COLLADA™

Tutorial, part II

Web3D symposium
04/2007

Dr. Rémi Arnaud
Sony Computer Entertainment

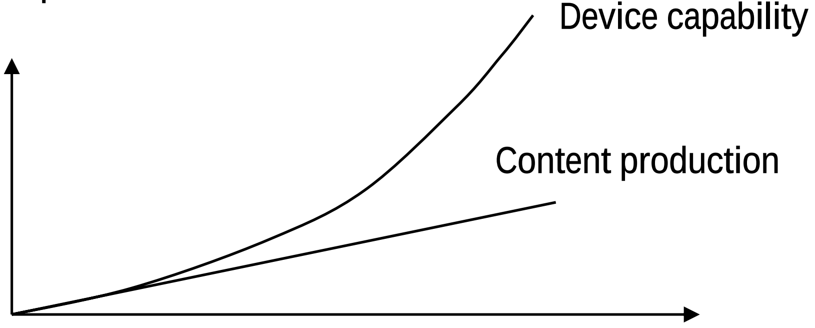
Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session

PS3 ducky - Image courtesy of Sony Computer Entertainmer

Design Goals (1)

- Content creation has become increasingly demanding
  - 3D devices expands in complexity and capabilities
  - Size of content ever growing
  - Shrinking production schedules
  - Ever growing budgets



- DCC tools not keeping pace with technology
  - Dynamic content
  - Shader FX
  - Physics
  - Collaborative development
- Need Export and Import !
  - Need flexibility creating content pipeline
  - Enable mixing tools for better productivity
  - Opening the door to a variety of third party tools
  - Need a language – import and export required for true communication

COLLADA

Design Goals (2)

- Improve Asset Management System
  - Assets are not files, but XML elements
  - Interactive development
  - Collaborative design
- Content pipeline processing
  - Contain enough information to enable creation of target optimized format
  - Can store both DCC centric formatted data and game engine data
    - Polygons with holes
    - Triangle strips
- Extensibility
  - Final user can easily extend the features
    - Extension by substitution
    - Extension by addition
  - Tool vendors can add elements
    - Tool specific data for no-loss import / export
    - Candidate for standardization in future releases
- Modularity
  - Libraries of assets
  - Many things can be externally referenced
- Need validation capability
  - Syntax to be validated, regardless of the content
    - XML Schema validation
  - Need to enable verification of exporter and importer quality
    - COLLADA needs a full fledge conformance test

Objectives

- Raise the quality and feature set of content
- Liberate content from proprietary formats and API
- Bring more tools to developers/artists
- Drastically improve productivity
- Lower the barrier to entry for tool & middleware companies
- Bring ‘advanced’ features into the main stream
- Enable user content flow

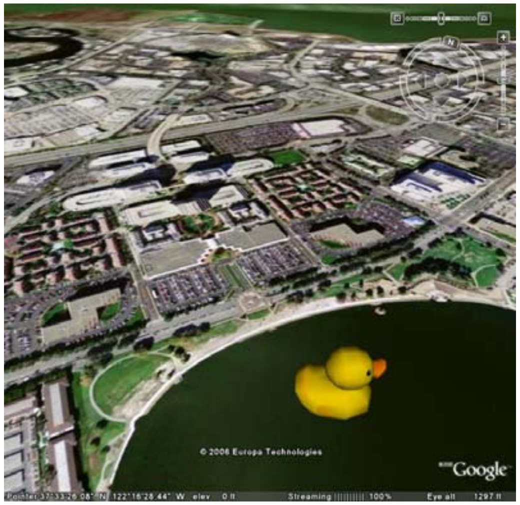
COLLADA adopted by Khronos

- COLLADA 1.4.0 adopted as industry standard by Khronos 01/06
  - Not an API, COLLADA is a language
  - Intermediate Digital Asset Exchange format (.dae)
  - XML Schema technology
- is the first standard specification

www.khronos.org/collada

Status

- 1.4.1 released August 2006
  - Patch update from 1.4.0
- Thousands of active users
  - Many professional game developers currently using COLLADA
    - THQ, EA, Konami, NCsoft, DoubleFine, Rockstar …
    - Unreal Engine, C4 engine, Ogre, Irrlicht Engine, …
  - Approaching 15,000 1.4.1 spec download
- Adopted by Google Earth for 3D models
  - COLLADA content manipulated by mainstream users
  - SketchUp, 3D warehouse
  - Photomodeler, realviz, 3D nature ….
- More DCC tools have COLLADA support
  - Since COLLADA 1.0: 3dsMax, Maya, XSI
  - Since COLLADA 1.4.x: Blender, Houdini
  - **Photoshop** CS3 extended introduces 3D, with COLLADA support
  - Work in progress: DAZ, Modo, Lightwave
- Official conformance test in development
  - Early access available to Khronos members
  - Publicly available reference viewer and coherency test



PS3 ducky on earth
Image courtesy of Google Inc (earth.google.com)

Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session

Digital Berlin – 44000 COLLADA buildings !

Why contribute to COLLADA ?

- There is only one good reason for standards to exist:
  - Companies supporting the standard make/save money if the standard exists
  - Contributing is expensive (resource, IP, membership)
    - Benefit need to be much higher than cost
- Follow the money
  - Sony Computer Entertainment make money selling content (games)
    - Better content = more customers = larger business
    - Better content tools = cheaper content = more incentive to create better content …
  - (Graphics) Hardware companies make money selling chips.
    - Need to keep the market hungry for more features/performance
    - Need more advanced content…
    - Also need standard API.
  - Digital Content Creation tools
    - Need to create content that can be used
    - Customers do not want their data prisoner of the tool
    - WARNING - Monopoles do not agree with this, they want to own your data, so you have to pay to get to it. (undocumented binary formats, licensed API – so source code).
  - Google makes money selling advertisement and technology
    - The more users = the more money
    - Content need to be attractive and entertaining
    - Focusing on user content

Why adding COLLADA support ?

- Yet another format ?
  - Reduce the number of formats you have to support – get rid of vintage formats
    - example: RealViz went from 6 to 1 format to support
  - Grow your customer base
    - COLLADA let you interface with more tools, in different business sectors
    - Example: SketchUp entry in the game industry for rapid prototyping
  - Follow COLLADA feature set and roadmap
    - Start with geometry and textures
    - Then add animation, physics, skinning ….
    - This provides your customer a clear roadmap, and a definite way to get there

    “We decided to use COLLADA because there are many features I do not yet support”
  - Stop wasting money writing exporters and fighting with DCC SDK bug
    - Max+Maya+Softimage+…. exporters is more expensive than 1 COLLADA importer
  - Reduce your support cost
    - COLLADA validation and coherency test
    - Exporters are supported, report the bug and let them do the work

Example COLLADA businesses

- Feeling Software - contractor
  - COLLADA plug-ins development for tool vendors
  - Gold support contracts for developers
  - Special projects – highly competitive due to re-use of COLLADA technology
- Pixel Academy - game developer
  - COLLADA 101 Web course
  - Special training sessions
  - Advanced follow-up trainings
    - Game development with COLLADA
    - …
- Other business opportunities
  - COLLADA asset management and content pipeline tools
  - COLLADA asset creation (sub-contracting to game developers)
  - COLLADA binary distribution format (codec)
  - COLLADA community / user content (flikr / youtube for 3D)

Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session

Image courtesy of DAZ Production (www.daz3d.com)

Developing Web Applications with COLLADA and X3D

A Whitepaper

Dr. Rémi Arnaud
remi@collada.org
Sony Computer Entertainment

Tony Parisi
tparisi@mediamachines.com
Media Machines, Inc.

March 25, 2007

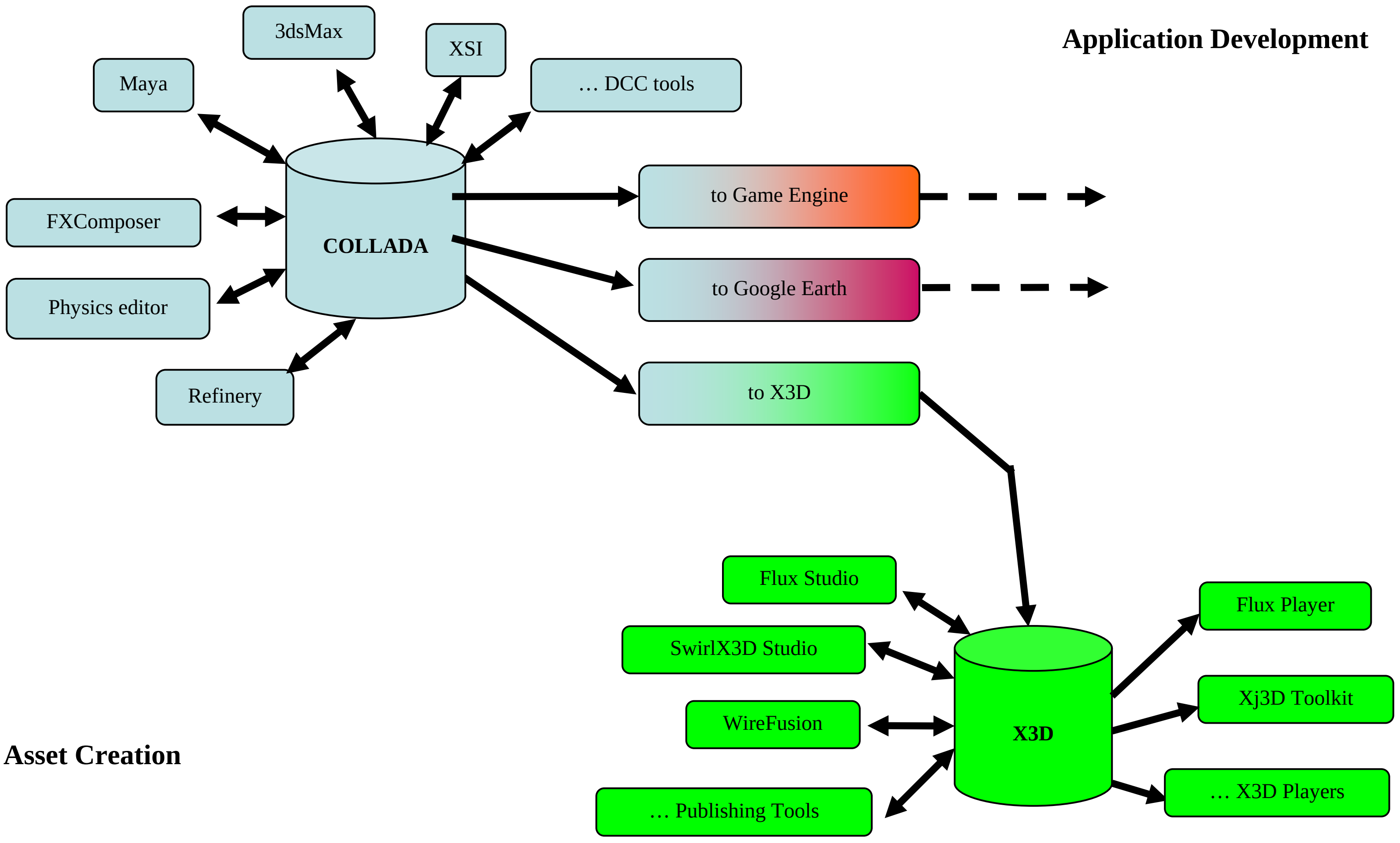
Comparing COLLADA and X3D

- *COLLADA is an intermediate format*
  - COLLADA enables the transformation of assets as they journey from content tools that use higher level description paradigms to applications that require platform-specific optimized descriptions.
  - COLLADA enables the free exchange of asset data, enabling developers to construct an authoring pipeline from multiple tools
- *X3D is a delivery format*
  - X3D focuses on the visualization of 3D assets within applications. X3D has principally been targeted for the Web (thus the name and focus of the Web3D Consortium
  - X3D specifies behaviors and interaction, and it includes both a specific *run-time model* that enables picking, viewing, navigation, and scripting, and an API to manipulate the scene graph at run-time.

X3D picks up where COLLADA leaves off

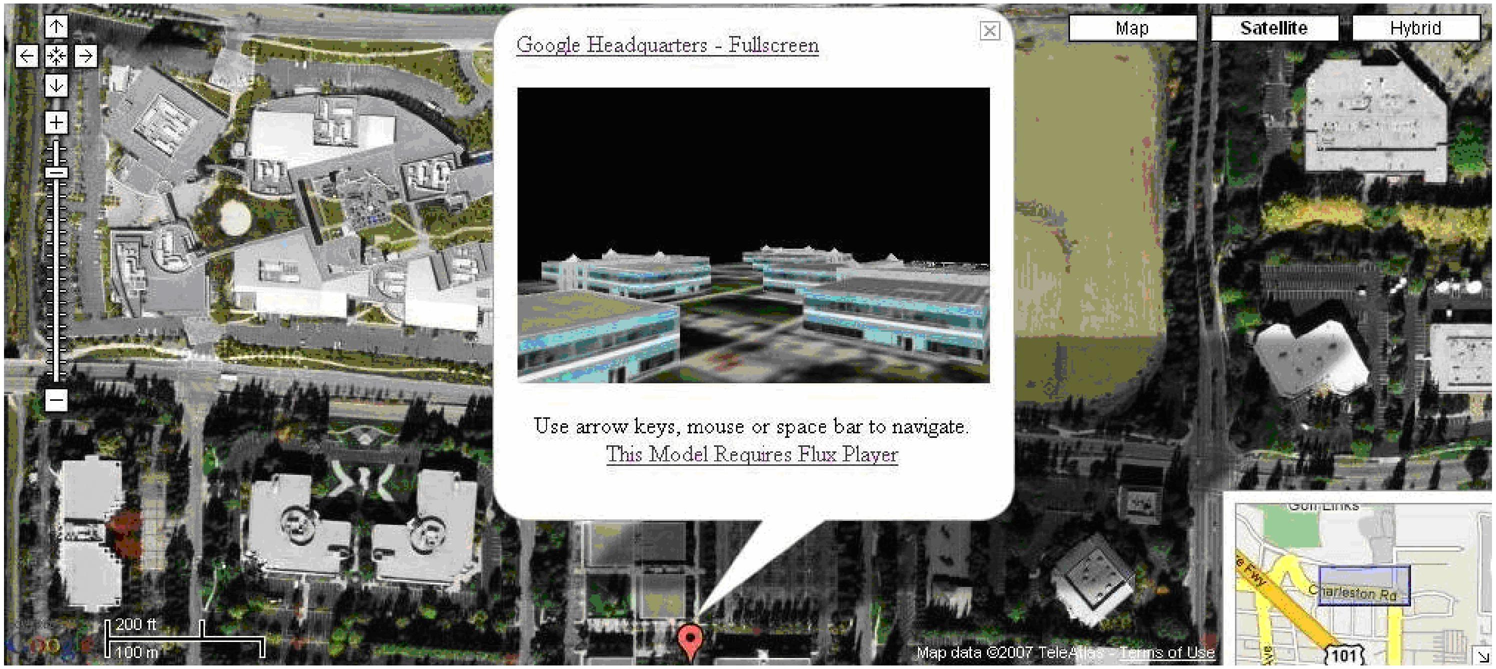
COLLADA defines the format for 3D assets but not their runtime semantics,
leaving that job to the application writer.

Using COLLADA for X3D Applications



Quality COLLADA-to-X3D converters

Sketchup -> COLLADA -> 3D Warehouse -> Flux Studio -> X3D -> Web page



Google Headquarters - Fullscreen

Use arrow keys, mouse or space bar to navigate.
This Model Requires Flux Player

“mashup” between Google Maps and 3D models

Image Courtesy Media Machines, Inc.

COLLADA or X3D – Which Should I Use?

- Example 1
  - Your application is a **content-creation tool** and you want your users to be able to develop assets for many markets. In this case, **COLLADA** is the appropriate choice. If in addition you want to target X3D run-time systems with value-added features such as interactivity, you may also want to implement support for X3D standards directly rather than going through an additional conversion. Typical applications include DCC tool such as 3dsMax, Maya, and Softimage.
- Example 2
  - Your application is a **content-editing tool targeting 3D Web publishing.** In this case you would implement only the **X3D** format. Typical applications include Flux Studio, WireFusion, and SwirlX3D.
- Example 3
  - Your application is a **run-time application** that can take advantage of the **X3D run-time** model. You use **X3D** directly, and use an X3D player as your application engine. Typical players include Flux Player, FreeWrl, and Xj3D. These applications can also directly load COLLADA content, if it makes sense to display content on the Web without any run-time interactivity, or use the COLLADA asset as an embedded object within a larger application (an X3D object type known as an “Inline”).

COLLADA or X3D – Which Should I Use?

- Example 4
  - Your application is a **run-time application** that requires its **own specific run-time.** In this case, use **COLLADA** in your toolchain, and create a COLLADA -> “Your Format” converter at the end of the toolchain. Typical applications include video games.
- Example 5
  - Your application is a **run-time application** that does not need to support X3D run-time specifics, but needs to be able to **accept content from a variety of sources.** In this case, directly load **COLLADA** in your application; you do not provide an off-line converter. Typical applications include Google Earth.

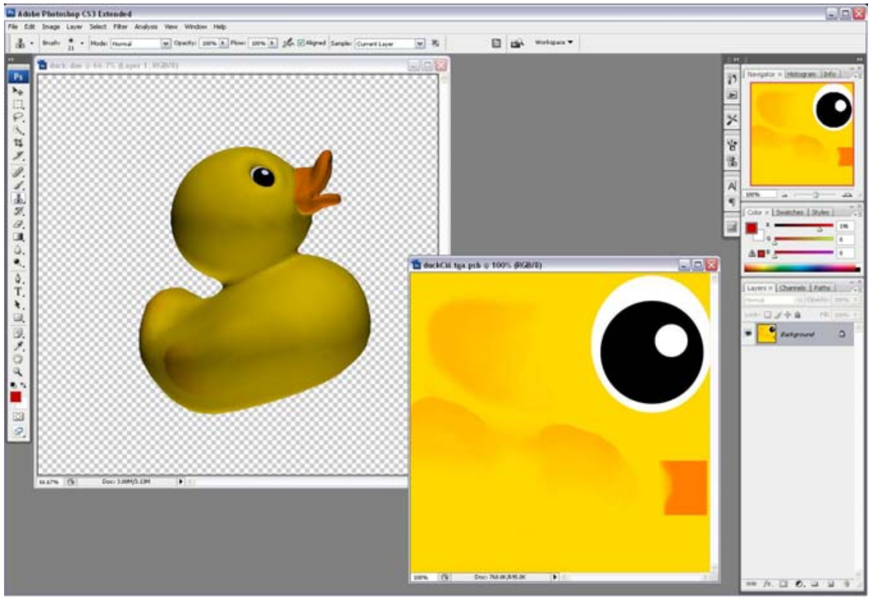
COLLADA & X3D – Ongoing collaboration

- COLLADA and X3D not only interoperate, but provide significant long-term value to content and application developers world wide.
- X3D development community can now take advantage of quality DCC tool COLLADA support, and save considerable effort by focusing on good quality COLLADA-to-X3D converters
- COLLADA and X3D teams have begun to identify areas where entire parts of one standard might be adopted for use directly within the other, starting with Physics
- Ideally, in the future, since both COLLADA and X3D are based upon XML, X3D might be able to directly accept COLLADA content without any conversion.

- tparisi@mediamachines.com
- The authors encourage questions and feedback on this whitepaper.

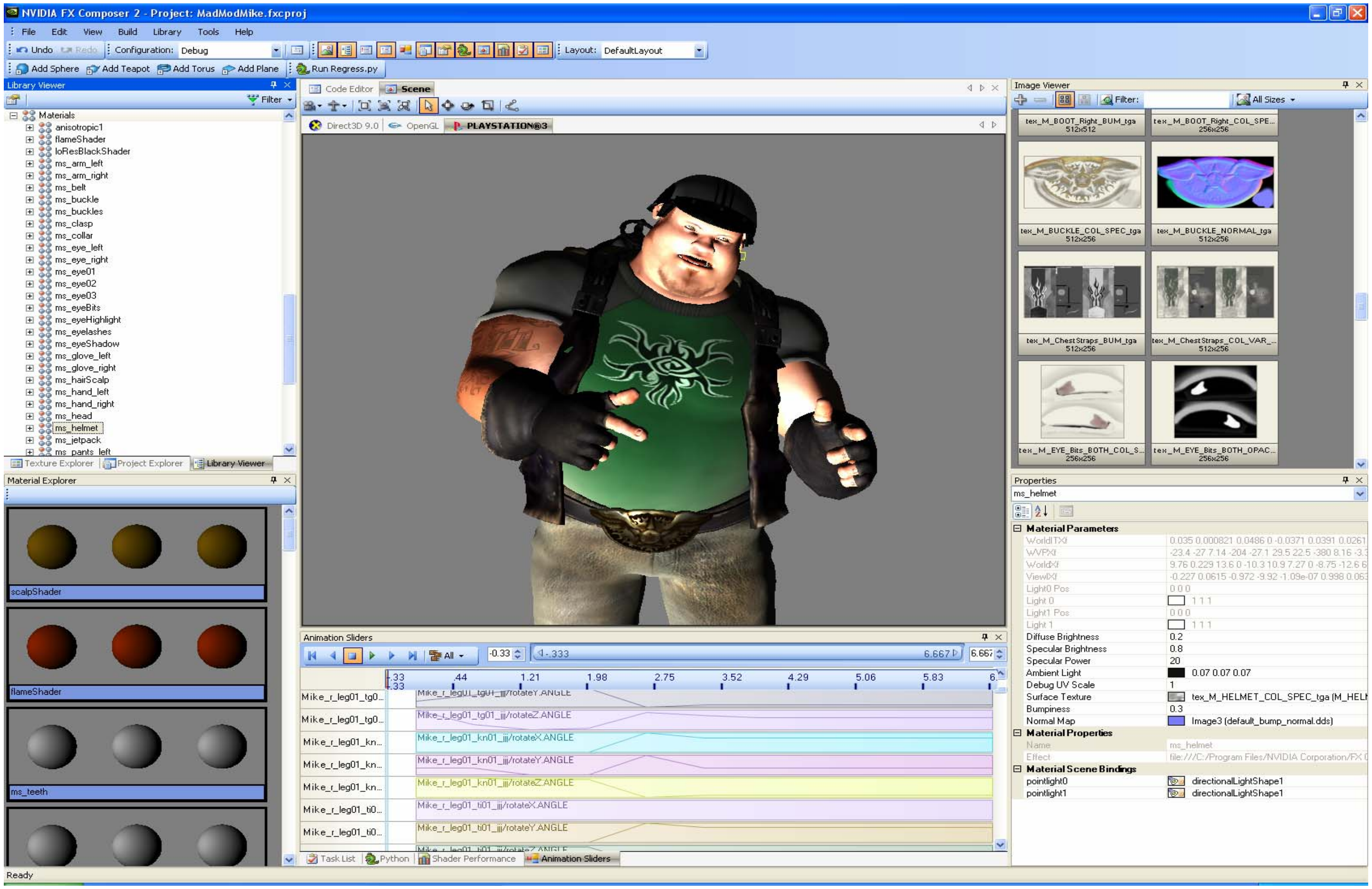
Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session


Adobe Photoshop CS3 Extended
File Edit Image Layer Select Filter Analysis View Window Help

Photoshop - Image courtesy Adobe Systems Incorporated

NVIDIA FXComposer 2.0 - a breakthrough


NVIDIA FX Composer 2 - Project: MadModMike.fxcproj
File Edit View Build Library Tools Help
Undo Redo Configuration: Debug Layout: DefaultLayout
Add Sphere Add Teapot Add Torus Add Plane Run Regress.py
Library Viewer
Filter
Materials
anisotropic1
flameShader
loResBlackShader
ms_arm_left
ms_arm_right
ms_belt
ms_buckle
ms_buckles
ms_clasp
ms_collar
ms_eye_left
ms_eye_right
ms_eye01
ms_eye02
ms_eye03
ms_eyeBits
ms_eyeHighlight
ms_eyelashes
ms_eyeShadow
ms_glove_left
ms_glove_right
ms_hairScalp
ms_hand_left
ms_hand_right
ms_head
ms_helmet
ms_jetpack
ms_pants_left
Texture Explorer Project Explorer Library Viewer
Material Explorer
scalpShader
flameShader
ms_teeth
Code Editor Scene
Direct3D 9.0 OpenGL PLAYSTATION®3
Animation Sliders
All -0.33 -1.333 6.667 6.667
-.33 .44 1.21 1.98 2.75 3.52 4.29 5.06 5.83 6.
Mike_r_leg01_tg0... Mike_r_leg01_tg01_jj/rotateY.ANGLE
Mike_r_leg01_tg0... Mike_r_leg01_tg01_jj/rotateZ.ANGLE
Mike_r_leg01_kn... Mike_r_leg01_kn01_jj/rotateX.ANGLE
Mike_r_leg01_kn... Mike_r_leg01_kn01_jj/rotateY.ANGLE
Mike_r_leg01_kn... Mike_r_leg01_kn01_jj/rotateZ.ANGLE
Mike_r_leg01_ti0... Mike_r_leg01_ti01_jj/rotateX.ANGLE
Mike_r_leg01_ti0... Mike_r_leg01_ti01_jj/rotateY.ANGLE
Task List Python Shader Performance Animation Sliders
Image Viewer
Filter: All Sizes
tex_M_BOOT_Right_BUM_tga 512x512
tex_M_BOOT_Right_COL_SPE... 256x256
tex_M_BUCKLE_COL_SPEC_tga 512x256
tex_M_BUCKLE_NORMAL_tga 512x256
tex_M_ChestStraps_BUM_tga 512x256
tex_M_ChestStraps_COL_VAR_... 512x256
tex_M_EYE_Bits_BOTH_COL_S... 256x256
tex_M_EYE_Bits_BOTH_OPAC... 256x256
Ready

Properties

ms_helmet

Material Parameters

WorldITXf	0.035 0.000821 0.0486 0 -0.0371 0.0391 0.0261
WVPXf	-23.4 -27 7.14 -204 -27.1 29.5 22.5 -380 8.16 -3.3
WorldXf	9.76 0.229 13.6 0 -10.3 10.9 7.27 0 -8.75 -12.6 6
ViewIXf	-0.227 0.0615 -0.972 -9.92 -1.09e-07 0.998 0.063
Light0 Pos	0 0 0
Light 0	□ 1 1 1
Light1 Pos	0 0 0
Light 1	□ 1 1 1
Diffuse Brightness	0.2
Specular Brightness	0.8
Specular Power	20
Ambient Light	■ 0.07 0.07 0.07
Debug UV Scale	1
Surface Texture	tex_M_HELMET_COL_SPEC_tga (M_HELM
Bumpiness	0.3
Normal Map	Image3 (default_bump_normal.dds)

Material Properties

Name	ms_helmet
Effect	file:///C:/Program Files/NVIDIA Corporation/FX C

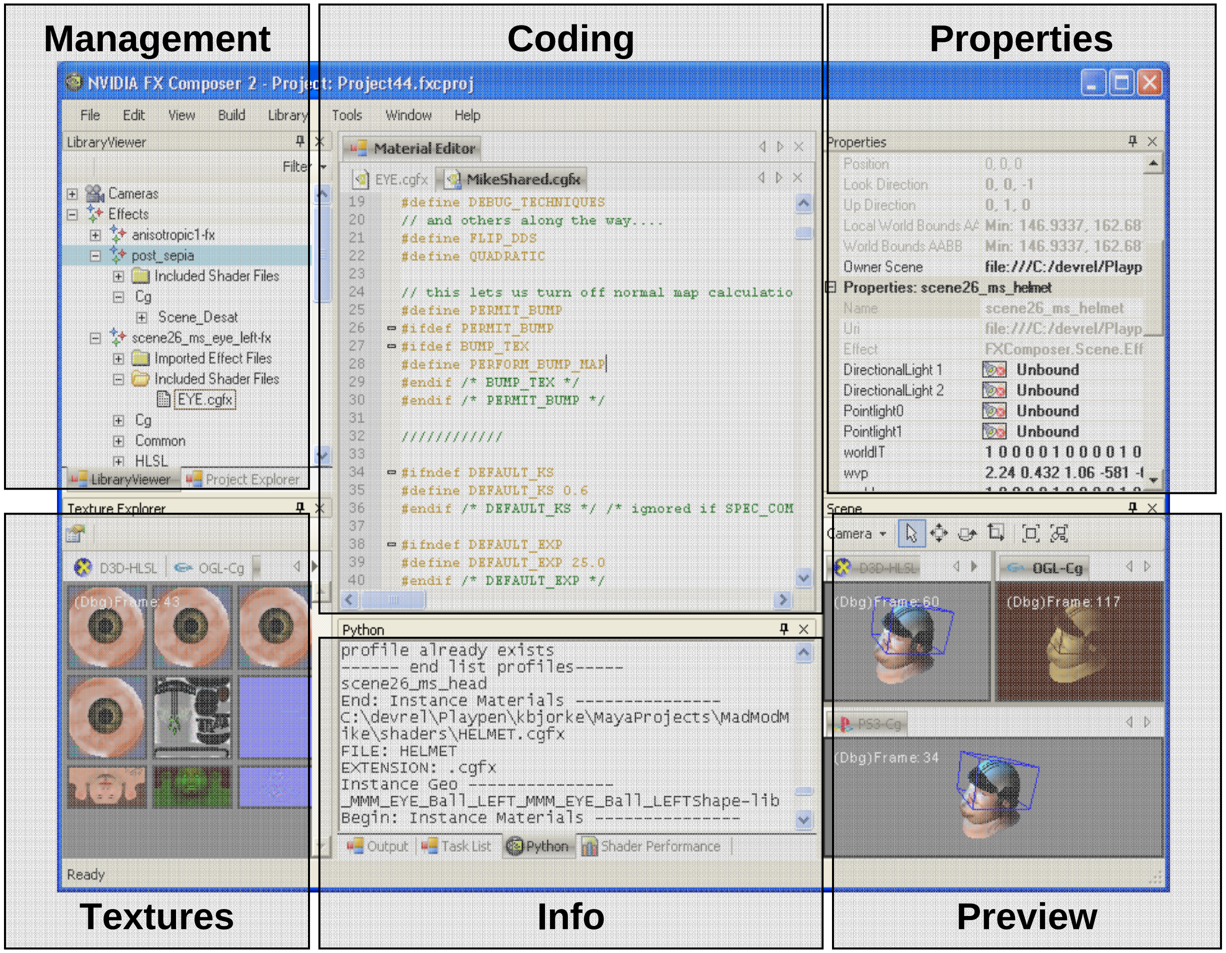
Material Scene Bindings

pointlight0	directionalLightShape1
pointlight1	directionalLightShape1

Image courtesy of NVIDIA

User Interface Overview

- Management
  - Project Explorer
  - Library Viewer
- Coding
  - Modify shader code
- Properties
  - Modify parameter values
- Preview
  - Multiple platform views
- Info
  - Tasks and errors
  - Python scripting
  - Shader Performance
- Textures
  - Explore image resources
  - View render targets



```
#define DEBUG_TECHNIQUES
// and others along the way....
#define FLIP_DDS
#define QUADRATIC

// this lets us turn off normal map calculatio
#define PERMIT_BUMP
#ifdef PERMIT_BUMP
#ifdef BUMP_TEX
#define PERFORM_BUMP_MAP
#endif /* BUMP_TEX */
#endif /* PERMIT_BUMP */

/////////////

#ifndef DEFAULT_KS
#define DEFAULT_KS 0.6
#endif /* DEFAULT_KS */ /* ignored if SPEC_COM

#ifndef DEFAULT_EXP
#define DEFAULT_EXP 25.0
#endif /* DEFAULT_EXP */
```

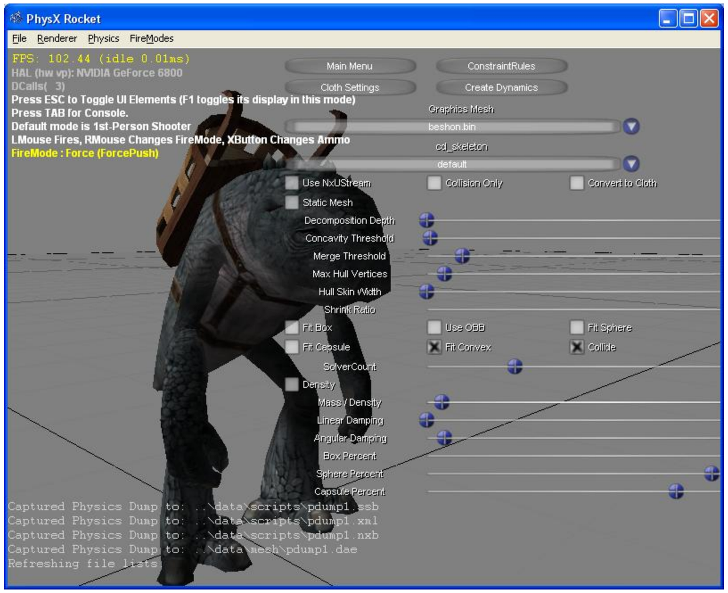
```
profile already exists
------ end list profiles-----
scene26_ms_head
End: Instance Materials ----------------
C:\devrel\Playpen\kbjorke\MayaProjects\MadModM
ike\shaders\HELMET.cgfx
FILE: HELMET
EXTENSION: .cgfx
Instance Geo ----------------
_MMM_EYE_Ball_LEFT_MMM_EYE_Ball_LEFTShape-lib
Begin: Instance Materials ----------------
```

Position	0, 0, 0
Look Direction	0, 0, -1
Up Direction	0, 1, 0
Local World Bounds AA	Min: 146.9337, 162.68
World Bounds AABB	Min: 146.9337, 162.68
Owner Scene	file:///C:/devrel/Playp
Properties: scene26_ms_helmet	
Name	scene26_ms_helmet
Uri	file:///C:/devrel/Playp
Effect	FXComposer.Scene.Eff
DirectionalLight 1	Unbound
DirectionalLight 2	Unbound
Pointlight0	Unbound
Pointlight1	Unbound
worldIT	1 0 0 0 0 1 0 0 0 0 1 0
wvp	2.24 0.432 1.06 -581 -(

TOOLS

- Edit/import/export COLLADA FX
  - FXComposer 2.0
  - 3dsMax
  - Maya
  - Softimage|XSI
- Parameters are expressed outside of the shader programs
  - Tools can deal with FX even if they cannot use the programs
    - i.e. 3dsMax supports only HLSL
      - Note – latest version has now Cg support as well
- COLLADA supports multiple profile
  - CG, GLSL, GLSL ES, … can all be specified
  - Same FX definition, different languages
  - Extensible – Renderman extension available
  -

COLLADA Physics



```
FPS: 102.44 (idle 0.01ms)
HAL (hw vp): NVIDIA GeForce 6800
DCalls(  3)
Press ESC to Toggle UI Elements (F1 toggles its display in this mode)
Press TAB for Console.
Default mode is 1st-Person Shooter
LMouse Fires, RMouse Changes FireMode, XButton Changes Ammo
FireMode : Force (ForcePush)
```

Graphics Mesh: beshon.bin

cd_skeleton: default

□ Use NxUStream □ Collision Only □ Convert to Cloth

□ Static Mesh

Decomposition Depth

Concavity Threshold

Merge Threshold

Max Hull Vertices

Hull Skin Width

Shrink Ratio

□ Fit Box □ Use OBB □ Fit Sphere

□ Fit Capsule ☒ Fit Convex ☒ Collide

SolverCount

□ Density

Mass / Density

Linear Damping

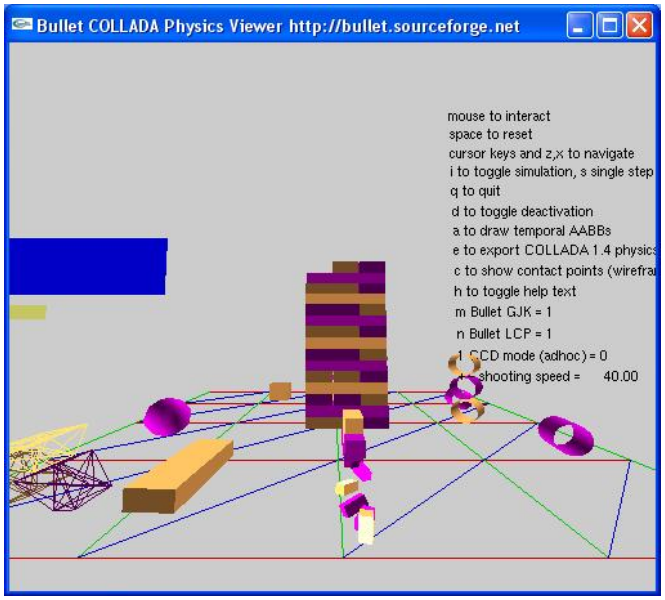
Angular Damping

Box Percent

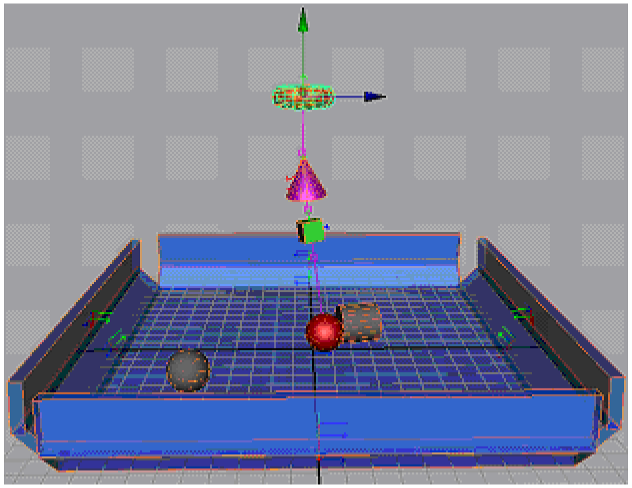
Sphere Percent

Capsule Percent

```
Captured Physics Dump to: ..\data\scripts\pdump1.ssb
Captured Physics Dump to: ..\data\scripts\pdump1.xml
Captured Physics Dump to: ..\data\scripts\pdump1.nxb
Captured Physics Dump to: ..\data\mesh\pdump1.dae
Refreshing file lists
```

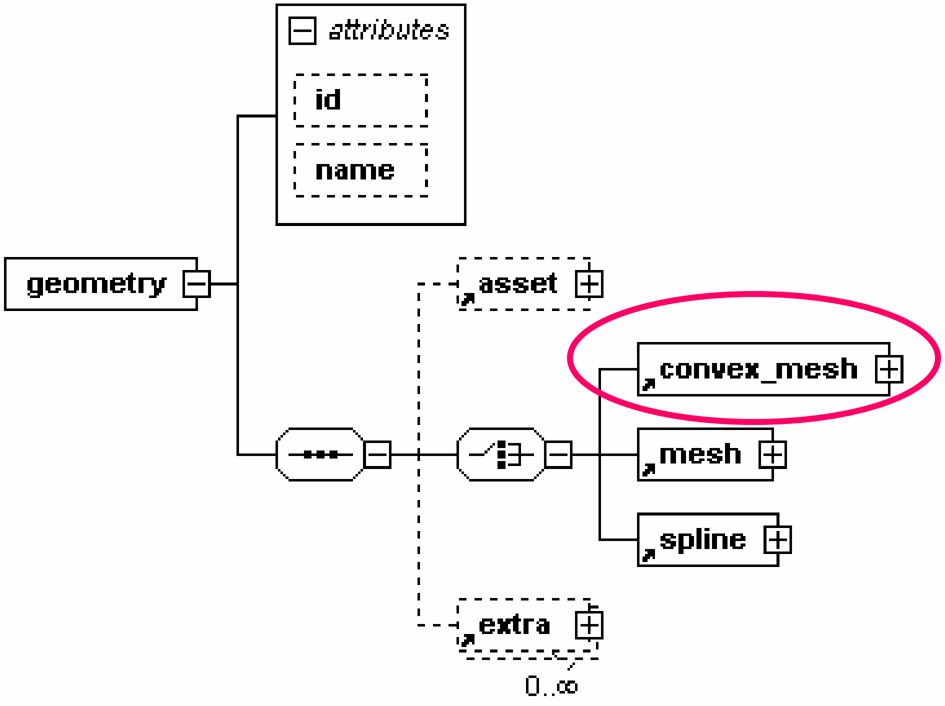


```
mouse to interact
space to reset
cursor keys and z,x to navigate
i to toggle simulation, s single step
q to quit
d to toggle deactivation
a to draw temporal AABBs
e to export COLLADA 1.4 physics
c to show contact points (wirefra
h to toggle help text
m Bullet GJK = 1
n Bullet LCP = 1
1 CCD mode (adhoc) = 0
+- shooting speed =     40.00
```

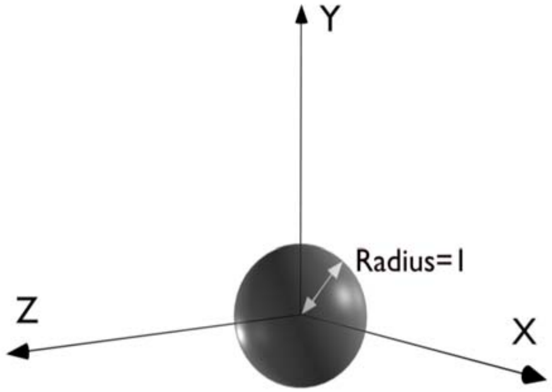


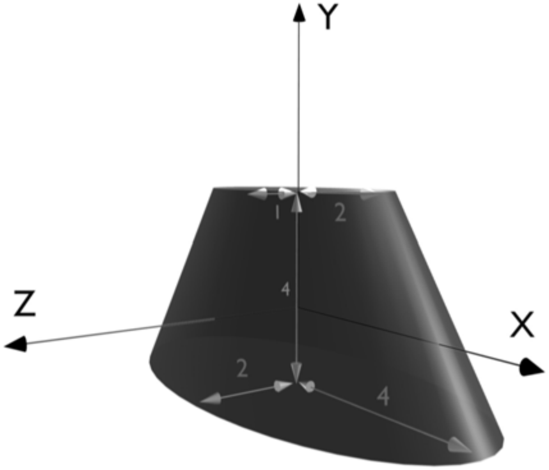
Images courtesy of Feeling Software (www.feelingsoftware.com), Bullet physics library (www.continuousphysics.com/Bullet/) and AGEIA (www.ageia.com)

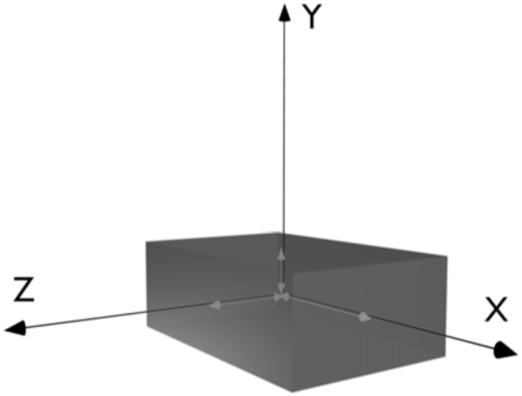
<geometry>

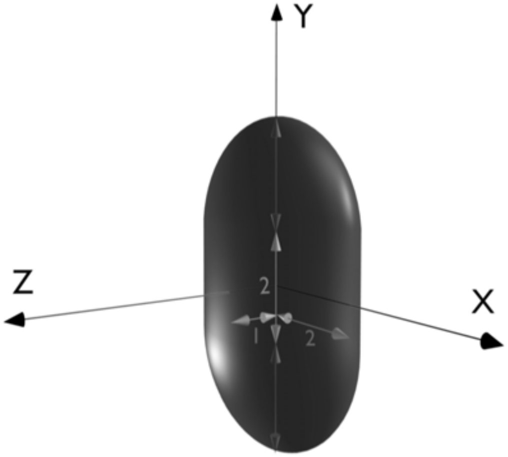

attributes
id
name
geometry
asset
convex_mesh
mesh
spline
extra
0..∞

Analytic shapes – for physics only









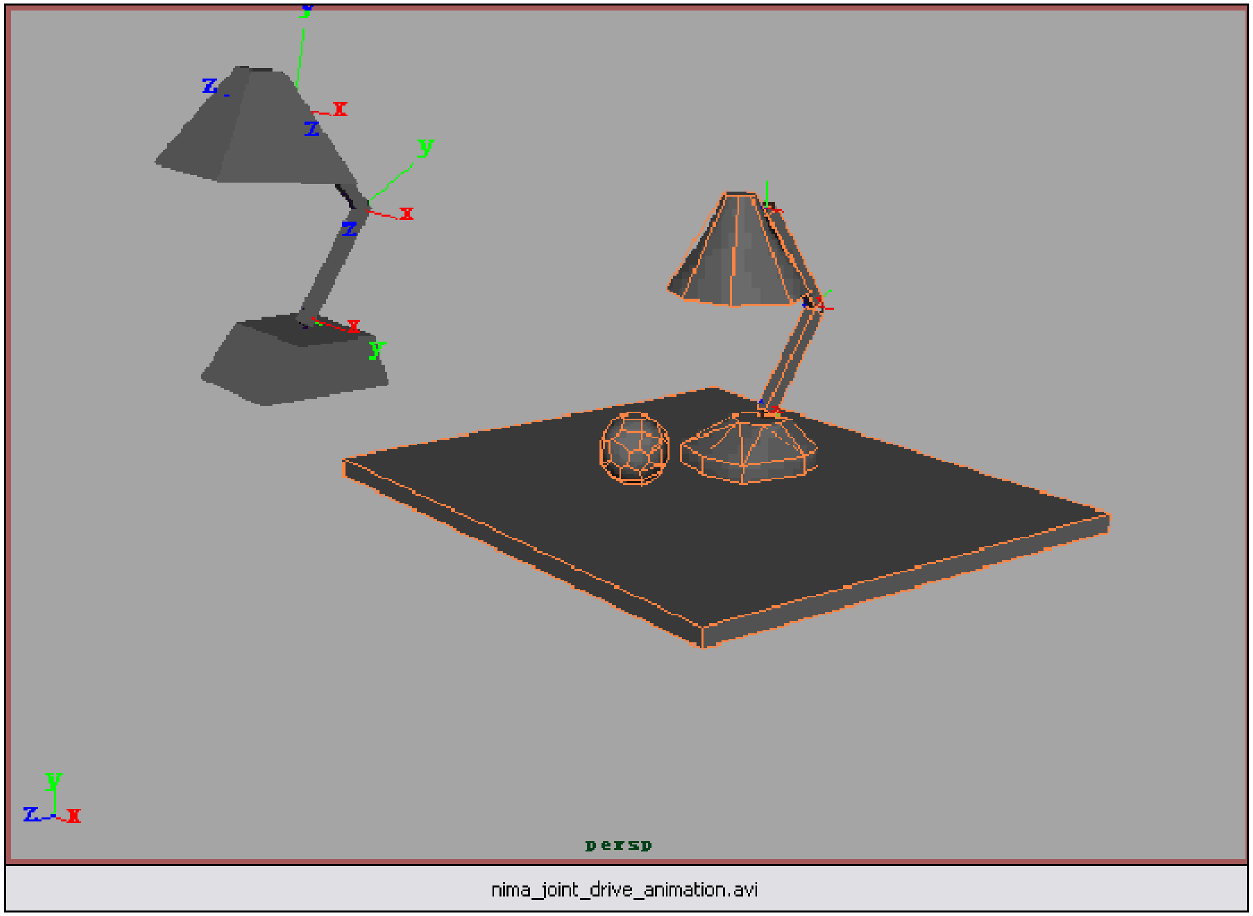
Tools and engines

- COLLADA Physics plug-ins for 3dsMax, Maya, Blender.
- Bullet engine is an open source physics engine available for many platforms, including the PLAYSTATION 3. It provides a physics viewer, which displays only the analytic shapes and show the effect of the physics, which can be used as a tool by a developer, even if using a different engine for the target platform.
- Both Bullet and AGEIA physics API offer the ability to export a COLLADA Physics document at any time during the simulation. This document contains all the values at the specific simulation time, providing a full snapshot of the physics engine. Those snapshots are really helpful to debug physics issues. Most issues are the result of incorrect data, that can be difficult to find without the COLLADA Physics snapshot and COLLADA Physics specialized viewers and tools.
- The AGEIA PhysX Rocket tool can import and export COLLADA documents with physics, as well as present an interface to let developers tweak any physical parameters.

Toward a PhysicsComposer tool ?

COLLADA Physics and Animation

- COLLADA defines the constrained joints and rigid body properties
- COLLADA also store the animation that will be applied to some of the joints (motor), the physics engine does the rest of the animation



nima_joint_drive_animation.avi

Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session

Viewer lunar.avi

Image courtesy of Feeling Software (www.feeelingsoftwrea.com)

1.4.x Features

- Mesh Geometry
- Transform hierarchy (rotation, translation, shear, scale, matrix)
- Materials
- Textures
- Shader programs (Cg, GLSL, HLSL, GLES)
- Shader effects (FX)
- Lights
- Cameras
- Skinning / bones
- Animation
- Physics (rigid bodies, constraints, rag dolls, collision volumes)
- Instancing
- Techniques
- Multi-representations
- Assets
- User data

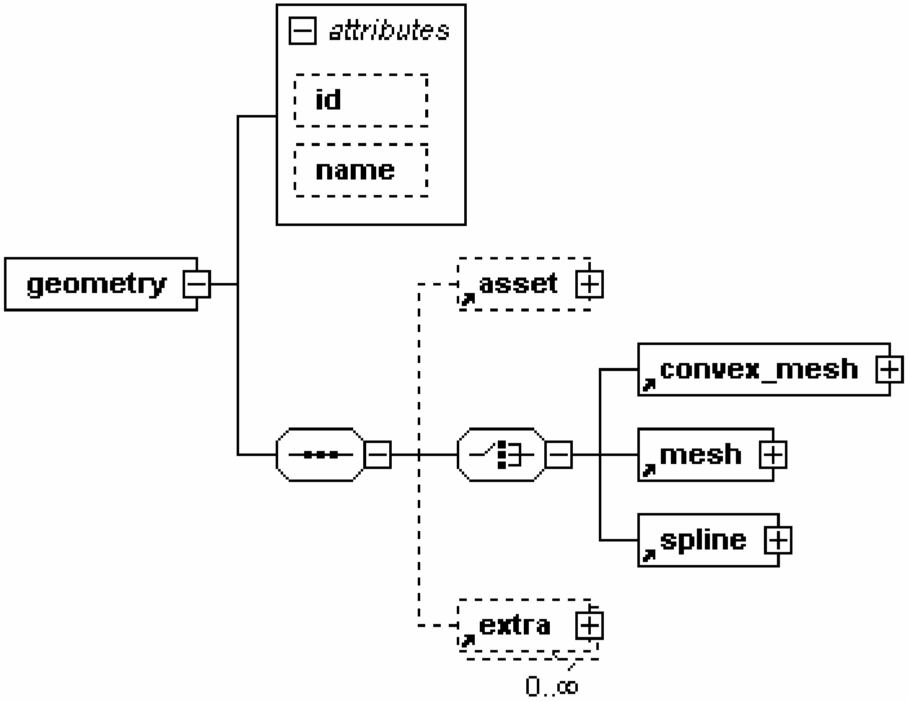
Geometry Schema

- Geometry as data-flow
  - Vertices are still the norm.
    - Attributes are variable.
  - Meshes contain vertices
    - Describes the shape / topology of the geometry.
    - Lots of high frequency data.

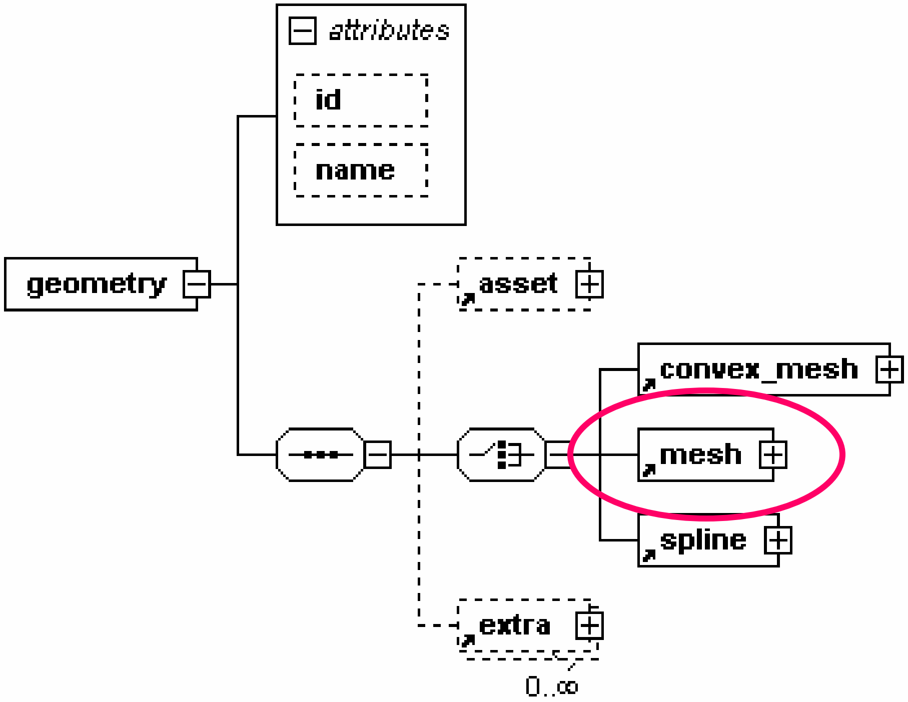
<accessor> (<source>) ➔ <input> (<mesh>)

<geometry>

- Geometry element
  - Contains various types of descriptions.
  - Instantiated into the scene.
  - Stored in the **<library_geometies>** element

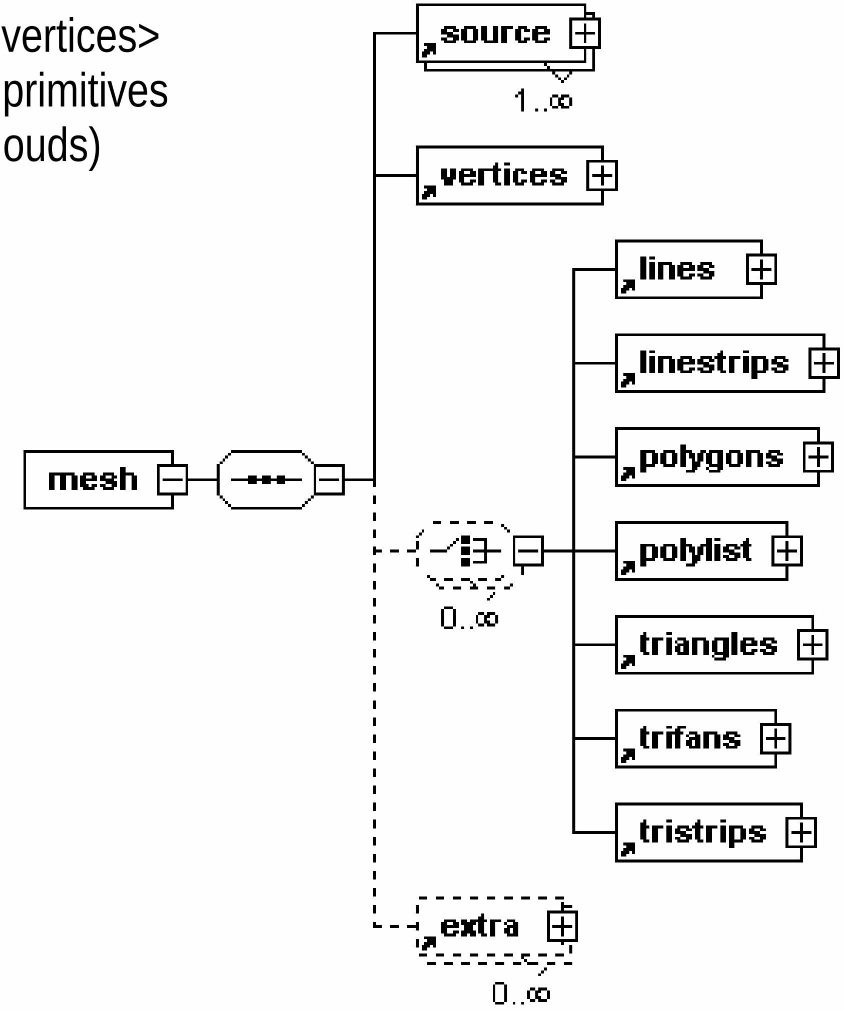


<geometry>

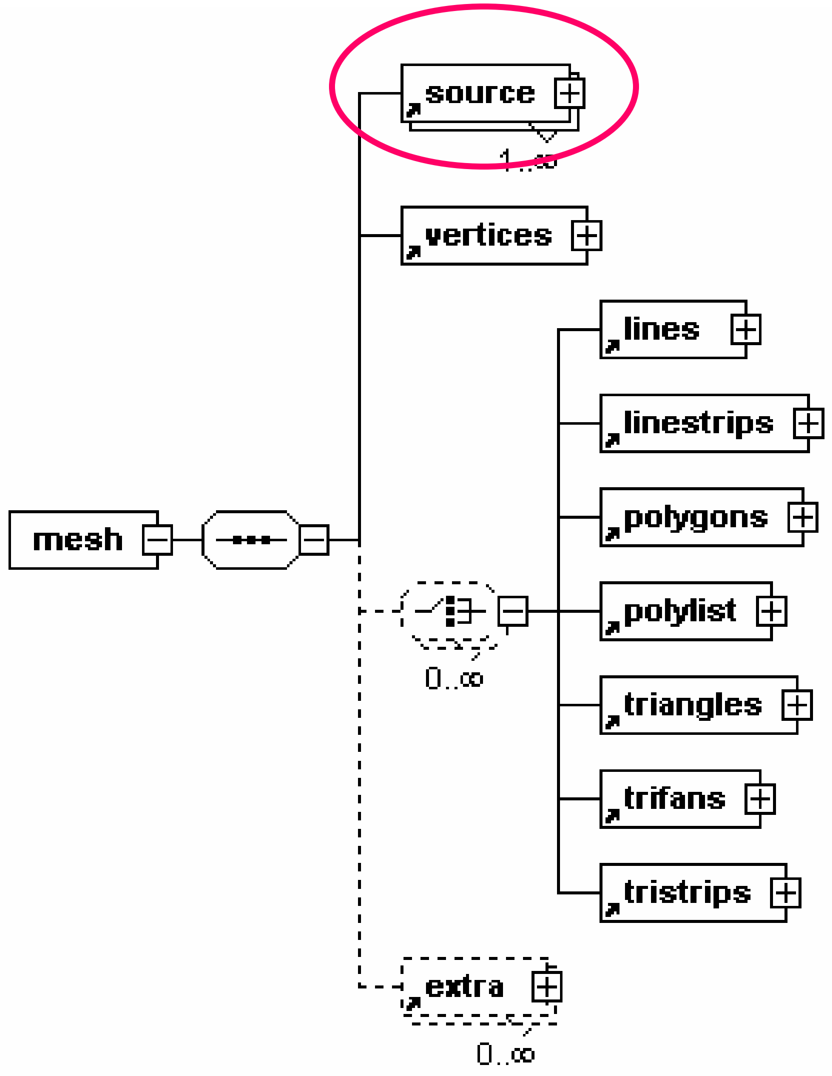

attributes
id
name
geometry
asset
convex_mesh
mesh
spline
extra
0..∞

<mesh>

- Mesh element
  - One or more <source>
  - Exactly one <vertices>
  - Zero or more primitives (e.g. point clouds)



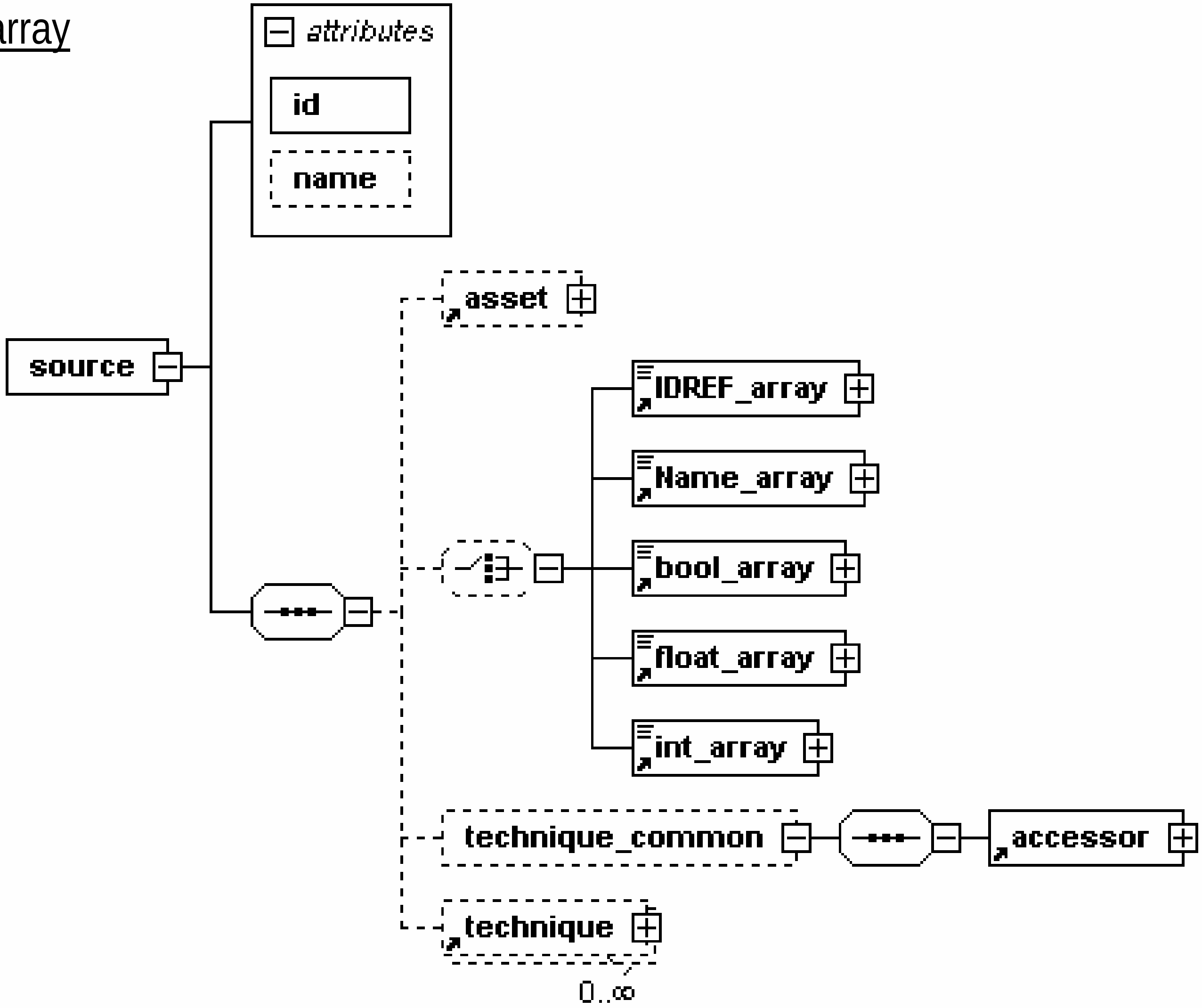
<mesh>



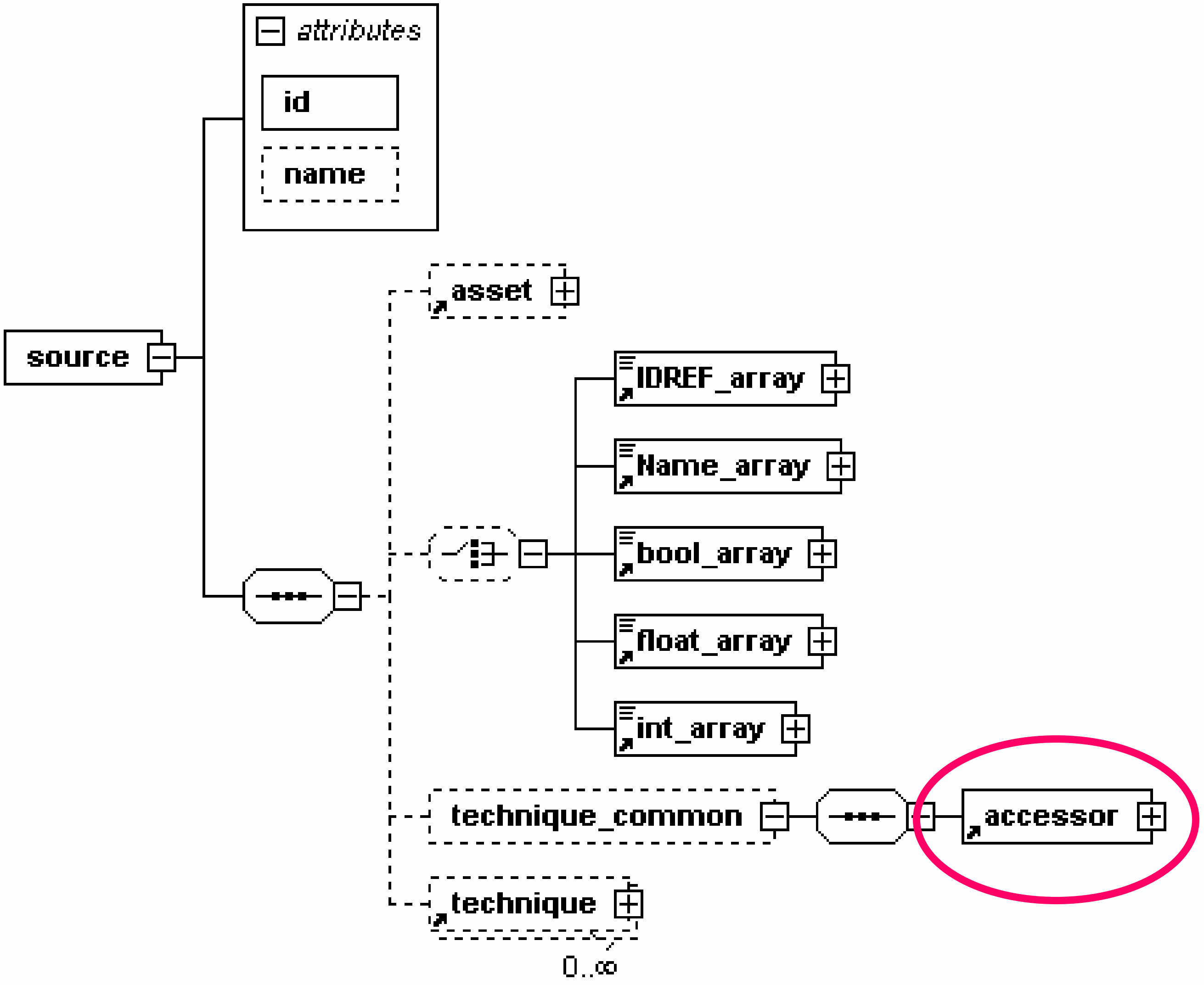
source
1..∞
vertices
lines
linestrips
polygons
polylist
triangles
trifans
tristrips
mesh
0..∞
extra
0..∞

<source>

- Data source element
  - Raw data stream
  - Strongly typed array
  - No semantics

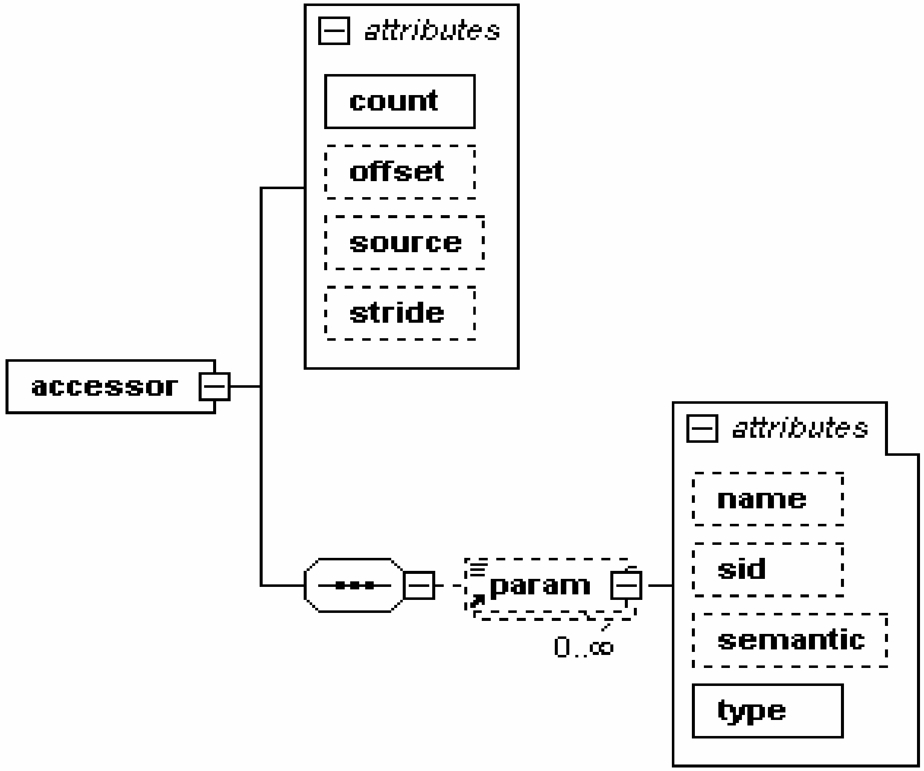


<source>


attributes
id
name
source
asset
IDREF_array
Name_array
bool_array
float_array
int_array
technique_common
accessor
technique
0..∞

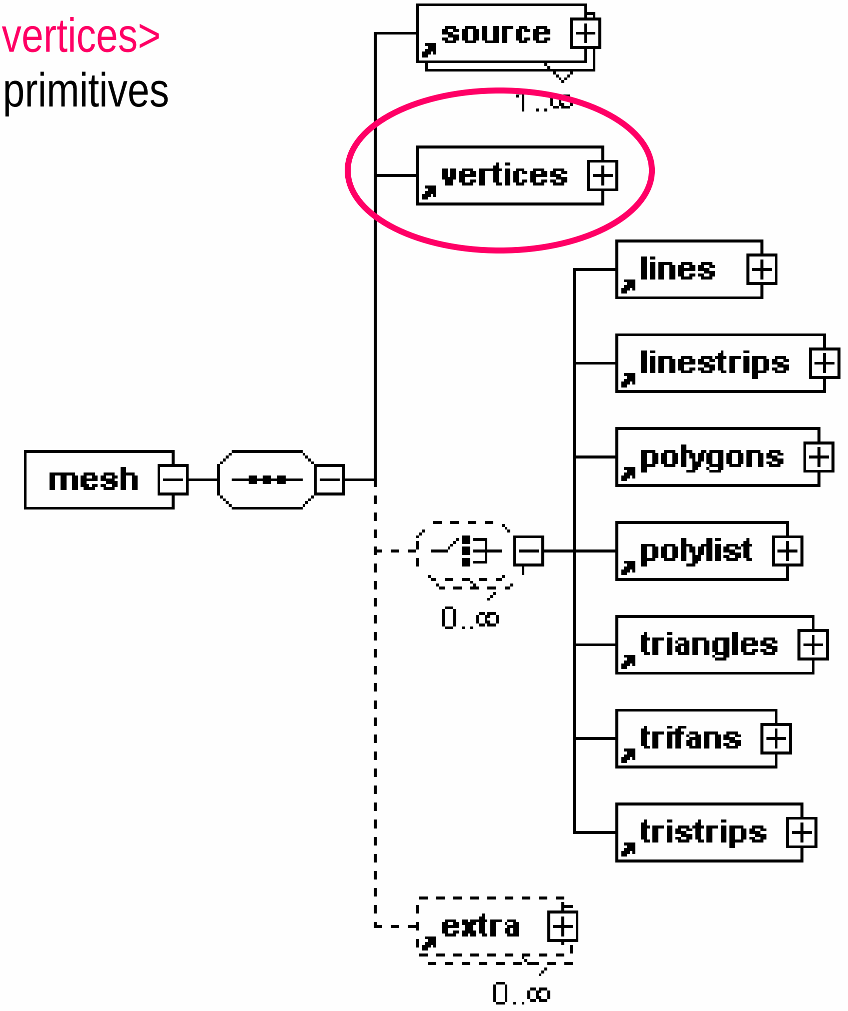
<accessor>

- Data accessor element
  - <source> output interface
  - Describes how to access the source data
  - Data can be stored externally



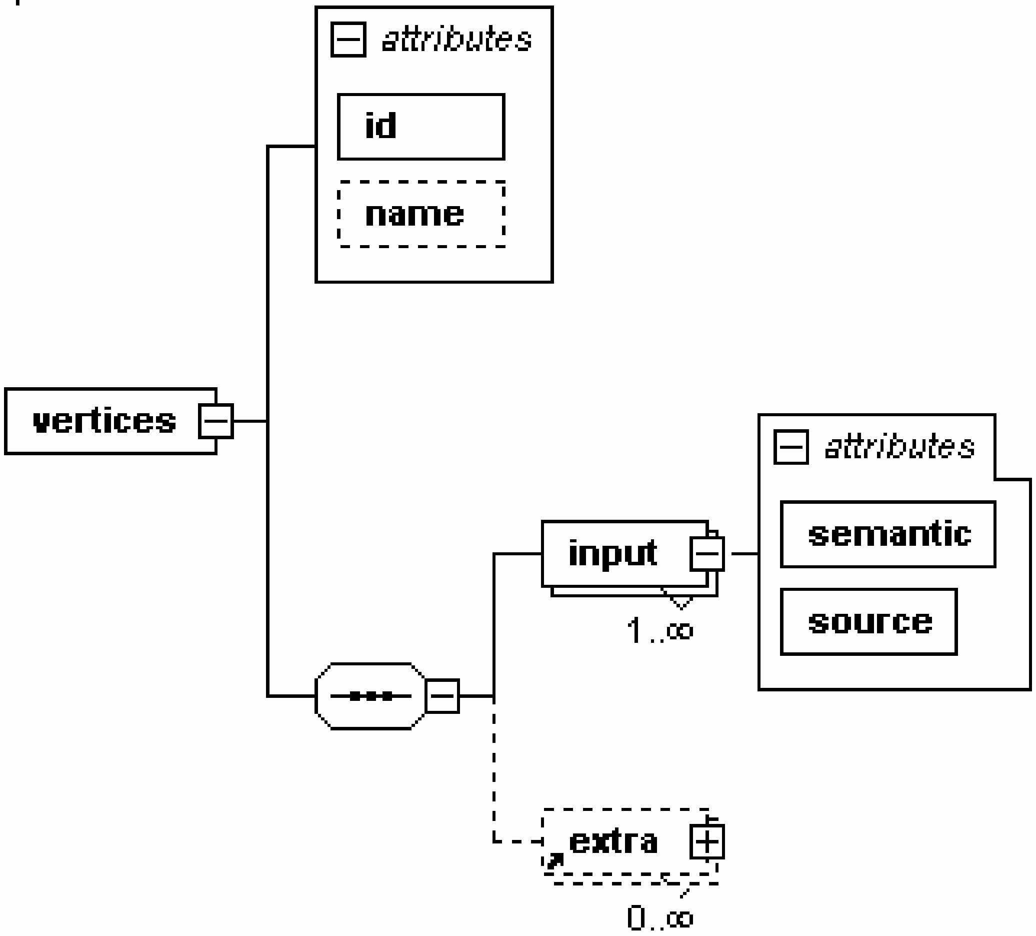
<mesh>

- Mesh element
  - One or more <source>
  - Exactly one <vertices>
  - Zero or more primitives

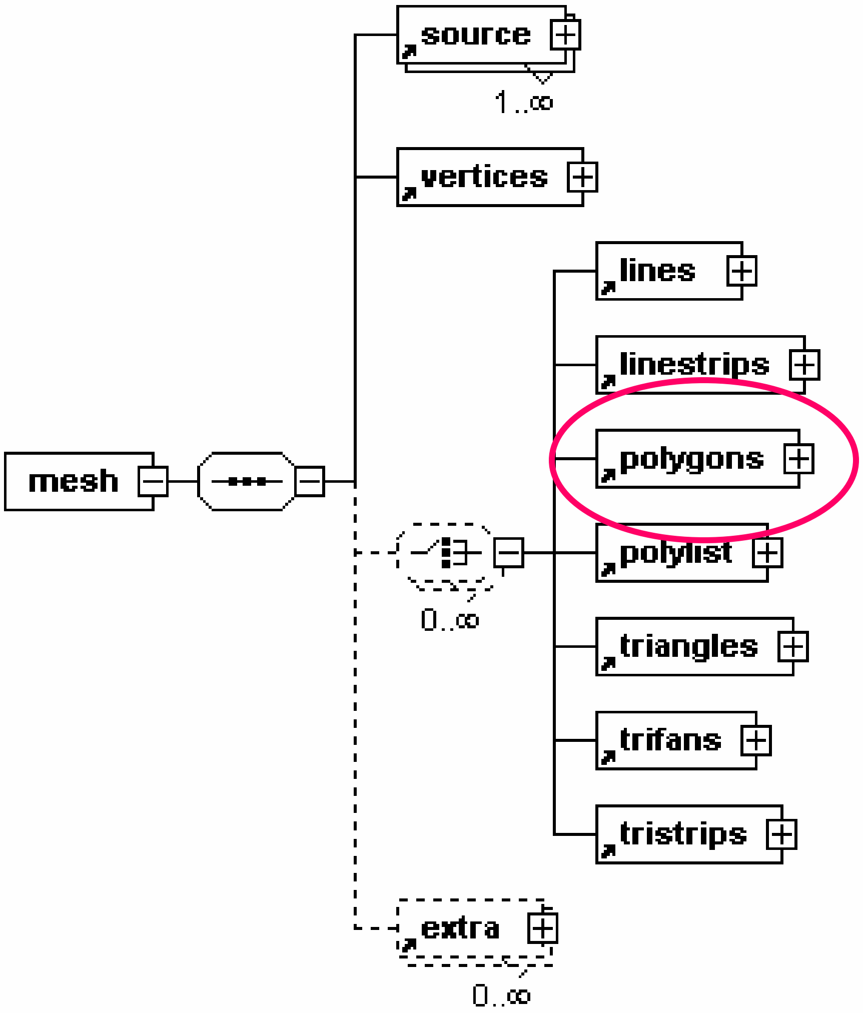


<vertices>

- Mesh vertex element
  - Groups mesh vertex attribute data
  - Collect data independent to primitive assembly
  - “POSITION” semantic required
  - Unlimited attributes
  - nDimentional data OK

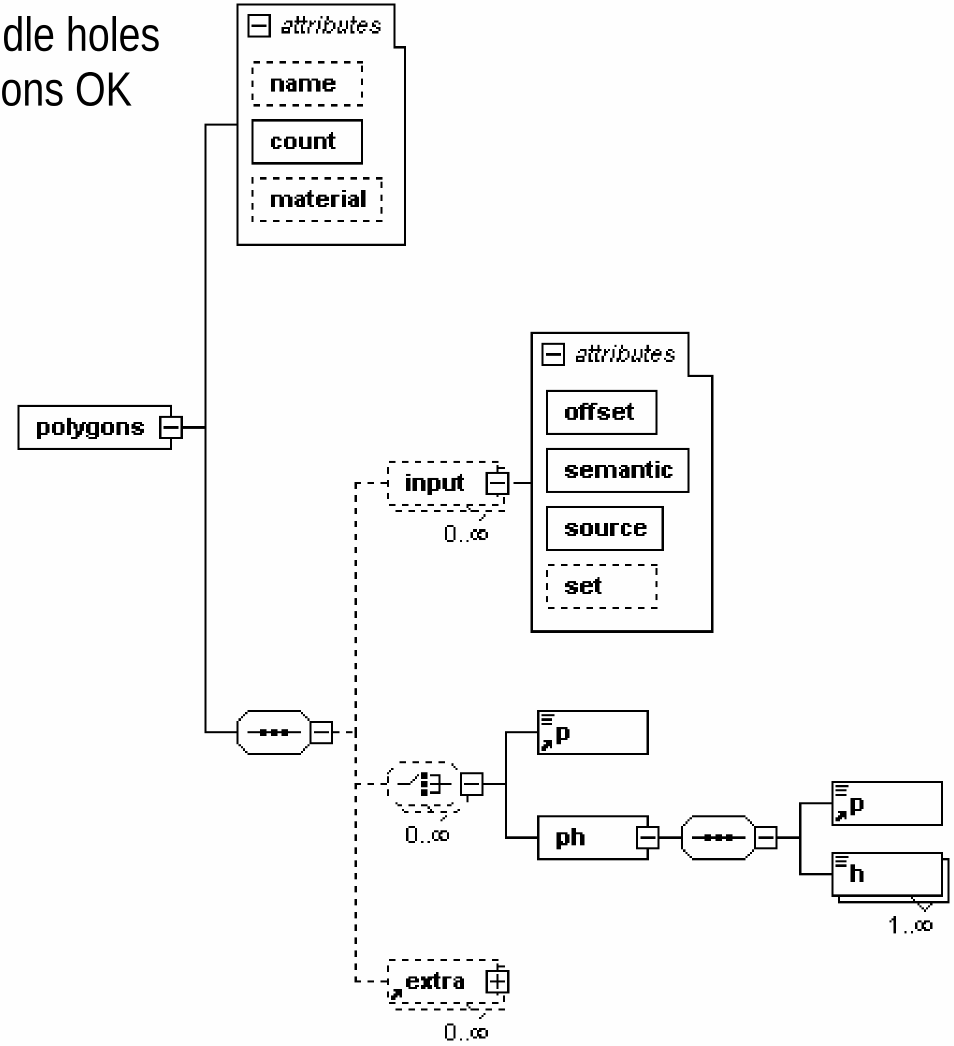


<mesh>


source
1..∞
vertices
lines
linestrips
polygons
mesh
polylist
0..∞
triangles
trifans
tristrips
extra
0..∞

COLLADA <polygons>

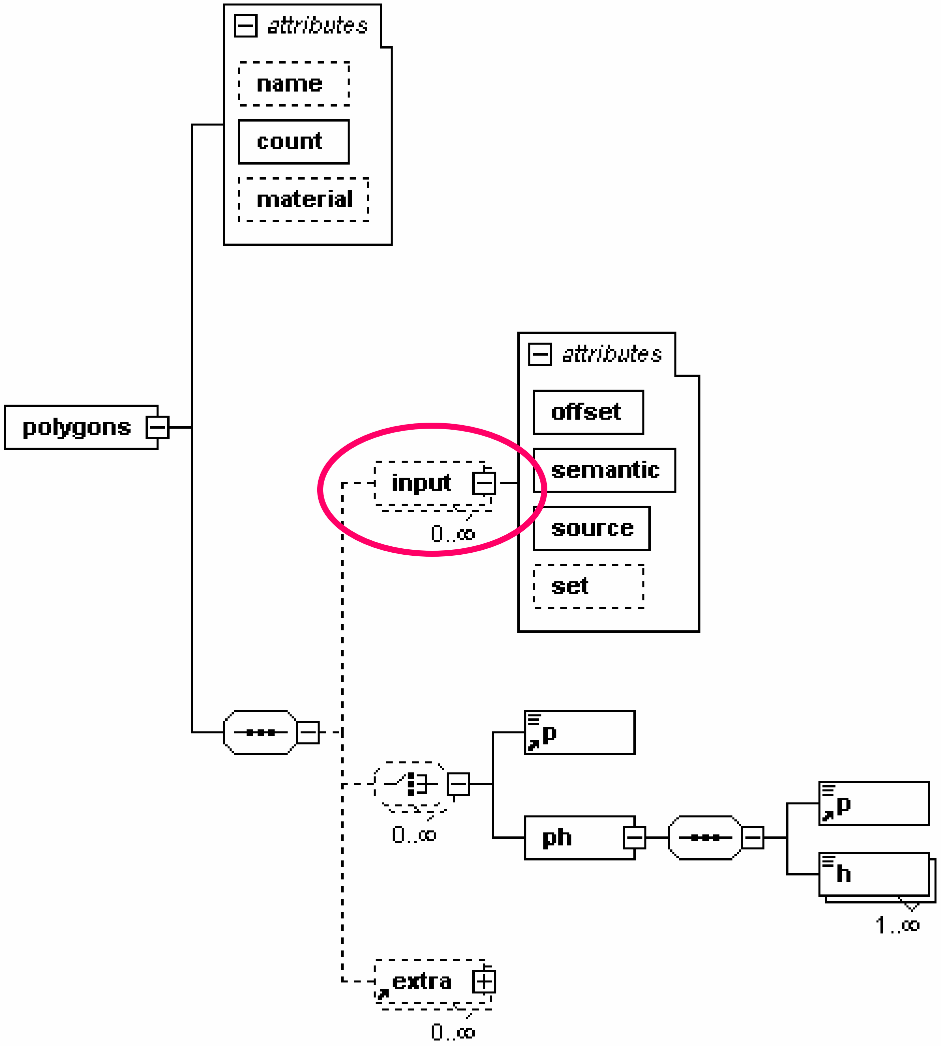
- Polygons element
  - Collates vertices into primitives.
  - General primitive, can handle holes
  - Complex non planar polygons OK



COLLADA <polygons>

- Input streams
  - Possible Input semantics:

VERTEX
TEXCOORD
NORMAL
COLOR
BINORMAL
UV
…



Geometry example (1)

- A super cool box

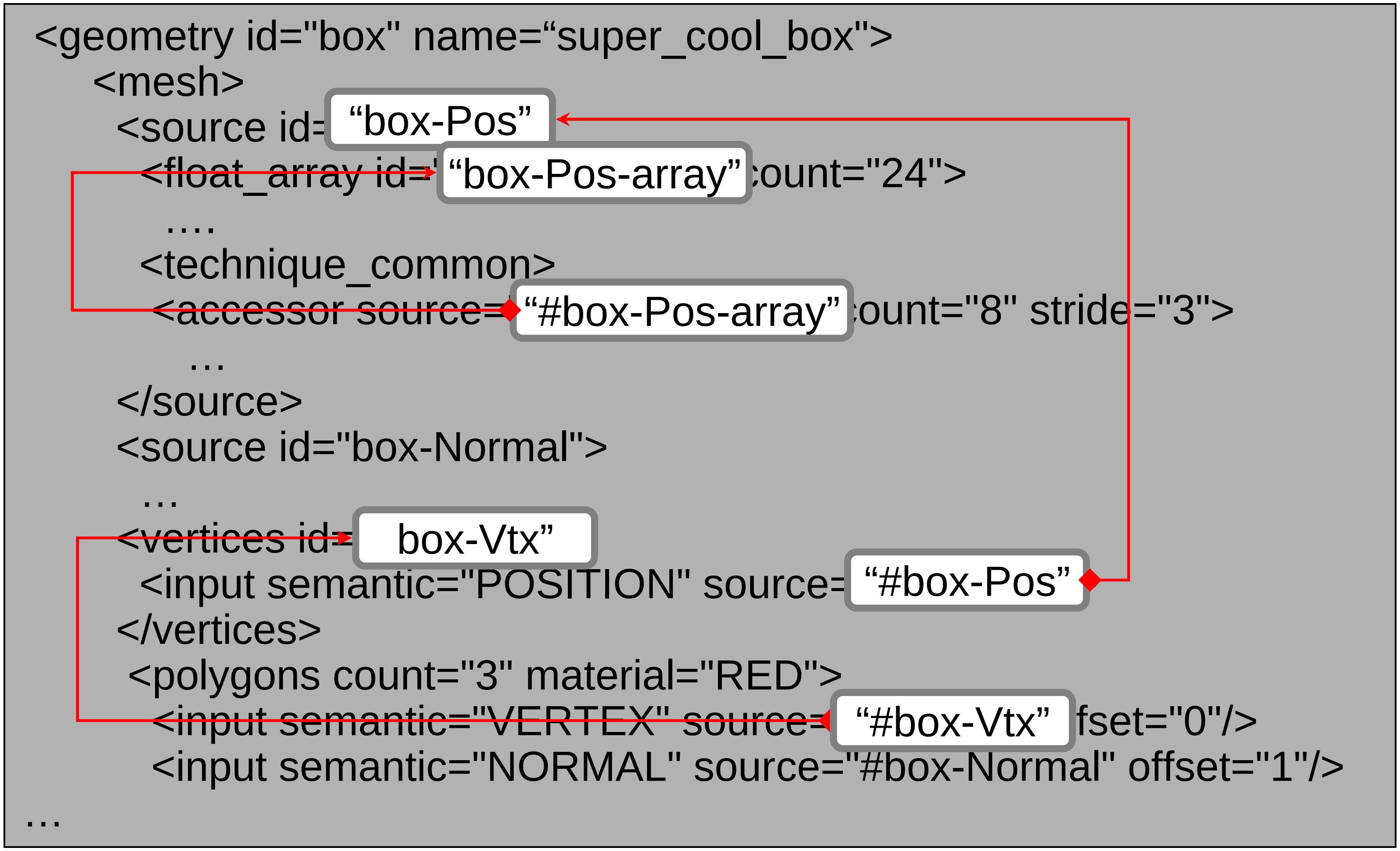
```
<geometry id="box" name=“super_cool_box">
     <mesh>
       <source id="box-Pos">
         <float_array id="box-Pos-array" count="24">
           -0.5  0.5  0.5  0.5  0.5  0.5 -0.5 -0.5  0.5  0.5 -0.5  0.5
           -0.5  0.5 -0.5  0.5  0.5 -0.5 -0.5 -0.5 -0.5  0.5 -0.5 -0.5
         </float_array>
         <technique_common>
          <accessor source="#box-Pos-array" count="8" stride="3">
            <param name="X" type="float"/>
            <param name="Y" type="float"/>
            <param name="Z" type="float"/>
          </accessor>
         </technique_common>
       </source>
       <source id=“box-0-Normal”>
…..
```

Geometry example (2)

```
…
  <vertices id="box-Vtx">
   <input semantic="POSITION" source="#box-Pos"/>
  </vertices>
  <polygons count="3" material="RED">
   <input semantic="VERTEX" source="#box-Vtx" offset="0"/>
   <input semantic="NORMAL" source="#box-0-Normal" offset="1"/>
   <p>0 4 2 4 3 4 1 4</p> <p>0 2 1 2 5 2 4 2</p> <p>6 3 7 3 3 3 2 3</p>
  </polygons>
  <polygons count="3" material="BLUE">
   <input semantic="VERTEX" source="#box-Vtx" offset="0"/>
   <input semantic="NORMAL" source="#box-0-Normal" offset="1"/>
   <p>0 1 4 1 6 1 2 1</p> <p>3 0 7 0 5 0 1 0</p> <p>5 5 7 5 6 5 4 5</p>
  </polygons>
 </mesh>
</geometry>
```

Geometry example (3)

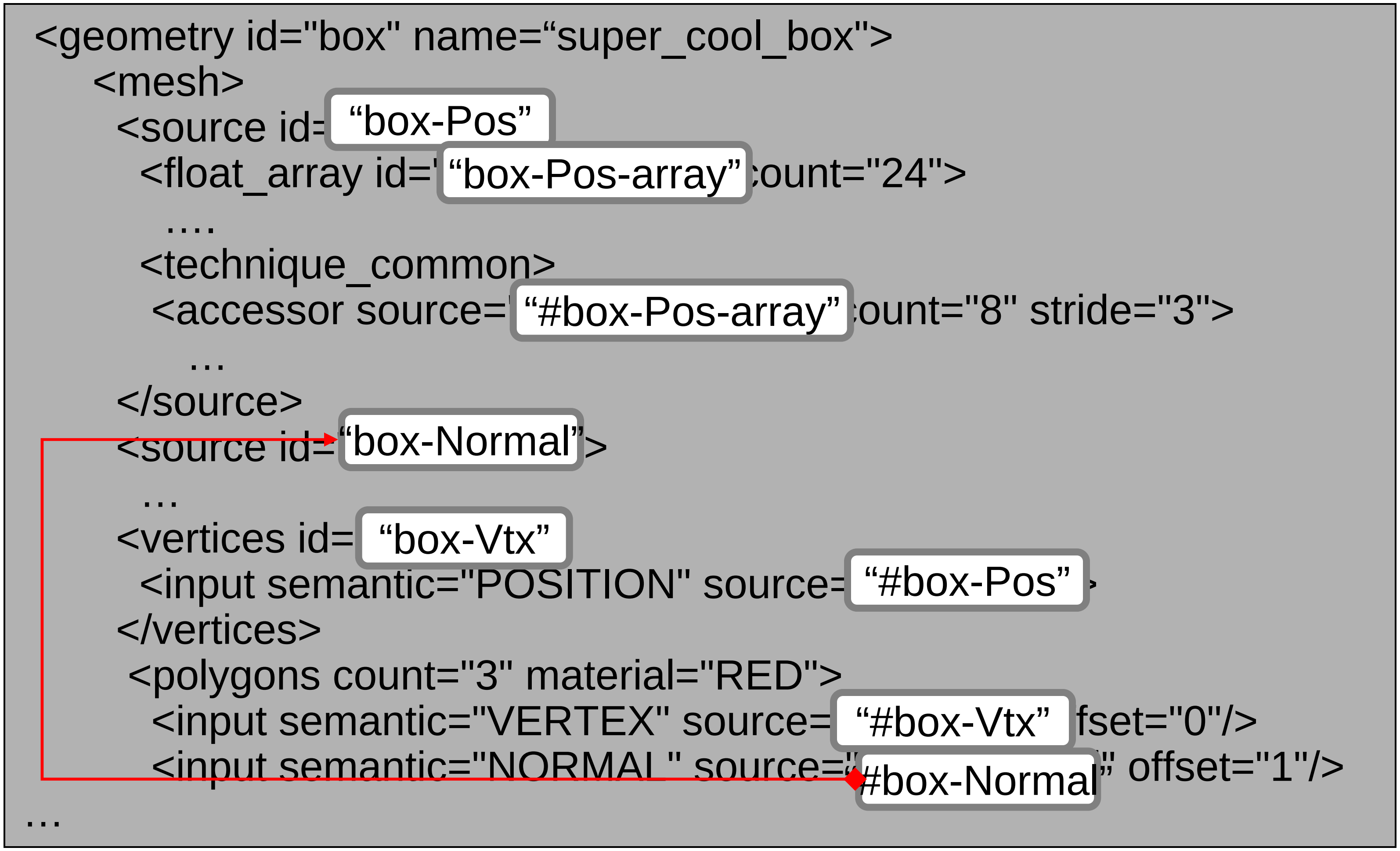
- Vertex data stream



```
<geometry id="box" name=“super_cool_box">
     <mesh>
       <source id= “box-Pos”
         <float_array id=' “box-Pos-array” count="24">
           ….
         <technique_common>
          <accessor source= “#box-Pos-array” count="8" stride="3">
            …
       </source>
       <source id="box-Normal">
         …
       <vertices id= box-Vtx”
         <input semantic="POSITION" source= “#box-Pos”
       </vertices>
        <polygons count="3" material="RED">
          <input semantic="VERTEX" source= “#box-Vtx” fset="0"/>
          <input semantic="NORMAL" source="#box-Normal" offset="1"/>
…
```

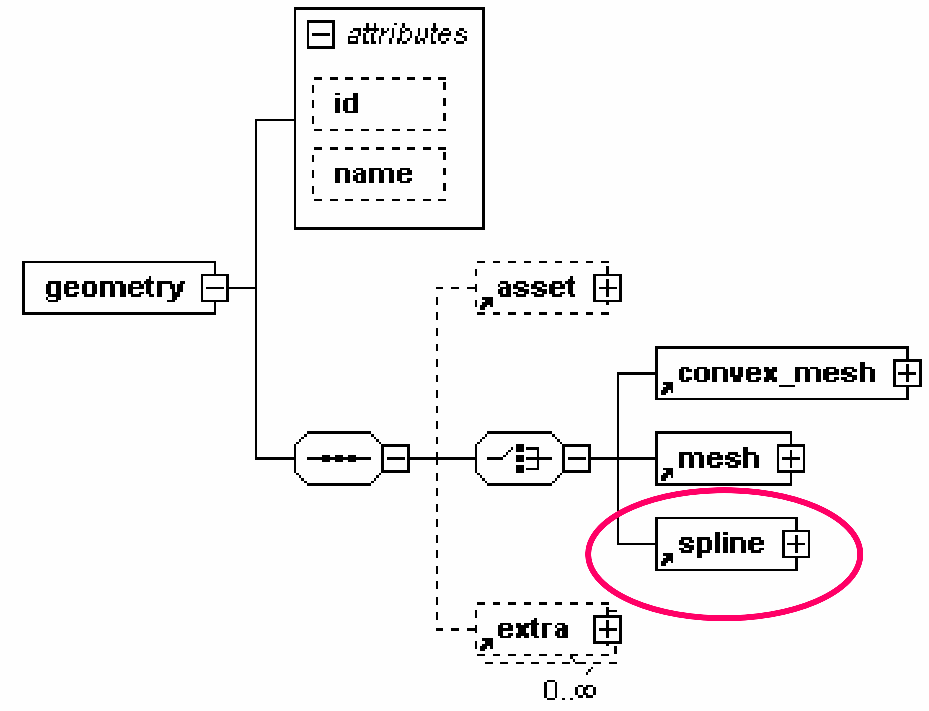
Geometry example (4)

- Normal data stream

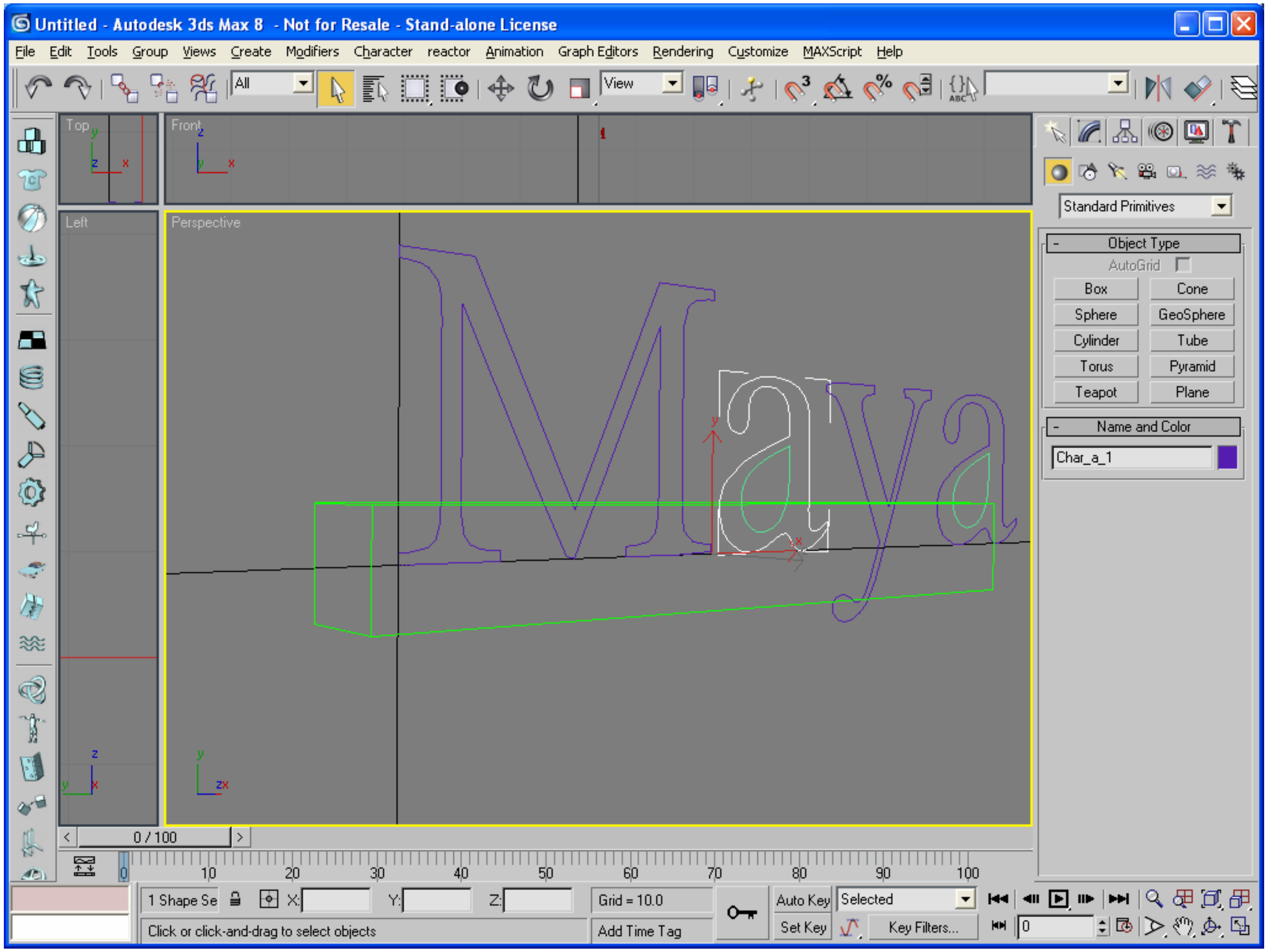


```
<geometry id="box" name=“super_cool_box">
    <mesh>
      <source id= “box-Pos”
        <float_array id=' “box-Pos-array” :ount="24">
          ….
        <technique_common>
         <accessor source=' “#box-Pos-array” :ount="8" stride="3">
            …
      </source>
      <source id=“box-Normal”>
        …
      <vertices id= “box-Vtx”
        <input semantic="POSITION" source= “#box-Pos” >
      </vertices>
       <polygons count="3" material="RED">
         <input semantic="VERTEX" source= “#box-Vtx” fset="0"/>
         <input semantic="NORMAL" source=“#box-Normal” offset="1"/>
…
```

<geometry>


attributes
id
name
geometry
asset
convex_mesh
mesh
spline
extra
0..∞

NURBS curves imported from Maya into 3dsMax



Char_a_1

1 Shape Se

Grid = 10.0

Selected

0

Click or click-and-drag to select objects

1.4.x Features

- Mesh Geometry
- Transform hierarchy (rotation, translation, shear, scale, matrix)
- Materials
- Textures
- Shader programs (Cg, GLSL, HLSL, GLES)
- Shader effects (FX)
- Lights
- Cameras
- Skinning / bones
- Animation
- Physics (rigid bodies, constraints, rag dolls, collision volumes)
- Instancing
- Techniques
- Multi-representations
- Assets
- User data

Material

- Assign material to geometry symbolically

```
…
  <vertices id="box-Vtx">
   <input semantic="POSITION" source="#box-Pos"/>
  </vertices>
  <polygons count="3" material="RED">
   <input semantic="VERTEX" source="#box-Vtx" offset="0"/>
   <input semantic="NORMAL" source="#box-0-Normal" offset="1"/>
   <p>0 4 2 4 3 4 1 4</p> <p>0 2 1 2 5 2 4 2</p> <p>6 3 7 3 3 3 2 3</p>
  </polygons>
  <polygons count="3" material="BLUE">
   <input semantic="VERTEX" source="#box-Vtx" offset="0"/>
   <input semantic="NORMAL" source="#box-0-Normal" offset="1"/>
   <p>0 1 4 1 6 1 2 1</p> <p>3 0 7 0 5 0 1 0</p> <p>5 5 7 5 6 5 4 5</p>
  </polygons>
 </mesh>
</geometry>
```

Instance and material binding

- <library_visual_scenes>
  - Document can store many scenes
- <scene>
  - Default scene for this document
  - <instance_visual_scene url="#...
- <node name=“Box” id=“Box”>
  - Scene contains nodes
- <instance_geometry url=“box”>
  - Instance geometries from <library_geometries>
- <bind_material>
  - This is how instanced geometry get their final material
- <technique_common>
  <instance_material symbol="RED" target="#redMaterial"/>
  <instance_material symbol="BLUE" target="#blueMaterial"/>
  - Now RED and BLUE have a real material bound

<material> and <effect>

- Stored in the <library_materials>
- <material> is an instanced of an <effect>
- <effect> is stored in <library_effects>
- A document containing effects is called a COLLADA FX document
- In short an effect is a full list of commands for the GPU subsystem
- Since there is no standard, COLLADA FX is using profiles:
- CG profile (cross API/platform), GL 2.0 (GLSL), DX (HLSL), OpenGL ES 1.x, OpenGL ES 2.0 (GLSL ES)
- <profile_COMMON> effect defines some basic rendering for prototyping and basic interchange
  - *Constant, Lambert, Phong, Blinn*
  - Mandatory for conformant files, can have several profiles for same effect

1.4.x Features

- Mesh Geometry
- Transform hierarchy (rotation, translation, shear, scale, matrix)
- Materials
- Textures
- Shader programs (Cg, GLSL, HLSL, GLES)
- Shader effects (FX)
- Lights
- Cameras
- Skinning / bones
- Animation
- Physics (rigid bodies, constraints, rag dolls, collision volumes)
- Instancing
- Techniques
- Multi-representations
- Assets
- User data

Texture Mapping (1)

- <material>

```
<image id="img01">
   <init_from> image_file.tga</init_from>
... <effect> …
 <newparam sid="surface">
     <surface> <init_from>img01</init_from> </surface>
</newparam>
 <newparam sid="sampler">
   <sampler2D> <source>surface</source> </sampler2D>
 </newparam>
 <profile_COMMON> <phong>
    <diffuse> <texture texture="sampler" texcoord=“myUVs"/>
… <material> …
<material id="material_id">
 <instance_effect url="#effect_id" />
</material>
```

Texture Mapping (2)

- <geometry> has to have “TEXCOORD” *input* in addition to *material*

```
...<geometry>…
<geometry id=“geometry”>
  <input semantic=“TEXCOORD” source=“#...” set=“123”
  <triangles material=“material_symbol” …
</geometry>
```

Texture Mapping (3)

- Binding can now happen when instancing the geometry

```
...<geometry>…
<geometry id=“geometry”>
  <input semantic=“TEXCOORD” source=“#...” set=“123”
  <triangles material=“material_symbol” …
</geometry>

… <scene>…
<instance_geometry>
  <bind_material>
    <instance_material symbol="material_symbol" target="#material_id">
      <bind_vertex_input semantic="myUVs"
     input_semantic="TEXCOORD" set=“123” />
     <bind_vertex_input semantic=“otherUVs“
     ….
```

Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session

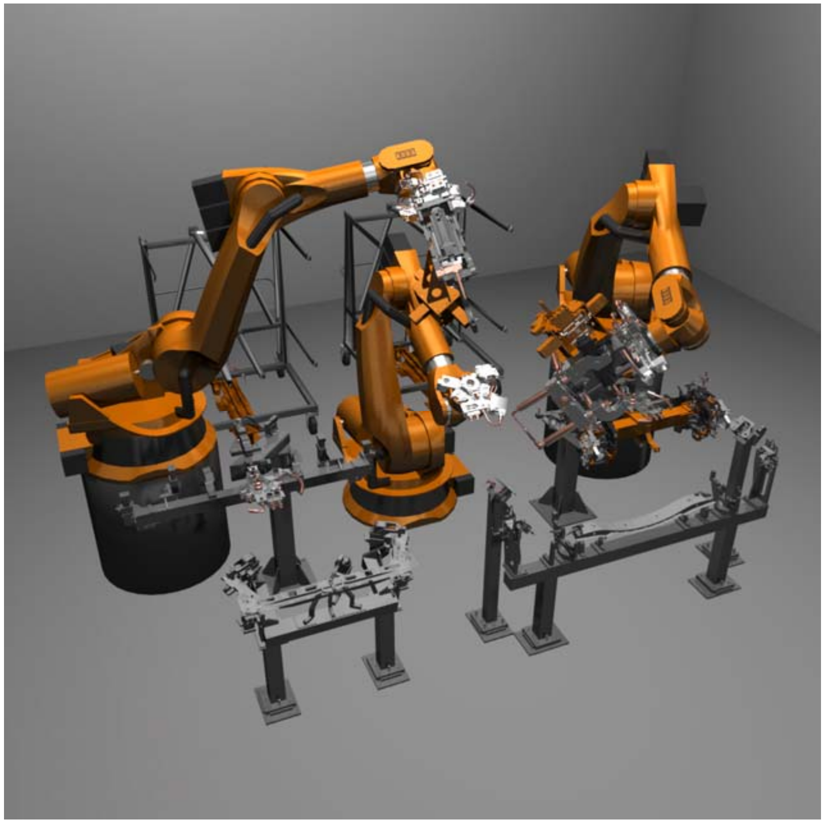
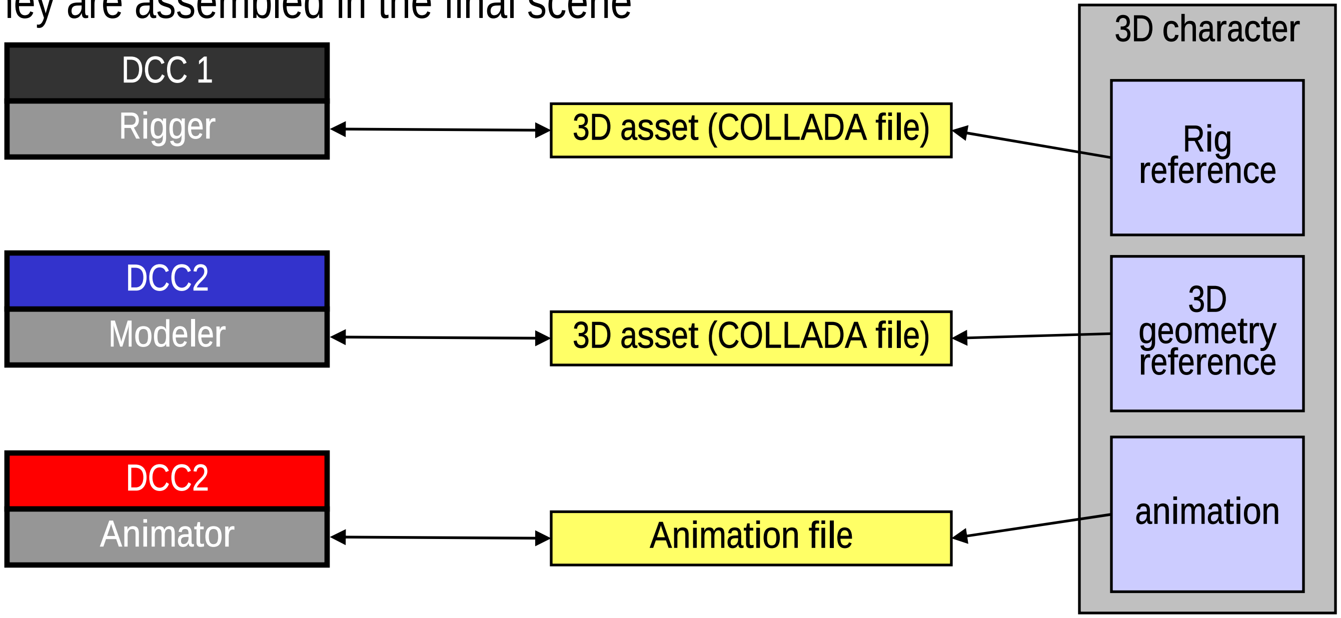


Image courtesy of DaimlerChrysler

COLLADA in workgroups and collaborative workspaces

- Taking advantage of the URI technology
- Assembly of scene involving asset created by multiple parties
  - Use reference models to externally reference 3D assets
- Multi task on 3D assets (i.e. rigging, modeling, animating)
  - Rig, geometry and animation are stored in different COLLADA files
  - They are assembled in the final scene



No more files !!!

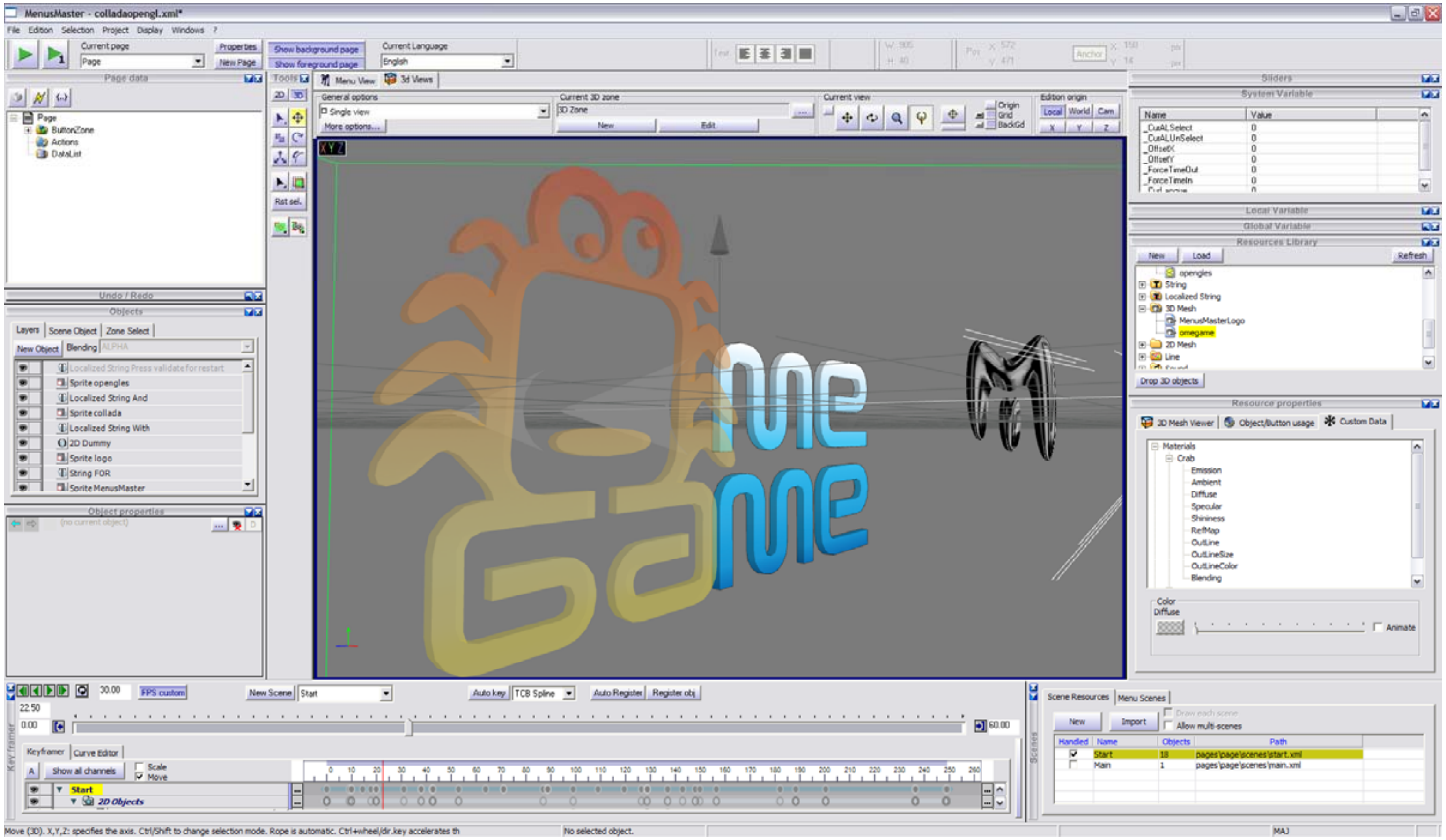
- Artist feedback loop is a major issue for embedded content development
  - Modify asset in tool, then export (wait), then load asset on target (wait), and then look at result with a different user interface
- Asset management based on opaque files
  - Can’t do a diff between two assets !
  - Cannot do a merge
  - All the software productivity tools are missing from content development
- Using an interactive database system, instead of a file system
  - Send updates in real time to the database
  - Listen to database changes – not all applications can handle this.
  - No more import/export
  - No problem if the tool crashes
  - Collaborative development
  - Simultaneous display of model on the target platform
- Collaboration COLLADA / Verse
  - Demonstrated at GDC (PS3/Windows/Linux/Mac)
  - Using COLLADA to load data in database and save data from database when work is done
  - Modified COLLADA DOM to listen to Verse database and real-time display on PS3

COLLADA Automation

- IK, Robot collision, Programming, Security, Human interaction
- Virtual factory, asset identification/management, cabling
- Presentations (rendering), training, simulation
- Interchange – Catia / Bentley / Kuka / ABB / Siemens / …

COLLADA for User Interface objects ?

Interested parties: OpenVG WG, Omegame, ...



Name	Value
_CurALSelect	0
_CurALUnSelect	0
_OffsetX	0
_OffsetY	0
_ForceTimeOut	0
_ForceTimeIn	0

Handled	Name	Objects	Path
☑	Start	18	pages\page\scenes\start.xml
□	Main	1	pages\page\scenes\main..xml

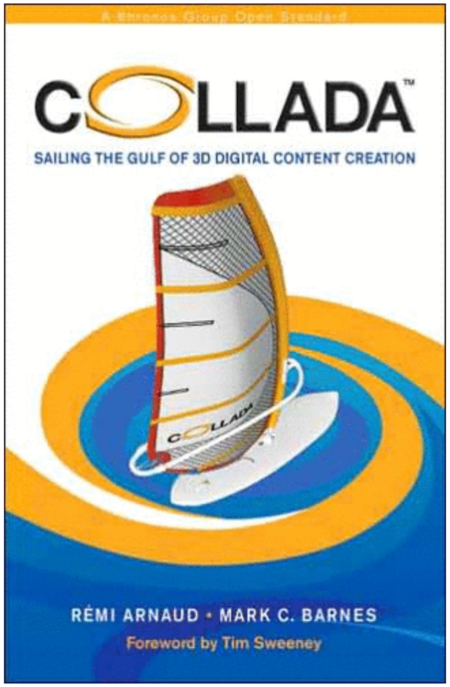
Move (3D). X,Y,Z: specifies the axis. Ctrl/Shift to change selection mode. Rope is automatic. Ctrl+wheel/dir.key accelerates th

No selected object.

Image courtesy of Omegame (www.omegame.com)

Agenda

- Goals and accomplishments
- COLLADA business model
- COLLADA and X3D
- COLLADA FX & Physics
- COLLADA XML design
- COLLADA Roadmap
- Q&A session


A Khronos Group Open Standard
COLLADA
SAILING THE GULF OF 3D DIGITAL CONTENT CREATION
RÉMI ARNAUD • MARK C. BARNES
Foreword by Tim Sweeney

www.collada.org
www.khronos.org/collada
collada@collada.org