Introduction to X3D

Roberto Ranon
ranon@dimi.uniud.it
HCI Lab
University of Udine, Italy
http://hcilab.uniud.it
Outline

- What is X3D?
- X3D basics
  - including some new capabilities and details on the (near) future of X3D
- How the ISO specification is organized
- X3D and VRML97
- Available SW tools
What is X3D?

• eXtensible 3D Graphics

• ISO open standard for (Web-enabled) interactive 3D content integrated with multimedia

• also an interchange format for integrated 3D graphics and multimedia

• developed by the Web3D Consortium as a successor of the VRML language

• already officially incorporated within the MPEG-4 multimedia standard
X3D supports ...

- **3D graphics** - Polygonal geometry, *parametric geometry*, hierarchical transformations, lighting, materials and *multi-pass/multi-stage* texture mapping

- **2D graphics** - Text, *2D vector and planar shapes displayed within the 3D transformation hierarchy*

- **Animation** - Timers and interpolators to drive continuous animations; *humanoid animation and morphing*

- **Spatialized audio and video** - Audiovisual sources mapped onto geometry in the scene

- **User interaction** - Mouse-based picking and dragging; *keyboard input*
X3D supports ... (2)

- **Navigation** - Cameras; user movement within the 3D scene; collision, proximity and visibility detection

- **Networking** - Ability to compose a single X3D scene out of assets located on a network; hyperlinking of objects to other scenes or assets located on the World Wide Web

- **User-defined objects** - Ability to extend built-in browser functionality by creating user-defined data types (PROTOs)

- **Scripting** - Ability to dynamically change the scene via programming and scripting languages

- **Physical simulation** - Humanoid animation; geospatial datasets; integration with Distributed Interactive Simulation (DIS) protocols
X3D Basics
A X3D file defines a **scene graph**, i.e. a direct acyclic graph describing the 3D world being created (e.g. 3D objects, interactivity)

Each node in the graph is an instance of one of the available node types

- e.g. a *Shape* node defines a shape, a *Transform* node positions, orients and scale its children nodes, a *DirectionalLight* node defines a light source in the world

Nearly identical to the VRML97 scene graph, but new nodes added to incorporate advances in 3D HW & SW
New nodes example

- With multitexturing, multiple textures may be applied to a single geometry node and blended according to a predefined set of operations.
  - e.g., for light mapping effects

- Multitexturing is accomplished using the MultiTexture, MultiTextureCoordinate, and MultiTextureTransform nodes
• X3D content can be expressed in
  - Classic VRML encoding, nearly backward compatible with VRML97
  - XML encoding, supported by Schema or DTD
• All encodings are interchangeable
<X3D version='3.0'
  profile='Interchange'>
  <head>
  </head>
  <Scene>
    <Group>
      <NavigationInfo
type='"EXAMINE"'/>
      <DirectionalLight/>
      <Transform
        translation='3.0 0.0 1.0'>
        <Shape>
          <Sphere radius='2.3'/>
          <Appearance>
            <Material
diffuseColor='1.0 0.0 0.0'
          </Appearance>
        </Shape>
      </Transform>
    </Group>
  </Scene>
</X3D>

#X3D V3.0 utf8
PROFILE Interchange
Group {
  children [
    NavigationInfo {
      type [ "EXAMINE" ]
    }
    DirectionalLight {}
    Transform {
      translation 3.0 0.0 1.0
      children [
        Shape {
          geometry Sphere { radius 2.3 }
          appearance Appearance {
            material Material {
              diffuseColor 1.0 0.0 0.0
            }
          }
        }
      ]
    }
  ]
}

XML encoding

Classic VRML encoding
• XML provides parsing, validation, XSLT conversion, ..., for free

• Integration with the next-generation web technologies, also exploiting the efforts of the W3C

• **Typical scenario** (also in e-learning apps): X3D, as a format that defines visual information, is the last stage in a production pipeline.
  - using XML stylesheets, one can keep data in some native XML format (which represents concepts, relationships, ...), and obtain a 3D representation as a single transformation step.
Example: CML to X3D

- CML (Chemical Markup Language) is a XML language for chemistry
- With a properly-written XML stylesheet, one can obtain a X3D representation of ANY CML file (or another visual representation, e.g. HTML)

cml2X3d stylesheet done by Nicholas Polys, VirtuWorlds LLC
• authors can define new nodes by using prototypes
  - e.g., to define application-specific high-level nodes

• A PROTO statement defines a new node type in terms of already defined (built-in or prototyped) node types
  - PROTO Cube [ ] { Box { } }

• Once defined, prototyped node types may be instantiated in the scene graph exactly like the built-in node types
  - Shape { geometry Cube { } }
- Complex dynamical behaviors in the 3D world can be impossible to implement without the power of a programming languages

- these kinds of decisions are expressed programmatically using the **Scene Access Interface (SAI)** internally (from Script nodes) or externally (from other application programs)

- X3D has a single unified API (VRML97 had an internal scripting API plus an external API, called EAI).

- ECMAScript and Java bindings are defined

- ECMAScript is required for conformant applications.
The (near) future of X3D

• Programmable X3D Shaders
  - more visual realism & efficiency
  - can experiment with them using BS Contact Preview

• CAD Distillation Format
  - extension of X3D designed to distill down high complexity CAD data to low complexity

• Binary Encoding for faster parsing (2-5 times) and downloading times (10-25 times)
  - with both loseless and lossy compression
How the X3D specification is organized
How the spec is organized

- X3D is composed by 3 separate ISO specs:
  - **X3D framework & SAI**: describes structural and runtime models, external programming functionality in abstract terms.
  - **X3D encodings**: specifies XML and Classic VRML encodings of X3D files
  - **X3D language bindings**: specifies the binding of the services in the X3D architecture to the ECMAScript and Java programming languages
Components and Profiles

- X3D objects and services are grouped into **components** (i.e., set of related functionalities)
- Components have **multiple levels** of increasing capability
- **Profiles** are a collection of components at specific levels designed to support particular application domains
- Authors specify required profile and (optionally) additional components at the beginning the X3D file
Example

• <X3D profile='Immersive'>
  <head>
    <component name='Geospatial'/>
    <component name='NURBS' level='2'/>
  </head>
  <Scene>
    <!-- Scene graph nodes are added here -->
  </Scene>
• </X3D>
• **Geometry**
  - Shape, Geometry3D, Geometry2D, Text

• **Environment**
  - Sound, Lighting, Navigation, Environmental Effects

• **Animation**
  - Interpolation, Event Utilities

• **Appearance**
  - Rendering, Texturing

• **Sensors**
  - Pointing Device Sensor, Key Device Sensor, Environmental Sensor, Time

• **Structure**
  - Networking, Grouping

• **Other**
  - Geospatial, H-Anim, NURBS, DIS, Scripting
• Interchange
• Interactive
• Immersive
• Full

• MPEG4 Interactive
• CDF (Cad Distillation Format)
Main functional differences:

- file header (e.g. you need to define a profile)
- access names for fields have changed (e.g. eventIn → inputOnly, exposedField → inputOutput)
- scripts can have inputOutput (i.e., exposed) fields
- runtime and programming model for scripting is consistent between programming languages and whether you are inside or outside the browser
- ...
Converting VRML97 files to X3D

- files that do not contain scripts or externprotos need only minor changes
  - e.g., change the header and insert the new PROFILE statement
- files that contain scripts may require more extensive modifications
- a X3dToVrml97 translator program is available, but not perfect at all
  - e.g. does not perform changes into script code
Available SW tools
X3D tools: browsers

• Commercial:
  - BitManagement BS Contact www.bitmanagement.de
  - Flux www.mediamachines.com
  - Octaga www.octaga.com
  - VCOM Venues X3D www.vcom3d.com

• Open-source:
  - FreeWRL www.crc.ca/FreeWRL
  - XJ3D www.xj3d.org
X3D tools: authoring

• Commercial:
  - VizX3D www.vizx3d.com

• Open-source:
  - X3D-Edit www.web3d.org
  - VRML to X3D translator ovrt.nist.gov/v2_x3d.html
  - Exporters for Maya, 3DStudio (under development), Blender, MilkShape www.web3d.org