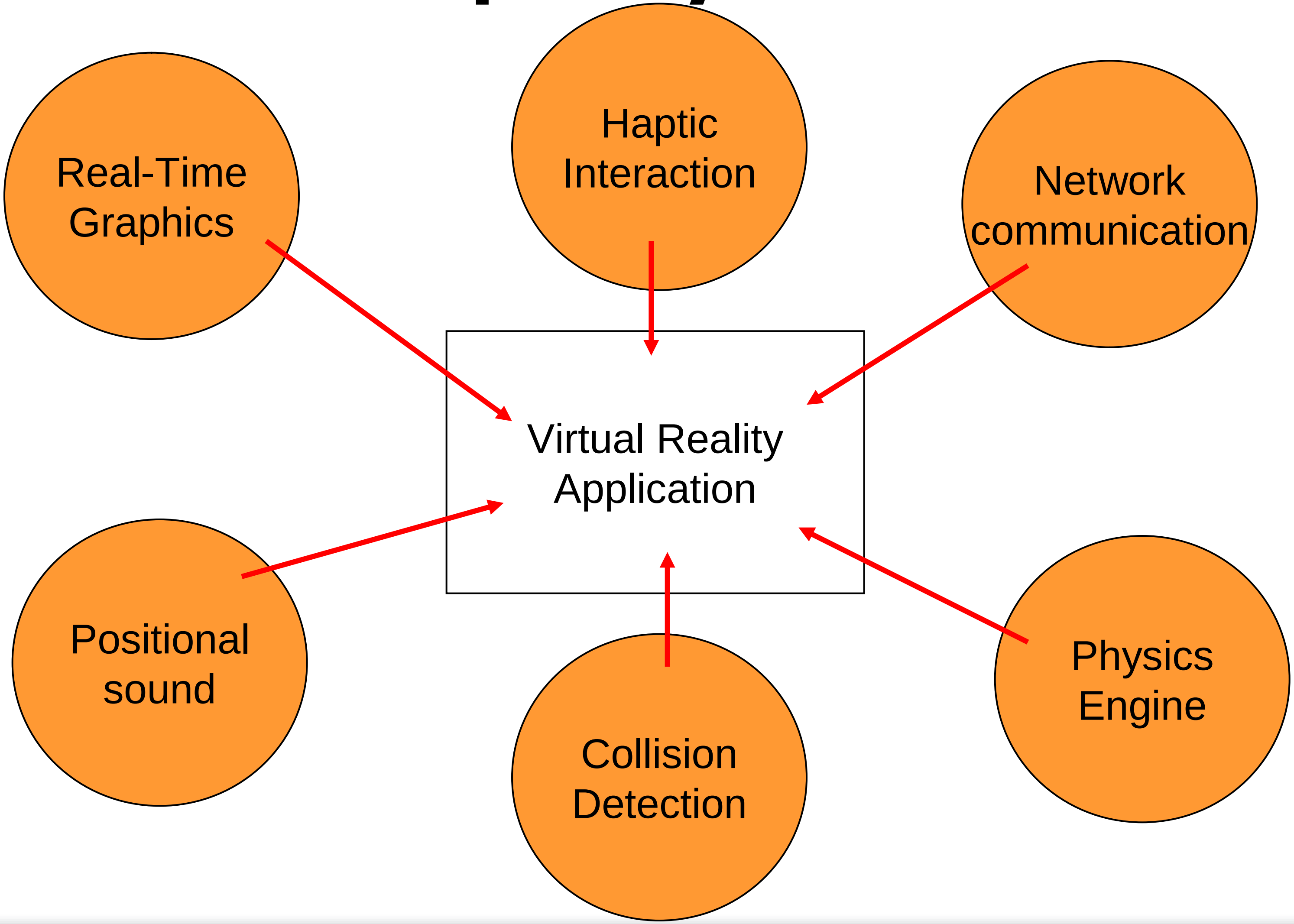
Simultaneous Presence, Telepresence and Virtual Presence

XVR
XTREME VR

VR installations can be complex systems

Integration is hard: many aspects to tackle, many subtle details easy to overlook

The complexity is intrinsic



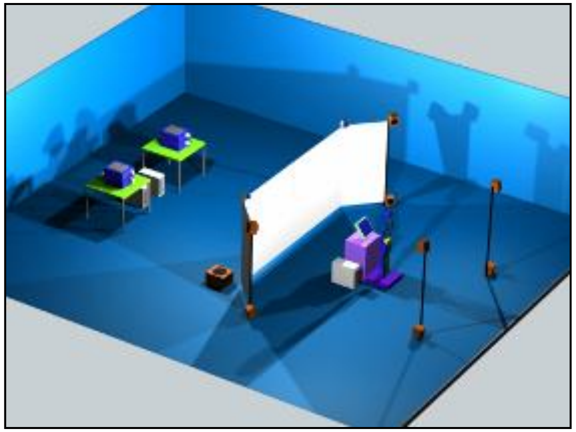
Writing VR applications is an hard task

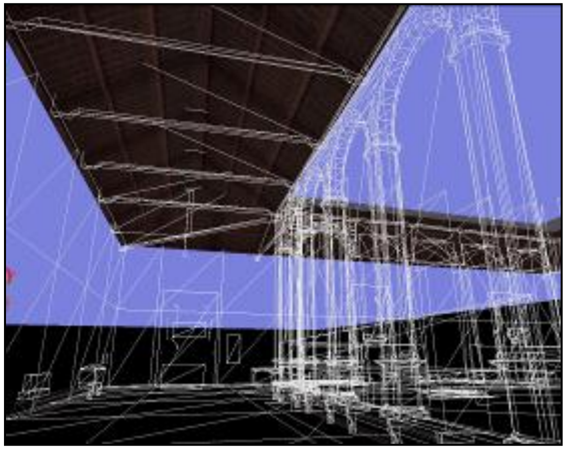
- Often require good C++ skill and a deep knowledge of several HW technologies (video/audio/haptic)

- Plenty of tools available, but mixing them is no trivial. Also, hi-performance tools and libraries need to be properly handled (otherwise performances might suffer)

- Multidisciplinary: team-work is a necessity

3D @ PERCRO



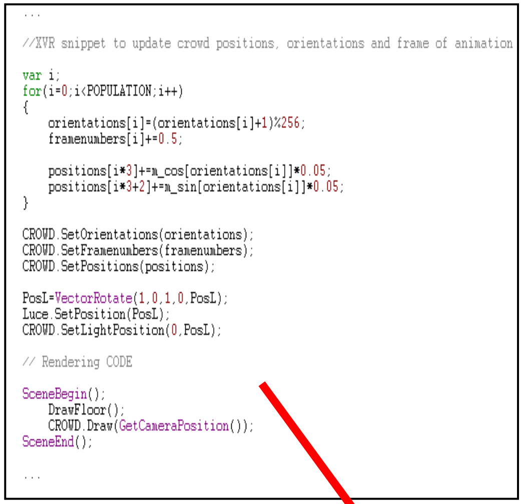


What is XVR?

- A fully integrated development environment
- S3D a C-like programming language (but VR-oriented)
- The IDE integrates a very fast compiler
- Using precompiled byte code
- The Virtual Machine executed in a Web plugin
- Applications can be embedded inside web pages
- Data exchange with the Web page JavaScript, Flash
- Extensibility through external C++ modules (custom or a-la CType)

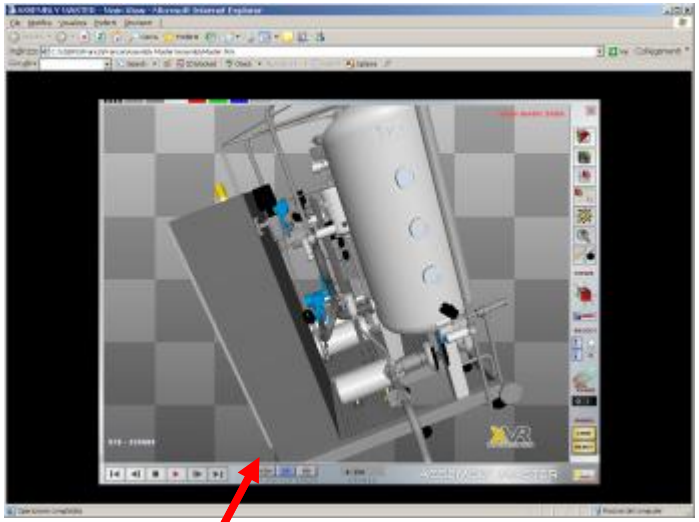
XVR Workflow

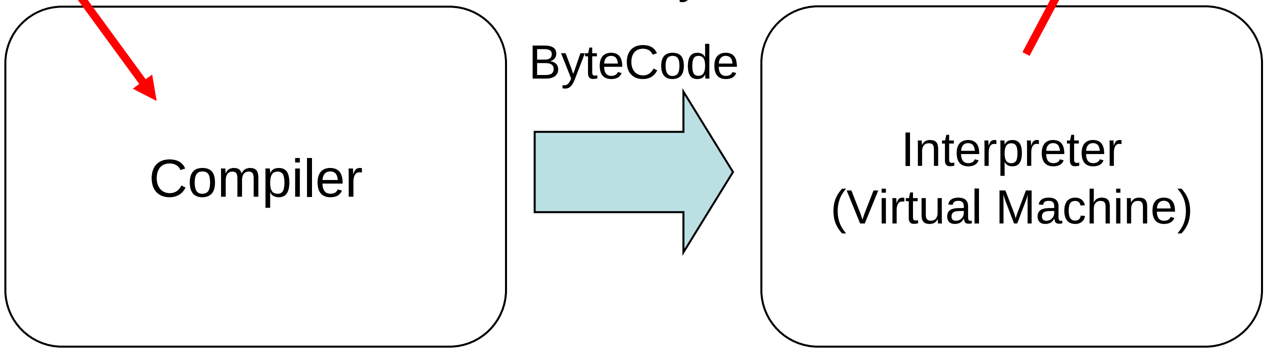
Dedicated scripting language



```
...

//XVR snippet to update crowd positions, orientations and frame of animation

var i;
for(i=0;i<POPULATION;i++)
{
    orientations[i]=(orientations[i]+1)%256;
    framenumbers[i]+=0.5;

    positions[i*3]+=m_cos[orientations[i]]*0.05;
    positions[i*3+2]+=m_sin[orientations[i]]*0.05;
}

CROWD.SetOrientations(orientations);
CROWD.SetFramenumbers(framenumbers);
CROWD.SetPositions(positions);

PosL=VectorRotate(1,0,1,0,PosL);
Luce.SetPosition(PosL);
CROWD.SetLightPosition(0,PosL);

// Rendering CODE

SceneBegin();
    DrawFloor();
    CROWD.Draw(GetCameraPosition());
SceneEnd();

...
```

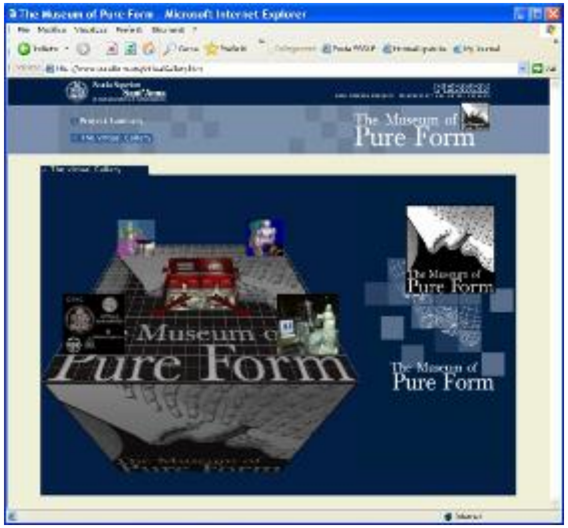
Output

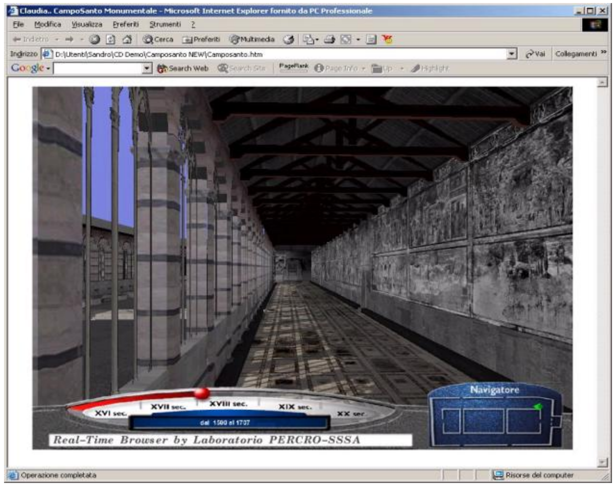


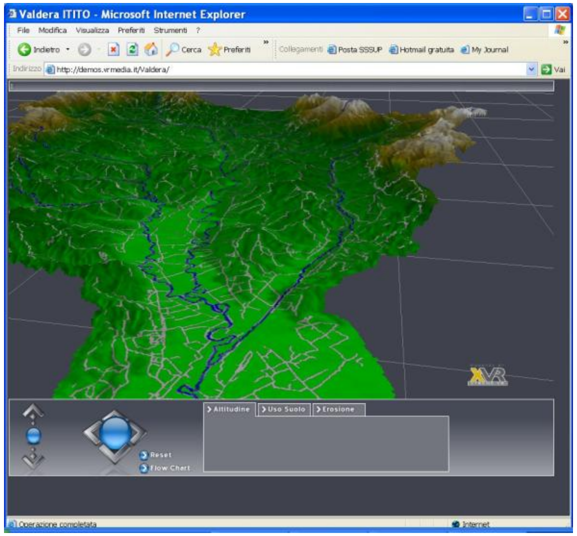


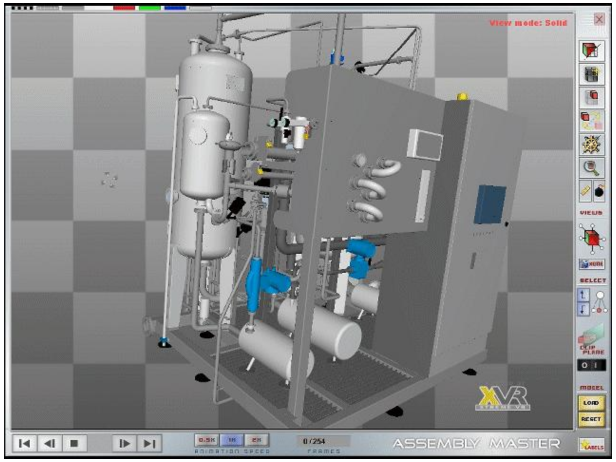
Binary ByteCode

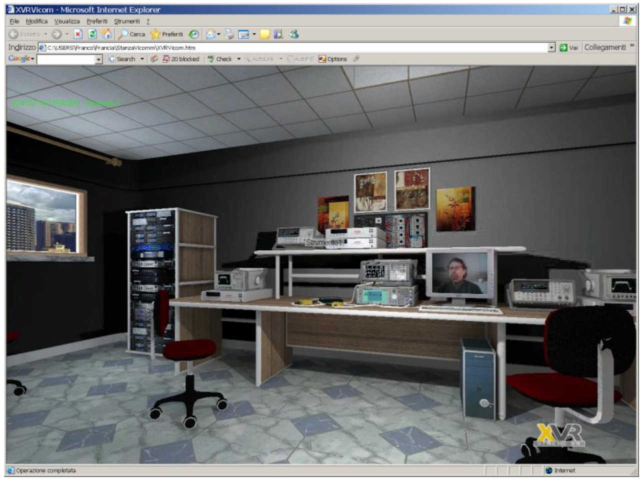
A WEB-enabled technology…

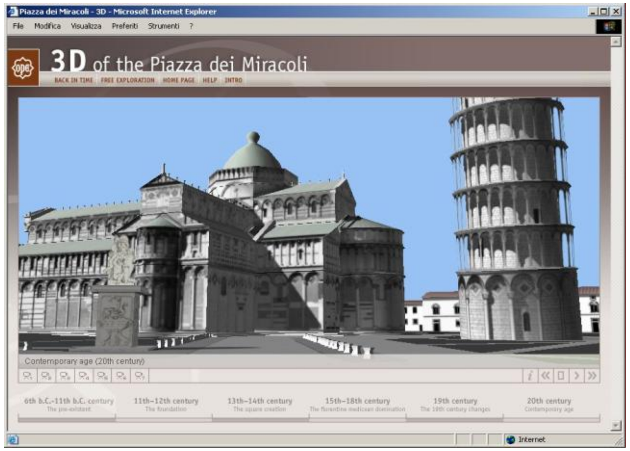



Claudia.. CampoSanto Monumentale - Microsoft Internet Explorer fornito da PC Professionale
XVI sec.
XVII sec.
XVIII sec.
XIX sec.
XX sec.
dal 1500 al 1737
Navigatore
Real-Time Browser by Laboratorio PERCRO-SSSA
Operazione completata
Risorse del computer

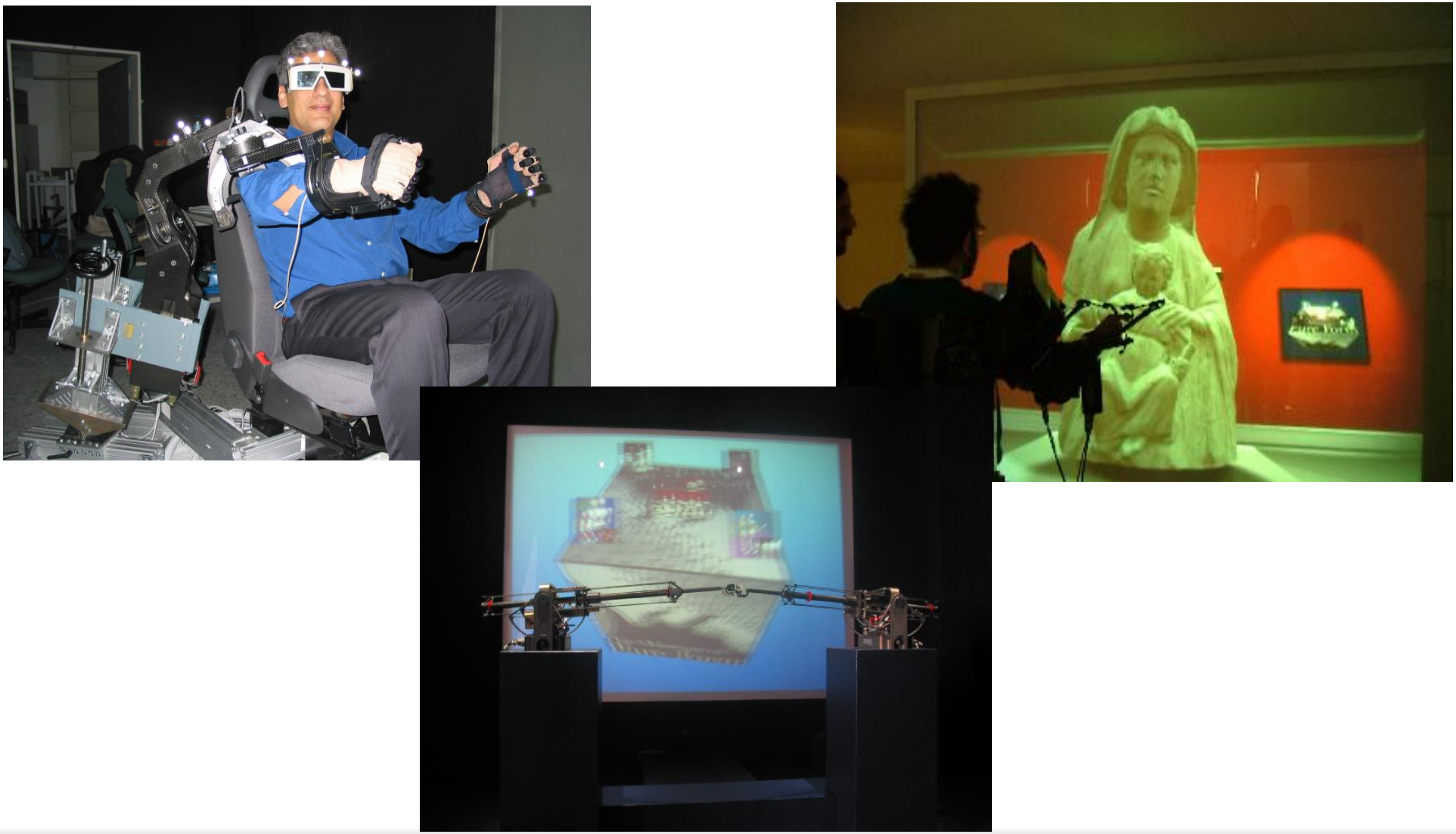

Valdera ITITO - Microsoft Internet Explorer
http://demos.vrmedia.it/Valdera/
Altitudine
Uso Suolo
Erosione
Reset
Flow Chart
Operazione completata
Internet






Piazza dei Miracoli - 3D - Microsoft Internet Explorer
3D of the Piazza dei Miracoli
BACK IN TIME
FREE EXPLORATION
HOME PAGE
HELP
INTRO
Contemporary age (20th century)
6th b.C.-11th b.C. century
11th-12th century
13th-14th century
15th-18th century
19th century
20th century
Internet

Advanced VR Installations



Limits of S3D

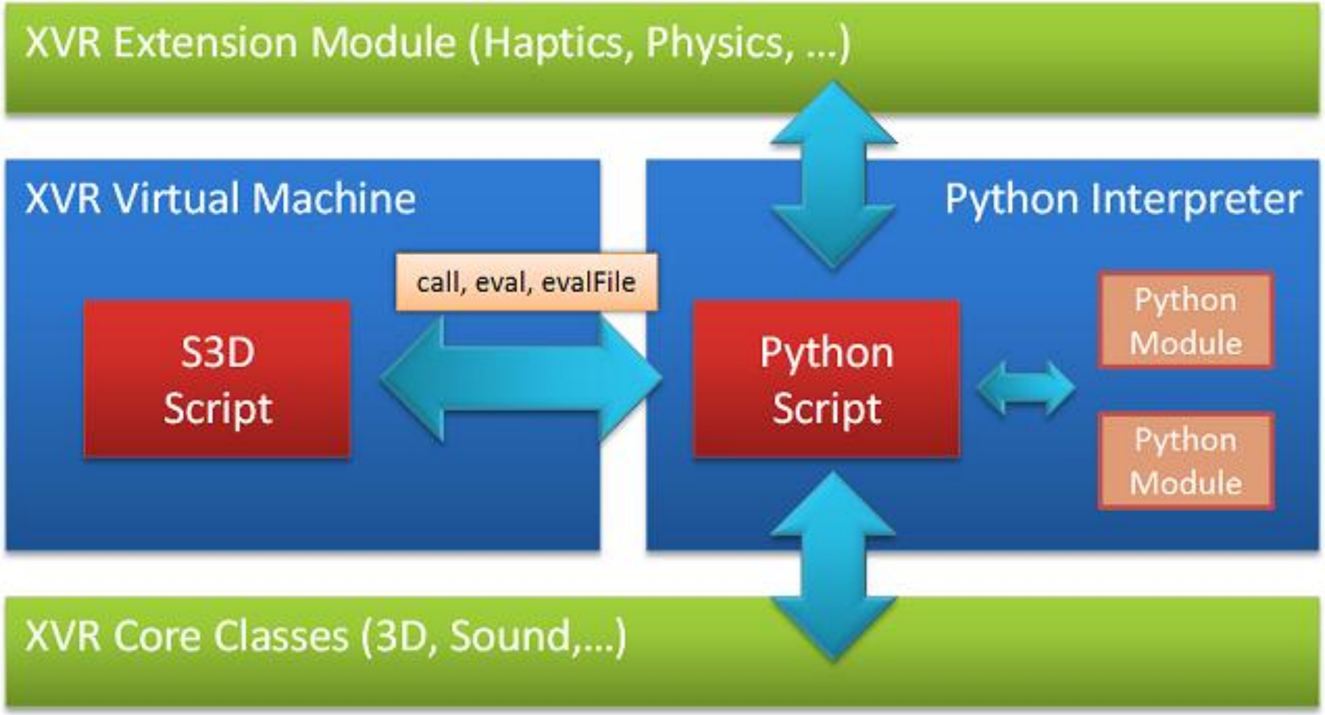
- Yet another scripting language although with a small learning curve
- No debugging tools
- Compile only language, no dynamic scripting
- No multi-threading

… now enters PYXVR

Introducing PYXVR

- **Scripting system for VR and 3D Web applications based on Python**
- **All the advantages of XVR and Python**
- **Python**
  - Wonderful language
  - Debugging (e.g. Winpdb)
  - Existing libraries
  - Dynamic Execution
- **XVR**
  - Web Deployment with versioning
  - Core 3D/VR libraries
- **PYXVR uses**
  1. Extending an existing XVR application with Python modules
  2. Developing a full 3D/VR application in Python

PYXVR Architecture



The Python script accesses all the functions (e.g. glColor) and objects of the XVR VM. Also the functions defined in the S3D script.

PYXVR application structure

The XVR engine load the application and invokes Callbacks

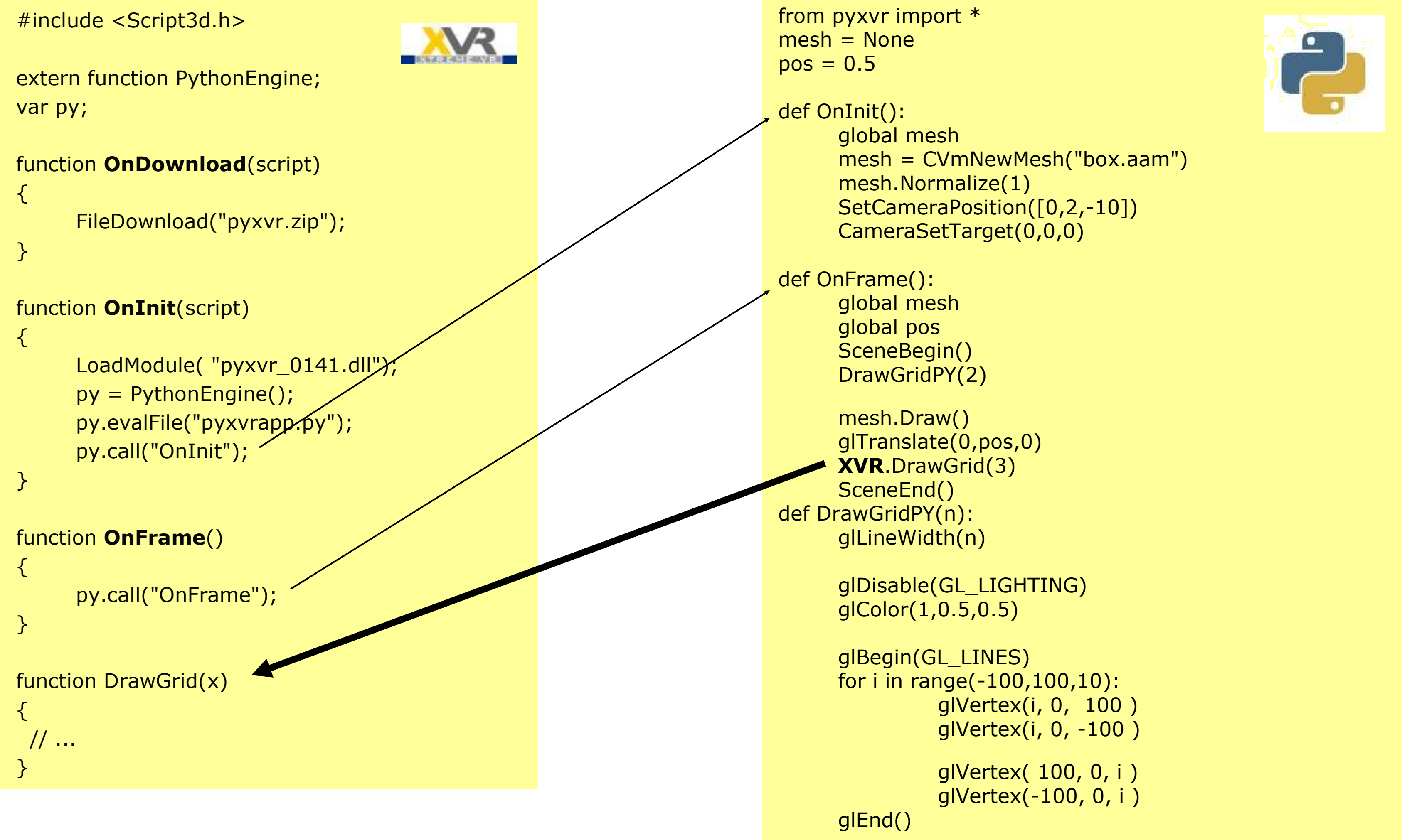
- OnDownload(param) for getting resources

  files are downloaded in a temporary directory and zip archives unpacked

- OnInit(param) for initialization
- OnFrame() at every rendering frame (~50Hz)
- OnTimer() about every 1ms
- OnEvent(event) for asynchronous events

The typical PYXVR application sends these events to Python

Minimal PYXVR Application



```
#include <Script3d.h>

extern function PythonEngine;
var py;

function OnDownload(script)
{
        FileDownload("pyxvr.zip");
}

function OnInit(script)
{
        LoadModule( "pyxvr_0141.dll");
        py = PythonEngine();
        py.evalFile("pyxvrapp.py");
        py.call("OnInit");
}

function OnFrame()
{
        py.call("OnFrame");
}

function DrawGrid(x)
{
  // ...
}
```

```
from pyxvr import *
mesh = None
pos = 0.5

def OnInit():
        global mesh
        mesh = CVmNewMesh("box.aam")
        mesh.Normalize(1)
        SetCameraPosition([0,2,-10])
        CameraSetTarget(0,0,0)

def OnFrame():
        global mesh
        global pos
        SceneBegin()
        DrawGridPY(2)

        mesh.Draw()
        glTranslate(0,pos,0)
        XVR.DrawGrid(3)
        SceneEnd()
def DrawGridPY(n):
        glLineWidth(n)

        glDisable(GL_LIGHTING)
        glColor(1,0.5,0.5)

        glBegin(GL_LINES)
        for i in range(-100,100,10):
                glVertex(i, 0,  100 )
                glVertex(i, 0, -100 )

                glVertex( 100, 0, i )
                glVertex(-100, 0, i )
        glEnd()
```

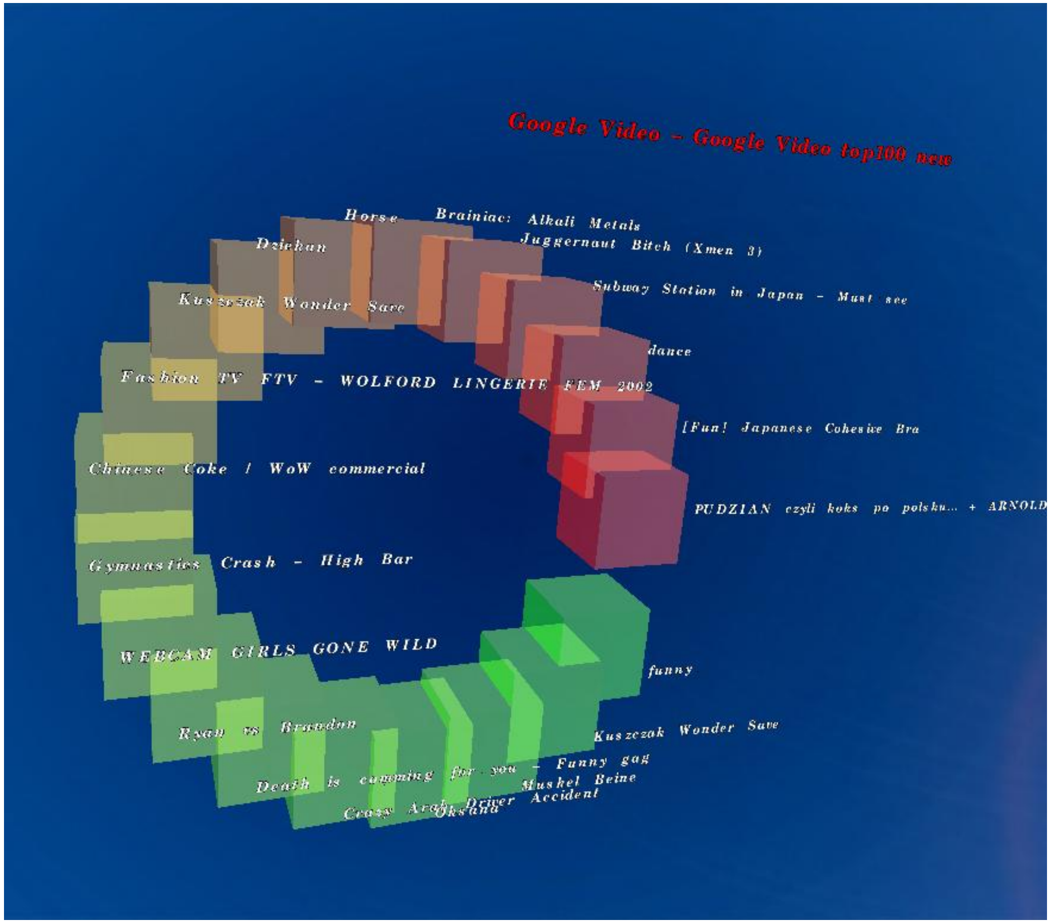
PYXVR Deployment

Core Components

Myscript.py	My Python Script
pyxvr.zip	PYXVR core and Python libraries
pyxvrmin.s3d.bin	Stub XVR application that loads Myscript.py

Web	Offline
Myscript.htm	pyxvr.exe

Example - RSS


Google Video - Google Video top100 new
Horse
Brainiac: Alkali Metals
Dziekan
Juggernaut Bitch (Xmen 3)
Kuszczak Wonder Save
Subway Station in Japan - Must see
dance
Fashion TV FTV - WOLFORD LINGERIE FEM 2002
[Fun] Japanese Cohesive Bra
Chinese Coke / WoW commercial
PUDZIAN czyli koks po polsku... + ARNOLD
Gymnastics Crash - High Bar
WEBCAM GIRLS GONE WILD
funny
Ryan vs Brandon
Kuszczak Wonder Save
Funny gag
Death is comming for you
Muskel Beine
Crazy Arab Driver
Oksana
Accident

Python provides modules

Feedparser based parsing of RSS

Type Mapping

Type mapping is fundamental, and primitive types are directly mapped (int,bool,String)

From XVR to Python

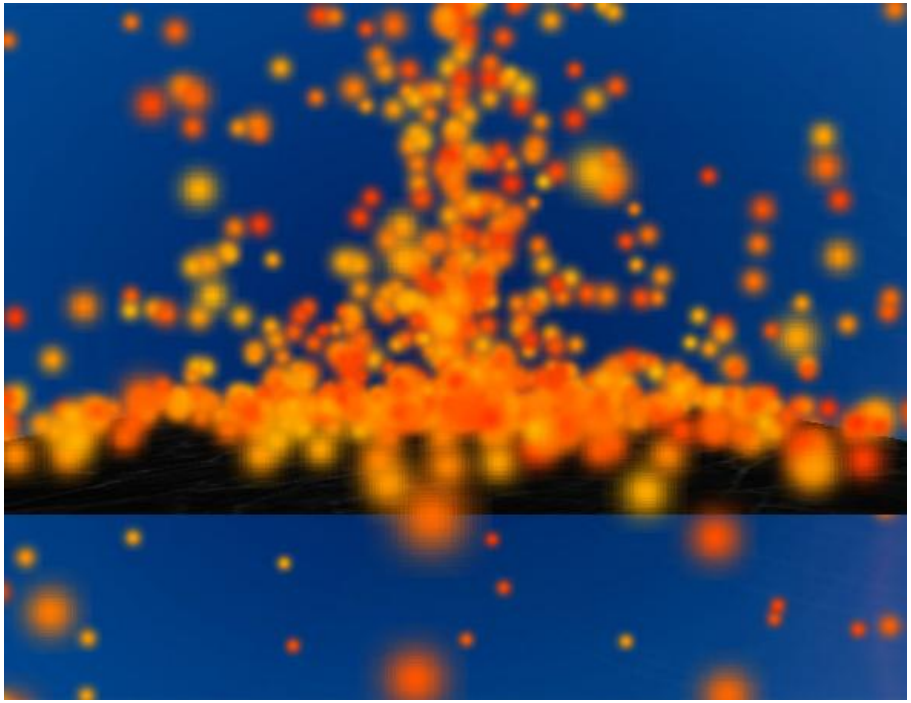
- vector of float [1,2,3] => array('f')
- array => List
- object => wrapper object of class XVRObject

From Python to XVR

- object => only of class XVRObject
- list, tuple => array (although could be vector)

Example – PYXVR particle

Python porting of the particle system



PYXVR - Particle Performance

A performance comparison only on the update of the particles, not *rendering*

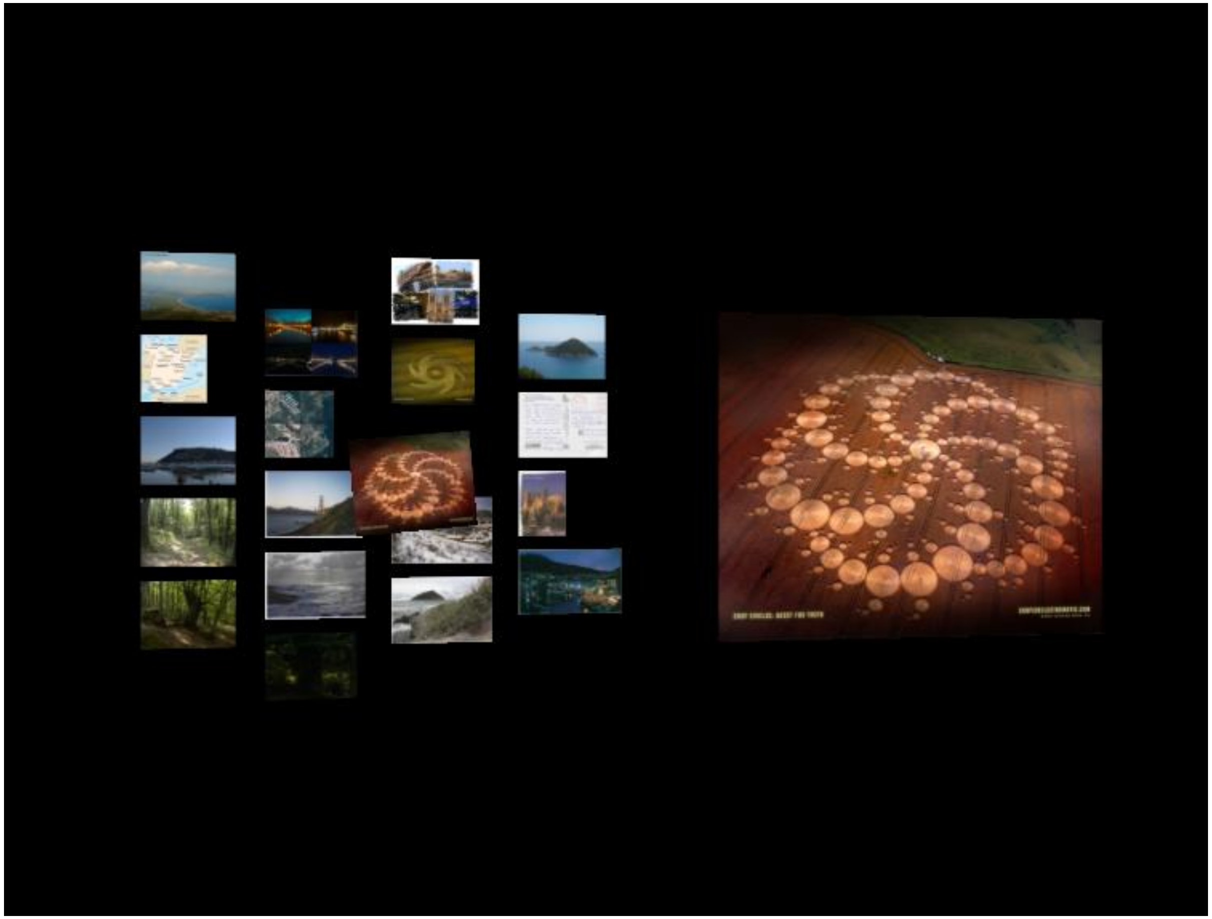
Number of Particles	S3D	PY	S3D Wine	PY Wine
10000	23fps	19fps	20fps	15fps
20000	12fps	10fps	11fps	8fps
40000	6fps	5fps	5fps	4fps

S3D is little faster because of optimized code for vector operations, and Python version could be optimized

Linux version is running Wine so it is using Python for Windows

Example – file access

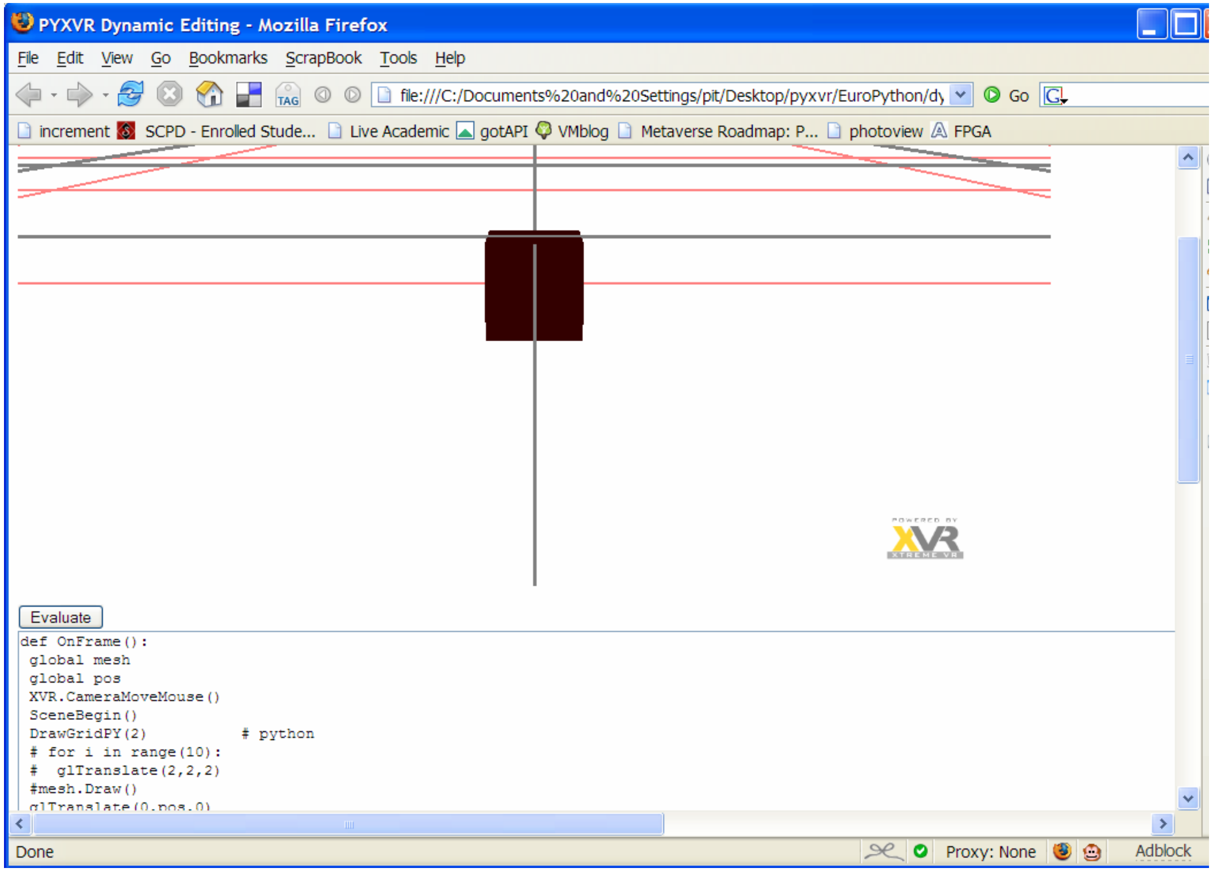
Simple 3D photo browser that uses Python file listing functions



Advanced use - On the fly coding

Modify the Python code and the 3D scene at run time

Should behave like a Python shell
Need improvements in error handling.


PYXVR Dynamic Editing - Mozilla Firefox
File Edit View Go Bookmarks ScrapBook Tools Help
file:///C:/Documents%20and%20Settings/pit/Desktop/pyxvr/EuroPython/dy
Go
increment SCPD - Enrolled Stude... Live Academic gotAPI VMblog Metaverse Roadmap: P... photoview FPGA
POWERED BY XVR XTREME VR
Evaluate
Done
Proxy: None
Adblock

```
def OnFrame():
 global mesh
 global pos
 XVR.CameraMoveMouse()
 SceneBegin()
 DrawGridPY(2)           # python
 # for i in range(10):
 #  glTranslate(2,2,2)
 #mesh.Draw()
 glTranslate(0,pos,0)
```

Advanced Uses - Stackless

- Virtual Reality applications with agents are pretty interesting
- Stackless provides a interesting way to change the programming model
- Just replace python24.dll with the one from Stackless

```
def Agent(id):
        life = random.randint(1,1000)
        pos = [random.random()*5-2.5,random.random()*3,0]
        print ("Agent ",id," ",life," ",pos)
        vel = [random.random()*0.05-0.025,random.random()*0.04,0]
        for i in range(life):
                pos[0] = pos[0] + vel[0]
                pos[1] = pos[1] + vel[1]
                pos[2] = pos[2] + vel[2]
                XVR.SetAgentPosition(id, pos[0],pos[1],pos[2])
                schedule()
        print "death"
```

Open Issues

- Security of execution from Web pages
- Improving method invocation performance (by name)
- Access of Python objects from XVR
- Windows only (except Linux using WINE)

Conclusions

- PYXVR is a tool for writing VR solutions, 3D Web applications or Games
- Based on the great Python language and an advanced VR toolkit

Enjoy it, just Google “PYXVR”