COLLADA – Digital Asset Schema Release 1.5.0

Release Notes

June 2008

Editors: Mark Barnes and Ellen Levy Finch, Sony Computer Entertainment Inc.

Khronos Group makes no, and expressly disclaims any, representations or warranties, express or implied, regarding this specification, including, without limitation, any implied warranties of merchantability or fitness for a particular purpose or noninfringement of any intellectual property. Khronos Group makes no, and expressly disclaims any, warranties, express or implied, regarding the correctness, accuracy, completeness, timeliness, and reliability of the specification. Under no circumstances will the Khronos Group, or any of its Promoters, Contributors, or Members or their respective partners, officers, directors, employees, agents, or representatives be liable for any damages, whether direct, indirect, special, or consequential damages for lost revenues, lost profits, or otherwise, arising from or in connection with these materials.

Publication date: June 2008

Khronos Group
P.O. Box 1019
Clearlake Park, CA 95424, U.S.A.

Sony Computer Entertainment Inc.
2-6-21 Minami-Aoyama, Minato-ku,
Tokyo 107-0062 Japan

Sony Computer Entertainment America
919 E. Hillsdale Blvd.
Foster City, CA 94404, U.S.A.

Sony Computer Entertainment Europe
30 Golden Square
London W1F 9LD, U.K.

Table of Contents

Release Notes for COLLADA 1.5.0 Schema and Specification

About This Document

These *Release Notes* provide an overview of changes for the 1.5.0 COLLADA Digital Asset schema release. The 1.5.0 version of the schema and the *COLLADA – Digital Asset Schema Release 1.5.0 – Specification* are available for download from:

http://www.khronos.org/collada/

Additional Web Resources

Additional information about COLLADA is available at the following web locations:

- http://collada.org: Additional technical information about COLLADA; directories of publicly available COLLADA extensions, plug-ins, and conditioners; and a public forum for COLLADA discussions.
- http://www.khronos.org/bugzilla/: Public COLLADA bug-reporting system.

Notation in This Document

Bugzilla reports from the public COLLADA bug-tracking system are referenced in this document with the prefix “P”; reports from the Khronos members-only Bugzilla site are prefixed with “K”.

COLLADA General Changes

All importers and exporters of COLLADA documents must now support the .zae archive format

Resolves report K-2975.

This is not a schema change, but is a change in requirements for applications that import and export COLLADA documents. For details, refer to the “Archive Packaging” section in the *Specification*, Chapter 2, “Tool Requirements and Options.”

Schema Changes Since Version 1.4.1

These Release Notes identify all user-visible changes to the schema but do not provide details about the features or implementation. For details about all COLLADA features, refer to the updated *COLLADA – Digital Asset Schema Release 1.5.0 – Specification.* For details about implementation, refer to the COLLADA 1.5.0 schema.

COLLADA 1.4.1 versus 1.5.0

Several changes in this schema release are not compatible with existing COLLADA 1.4.1 documents. However, migrating from 1.4.1 to 1.5.0 is not required and 1.4.1 remains current and valid. If you don't need the features of 1.5.0, then continue to use the 1.4.1 schema. Please read these *Release Notes* carefully to decide whether you need the 1.5.0 features.

(General) Imported namespace (for xml:base element) now uses updateable schema location

Resolves report K-1885.

The COLLADA **schemaLocation** for the xs namespace now points to the continually revised XML schema:

```
<xs:import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="http://www.w3.org/2001/xml.xsd"/>
```

(New) COLLADA now supports kinematics

Resolves report K-2964 (Spec: K-2965).

Kinematics enables content creators to attach kinematical properties to objects in a visual scene.

Nodes in a visual scene can be controlled by a kinematical simulation. This is done by one or more kinematics models. A kinematics model consists of joints and links. A kinematics model can be controlled by one or more articulated systems. An articulated system enhances a kinematics model with kinematical or dynamical properties.

The top-level elements for kinematics in COLLADA are several libraries, which are children of **<COLLADA>** (see **<COLLADA>** note), and **<instance_kinematics_scene>**, which is a child of **<scene>**. Elements that support kinematics are the following; refer to the *Specification* for details.

- Joints

<joint>	Defines a single joint with one or more degree of freedom.
<library_joints>	Declares a module of **<joint>** elements.
<prismatic>	Defines a single translational degree of freedom of a joint.
<revolute>	Defines a single rotational degree of freedom of a joint.

- Kinematics Models

<attachment_end>	Defines one end of the closed loop in an attachment.
<attachment_full>	Connects two links.
<attachment_start>	Connects two links and defines one end of a closed loop.
<instance_joint>	Instantiates a COLLADA joint resource.
<instance_kinematics_model>	Instantiates a COLLADA **<kinematics_model>** resource.
<kinematics_model>	Describes a kinematics model.

<library_kinematics_models>	Provides a library in which to place **<kinematics_model>** elements.
<link>	Represents a rigid kinematical object without mass whose motion is constrained by one or more joints.

- Articulated Systems

<articulated_system>	Categorizes the declaration of generic control information for kinematics systems.
<axis_info>	Contains axis information to describe the kinematics or motion behavior of an articulated model.
<bind> (kinematics)	Binds inputs to kinematics parameters upon instantiation.
<connect_param>	Creates a symbolic connection between two previously defined parameters.
<effector_info>	Specifies additional dynamics information for an effector.
<frame_object>	Contains information for a frame used for kinematics calculation.
<frame_origin>	Contains information for a frame used for kinematics calculation.
<frame_tcp>	Contains information for a frame used for kinematics calculation.
<frame_tip>	Contains information for a frame used for kinematics calculation.
<instance_articulated_system>	Instantiates a COLLADA **<articulated_system>** resource.
<kinematics>	Contains additional information to describe the kinematical behavior of an articulated model.
<library_articulated_systems>	Provides a library in which to place **<articulated_system>** elements.
<motion>	Contains additional information to describe the dynamics behavior of an articulated model.

- Kinematics Scenes: The kinematics scene is the instantiated model for a concrete scene. It defines the links that are used and the current configuration of the scene, especially default or current joint values:

<bind_joint_axis>	Binds a joint axis of a kinematics model to a single transformation of a node.
<bind_kinematics_model>	Binds a kinematics model to a node.
<instance_kinematics_scene>	Instantiates a COLLADA **<kinematics_scene>** resource.
<kinematics_scene>	Embodies the entire set of information that can be articulated from the contents of a COLLADA resource.
<library_kinematics_scenes>	Provides a library in which to place **<kinematics_scene>** elements.

(New) COLLADA now supports boundary representations (B-reps)

Resolves report K-2960 (Spec K-2961).

Boundary representation (B-rep) models are composed of two parts: topology and geometry. The topology specifies different entities (vertices, edges, and so on) that limit the corresponding unbounded geometry. Geometric entities include lines, curves, and surfaces.

The top-level element is **<brep>**, which is a child of **<geometry>**. Elements that support B-rep are the following; refer to the *Specification* for details.

- Geometry

brep	Describes a boundary representation (B-rep) structure.

- Curves

circle	Describes a circle in 3D space.
curve	Describes a specific curve.
curves	Contains all curves that are used in the B-rep structure.
ellipse	Describes an ellipse in 3D space.
hyperbola	Describes a hyperbola in 3D space.
line	Describes a single line in 3D space.
nurbs	Describes a NURBS curve in 3D space.
parabola	Describes a parabola in 3D space.
surface_curves	Contains all parametric curves (pcurves) that are used in the B-rep structure.

- Topology

edges	Describes the edges of a B-rep structure.
faces	Describes the faces of a B-rep structure.
pcurves	Specifies how an edge is represented in a face's parametric space.
shells	Describes the shells of a B-rep structure.
solids	Describes the solids of a B-rep structure.
wires	Describes the wires of a B-rep structure.

- Surfaces

cone	Describes a conical surface.
cylinder	Describes an unlimited cylindrical surface.
nurbs_surface	Describes a NURBS surface in 3D space.
plane (in Physics)	Describes an infinite planar surface.
sphere (in Physics)	Describes a centered sphere primitive.
surface	Describes a specific surface.
surfaces	Contains all surfaces that are used in the B-rep structure.
swept_surface	Describes a surface by extruding or revolving a curve.
torus	Describes a torus in 3D space.

- Transformation

orient	Describes the orientation of the object frame.
origin	Describes the origin of the object frame.

(all) All name arguments are now of type `xs:token`

Resolves reports P-9, K-532 (Spec: P-64).

The name argument on all elements where it occurs is now of type **xs:token**. This allows the argument values to contain a single space between words.

(all) Types for SIDs and SID references now exist

Resolves report K-1293 (Spec: K-1886, K-3154).

Two new types have been created:

- **sid_type**: A scoped identifier. All **sid** attributes are now of type **sid_type**.
- **sidref_type**: A reference to a scoped identifier. All elements and attributes that refer to another element's SID now are of type **sidref_type** (in the following list, indicated by *xxx*):

- **<channel target=** *“xxx”***>**
- **<***renderstate_name* **param=***“xxx”***>**
- **<bind target=***“xxx”***>**
- **<instance_rigid_body body=***“xxx”*>
- **<instance_rigid_constraint constraint=***“xxx”*>
- **<param ref=***“xxx”*>
- **<texenv sampler=***“xxx”*>
- **<SIDREF>***xxx***</SIDREF>**
- **<texture*><param>***xxx***</param></texture*>**

See also the **<SIDREF_array>** note for a new related array element.

Refer to “Address Syntax” in Chapter 3: “Schema Concepts” in the *Specification* for information about valid SIDs and SIDREFs.

(FX) <argument>’s unit attribute changed name and type

Resolved: No report filed.

The **unit** attribute of **<argument>** has been renamed to **sampler** and is now of type **sidref_type**, which points to a sampler.

(FX) <array> now has resizable attribute

Resolved: No report filed.

In CG scope, the **<array>** element now has the **resizable** attribute, which is a Boolean that allows the array to change size.

(Core) <asset> now has <extra> child

Resolves report K-617 (Spec: K-3272).

The **<asset>** element now allows zero or more **<extra>** child elements.

(Core) <asset> now has geographic coverage information

Resolves report K-2955 (Spec: K-2863).

The **<coverage>/<geographic_location>** elements are now defined in **<asset>**. Refer to the **<asset>** and **<geographic_location>** elements in the “Core Elements Reference” chapter of the *Specification* for details.
For example:

```
<asset>
    <coverage>
      <geographic_location>
        <longitude>-105.2830</longitude>
        <latitude>40.0170</latitude>
        <altitude mode="relativeToGround">0</altitude>
      </geographic_location>
    </coverage>
    <created>2008-01-28T20:51:36Z<created/>
    <modified>2008-01-28T20:51:36Z<modified/>
</asset>
```

(Physics) <capsule>'s <radius> child element now has three values

Resolves report K-3168 (Spec: K-3169).

<capsule>/<radius> now contains three floating-point values, not two.

(Core) <COLLADA>'s `version` attribute supports 1.5.0

Resolved: No report filed.

The **<COLLADA>** element's **version** attribute supports only the value "1.5.0" for this version of the schema. All 1.5.0 COLLADA documents should refer to the current COLLADA schema as follows:

```
<COLLADA xmlns="http://www.collada.org/2008/03/COLLADASchema" version="1.5.0">
```

(Core) <COLLADA> includes additional libraries

Resolved: Integration of new features.

The **<COLLADA>** element now recognizes the following additional children:

- **<library_articulated_systems>** (for Kinematics)
- **<library_formulas>**
- **<library_joints>** (for Kinematics)
- **<library_kinematics_models>** (for Kinematics)
- **<library_kinematics_scenes>** (for Kinematics)

(FX) <color_target>, <depth_target>, and <stencil_target> now directly reference an image

Resolved: No report filed.

<color_target>, **<depth_target>**, and **<stencil_target>** now require exactly one of these child elements:

<param> (reference)	References a sampler parameter to determine which image to use.
<instance_image>	Directly instantiates a renderable image.

(FX) <connect_param> has been removed.

Resolves report K-3156 (Spec: K-459).

<connect_param> is no longer valid in **<array>**, **<usertype>**, or most cases of **<setparam>**. It is now valid only in Kinematics **<setparam>** elements and **<instance_formula>/<setparam>**.

(Core) <contributor> now has email and website children

Resolves report K-370 (Spec: K-3117).

The **<contributor>** element now has **<author_email>** and **<author_website>** child elements. For example:

```
<asset>
  <contributor>
    <author>Bob the Artist</author>
    <author_email>bob@bobartist.com</author_email>
    <author_website>http://www.bobartist.com</author_website>
    <authoring_tool>Super3DmodelMaker3000</authoring_tool>
    <comments>This is a big Tank</comments>
    <copyright>Bob's game shack: all rights reserved</copyright>
    <source_data>c:\models\tank.s3d</source_data>
  </contributor>
</asset>
```

(FX) <create_2d>, <create_2d>, and <create_cube> have been added to allow the customized initialization of textures for an image

Resolved: No report filed.

These elements are children of **<image>**. Their basic syntax is:

```
<create_2d>
    <size_exact width="..." height="..."/> or
        <size_ratio width="..." height="...">
    <mips levels="..." auto_generate="true"/> or <unnormalized />
    <format ... />
    <init_from .../>
</create_2d>

<create_3d>
   <size width="..." height="..." depth="..."/>
   <mips levels="..." auto_generate="true"/>
   <array length="..." .../>
   <format ... />
   <init_from .../>
</create_3d>

<create_cube>
   <size width="..."/>
   <mips levels="..." auto_generate="false"/>
   <array length="..." .../>
   <format ... />
   <init_from... />
</create_cube>
```

For more information, see the **<image>**, **<format>**, and **<init_from>** notes and refer to the *Specification*.

(Physics) <dynamic> in <instance_rigid_body> and <rigid_body> now has a default value

Resolves report P-51 (Spec: P-65).

<dynamic> in **<instance_rigid_body/<technique_common>** and **<rigid_body/<technique_common>** now has a default value of **true**.

(Core) <evaluate_scene> and <render> have been redesigned to enhance extensibility

Resolves report K-366.

Changes to **<evaluate_scene>**:

- Three new optional attributes – **id**, **sid**, and **enable**.
- Two new children – **<asset>** and **<extra>**.
- **<render>** child element is now optional (can appear 0 or more times). (See also **<render>** note.)

Changes to **<render>**:

- Two new optional attributes – **name** and **sid**.
- New child element – **<extra>**.
- **<instance_effect>** child has been renamed **<instance_material>** and has been redefined (Note: this is *not* the same as the **<bind_material>/<technique_common>/<instance_material>**.)

Changes to **<instance_material>** (compared to **<instance_effect>**):

- Removed attributes **name** and **sid**, as only one **<instance_material>** is allowed.
- Replaced child element **<technique_hint>** with new element **<technique_override>**, which allows you to target specific techniques and passes inside a material rather than having to split the effects techniques and passes into multiple effects.
- Replaced child element **<setparam>** with **<bind>**.

The basic syntax is:

```
<evaluate_scene name="..." id="..." sid="..." enable="...">
   <asset ... />
   <render camera_node="..." name="..." sid="..." >
      <layer .../>
      <instance_material url="...">
         <technique_override ref="..." pass="..." />
         <bind .../>
         <extra  .../>
      </instance_material>
      <extra ...>
   </render>
   <extra ... />
</evaluate_scene>
```

(Core) <float_array> now has stronger type constraints

Resolves report P-32 (Spec: P-37).

The **digits** and **magnitude** attributes now have restricted value ranges; their definitions are:

digits	**xs:unsignedByte**	The number of significant decimal digits of the floating-point values that can be contained in the array. The minimum value is 1; the maximum is 17. The default is 6. Optional.
magnitude	**xs:short**	The largest exponent of the floating-point values that can be contained in the array. The maximum value is 308; the minimum is -324. The default is 38. Optional.

(FX) <format> added to describe <image> assets

Resolved: No report filed.

This new element describes the formatting or memory layout expected of an **<image>** asset. Within **<image>**, it is a child of **<create_2d>**, **<create_3d>**, and **<create_cube>**. **<format>**'s children can be either or both of the following:

```
<hint channels="..." range="..." precision="..." space="..." />
   or
<exact>string-specifying-texel-format</exact>
```

Refer to the *Specification* for details.

(Core) <formula>, <instance_formula>, and <library_formulas> added to support MathML

Resolves report K-2962 (Spec: K-2963).

<formula> can appear in **<library_formulas>**, **<animation_clip>**, and the new kinematics elements **<kinematics_model>**/**<technique_common>** and **<kinematics>**/**<axis_info>**. **<formula>** provides the capability of defining functions in COLLADA. In a COLLADA document with access to the MathML schema, **<formula>/<technique_common>** can contain any valid MathML XML. Refer to the *Specification* for details.

(FX) <generator> has been removed

Resolves reports K-464, K-636, K-3157.

(FX) <image> has been redesigned and <instance_image> has been added

Resolved: No report filed.

The **<image>** element has the following changes. Refer to the *Specification* for details:

- Removed attributes **format**, **height**, **width**, and **depth**.
- Added attribute **sid**:
  ```
  <image id="..." name="..." sid="...">
  ```
- Replaced the old choices for source information (**<data>** and **<init_from>**) with:
  <init_from> or <create_2d> or <create_3d> or <create_cube>
  For more information, see the notes for **<init_from>** and **<create_2d>**, **<create_3d>**, **<create_cube>**.
- A new **<renderable>** child element specifies whether the image is a render target. It has an optional Boolean attribute, **share**, that indicates whether to share, rather than clone, the render target among all instances:
  ```
    <renderable share="..." />
  ```
- The **<image>** element is now valid only as a child of **<library_images>**; all other locations are invalid.
- The new **<instance_image>** element is valid as a child of **<sampler_*>**, **<color_target>**, **<depth_target>**, and **<stencil_target>**.

The basic syntax is now:

```
<image id="..." name="..." sid="...">
   <asset .../>
   <renderable share="..." />
   <init_from.../>  or <create_2d.../> or <create_3d.../> or <create_cube.../>
   <extra .../>
</image>
```

(FX) <init_from> has been redesigned to provide better initialization control over images

Resolved: No report filed.

<init_from> can now be a child of **<image>**, **<create_2d>**, **<create_3d>**, or **<create_cube>**. The specific attributes and children allowed vary by parent, but the basic syntax is:

```
<image>
   <init_from mips_generate="...">
```

```
      <ref> url </ref>
         or
      <hex format="...">embedded_image_data</hex>
   </init_from>
</image>

or

<create_*>
   <init_from array_index="..." mip_index="..." depth="..." face="...">
      <ref> url </ref>
         or
      <hex format="...">embedded_image_data</hex>
   </init_from>
</create_*>
```

For details, see the **<image>** and **<create_*>** notes and refer to the *Specification*.

(Core) <instance_node> now has proxy attribute to support multirepresentation and level of detail (LOD)

Resolves report K-3166 (Spec: K-2974).

The **proxy** attribute is an optional **xs:anyURI**. The application can decide to resolve either the URL in the **url** attribute or the URL in the **proxy** attribute. Both resolve into a **<node>** element. More information about how applications can decide which path to follow should be stored in the **<extra>** element of **<instance_node>**. See the *Specification* for details.

(FX) <pass> has been redesigned with new child elements <states> and <evaluate>

Resolved: No report filed.

The new basic syntax for **<pass>** is the following:

```
<pass sid="..." >
   <annotate   />
   <states>
      render_states_list
   </states>
   <program .../>  <!-- in CG, GLES2, and GLSL scope only -->
   <evaluate>
      <color_target .../>
      <depth_target .../>
      <stencil_target .../>
      <color_clear .../>
      <depth_clear .../>
      <stencil_clear .../>
      <draw .../>
   <extra .../>
</pass>
```

The differences are:

- Most child elements that had been directly below **<pass>** have been moved below new child elements **<states>** and **<evaluate>**.
- The old **<shader>** child element has been replaced by the **<program>** element. See the **<program>** note.

(FX) <profile_*> and <profile_*>/<technique> children have been removed

Resolved: No report filed.

The following children no longer appear directly under the **`<profile_*>`** elements in most cases:

- **`<image>`**, **`<newparam>`**, **`<setparam>`**, **`<code>`**, **`<include>`**

The following children no longer appear directly under the **`<profile_*>`**/**`<technique>`** elements:

- **`<array>`**, **`cg_value_type`**, **`<connect_param>`**, **`<usertype>`**

(FX) <profile_BRIDGE> added

Resolved: No report filed.

This profile provides support for referencing effect profiles written with external standards that are not covered by other **`<profile_*>`** elements. Valid under **`<effect>`**. The basic syntax is:

```
<profile_BRIDGE id="..." platform="..." url="...">
   <asset .../>
   <extra .../>
</profile_BRIDGE>
```

(FX) <profile_GLES2> added

Resolved: No report filed.

Valid under **`<effect>`**. The basic syntax is:

```
<profile_GLES2 id="..." language="..." platforms="...">
   <asset .../>
   <code  .../>
   <include .../>
   <newparam .../>
   <technique>
      <asset .../>
      <annotate .../>
      <pass .../>
      <extra .../>
   </technique>
   <extra .../>
</profile_GLES2>
```

(FX) <profile_*> elements now all have `id` attribute

Resolves reports K-322, P-28 (Spec: K-329).

(FX) <profile_GLSL> now has `platform` attribute

Resolves report K-322 (Spec: K-329).

This now matches the functionality of other existing **`<profile_*>`** elements.

(FX) <program> added to better define shaders with new children including <compiler>, <linker>, <binary>, <bind_attribute>

Resolved: No report filed.

The new **<program>** element, a child of **<pass>**, provides facilities for linking multiple shaders to produce a pipeline for geometry processing. The valid elements are different in different profiles, but the complete list of possible elements is:

```
<program>
   <shader stage="...">
      <sources  .../>
      <compiler platform="..." target="..." options="...">
         <binary .../>
      </compiler>
      <bind_uniform .../>
      <extra .../>
   </shader>
   <linker platform="..." target="..." options="...">
      <binary .../>
   </linker>
   <bind_attribute symbol="...">
      <semantic .../>
   </bind_attribute>
   <bind_uniform .../>
</program>
```

The **<bind_attribute>** binds semantics to the vertex attribute inputs of a shader. For details, see the *Specification*.

See also the note for **<shader>**.

(FX) <render> has been updated

Resolved: No report filed.

Changes to the **<render>** element:

- Now has **name** and **sid** attributes.
- Now allows the **<extra>** child element.
- **<instance_effect>** is no longer a valid child element

(Physics) <rigid_body> now has id attribute

Resolves report K-1820 (Spec: K-1821).

(Core) <sampler> now has attributes to specify the sampled value before the first key and after the last key

Resolves report K-2973 (Spec: K-3171).

<sampler> now has attributes **pre_behavior** and **post_behavior**, which indicate what the sampled value should be before the first key and after the last key. Refer to the *Specification* for details. Example:

```
<sampler pre_behavior="CONSTANT" post_behavior="GRADIENT">
```

(FX) <sampler*> child elements have been redesigned

Resolved: No report filed.

`<sampler1D>`, **`<sampler2D>`**, **`<sampler3D>`**, **`<samplerCUBE>`**, **`<samplerDEPTH>`**, and **`<samplerRECT>`** child elements have the following changes:

- **`<instance_image>`** added, which instantiates a default image from which the sampler is to consume.
- **`<mipmap_maxlevel>`** renamed to **`<mip_max_level>`**.
- **`<mipmap_bias>`** renamed to **`<mip_bias>`**.
- **`<mip_min_level>`** added, which is the minimum progressive levels to begin to evaluate.
- **`<max_anisotropy>`** added, which is the number of samples that can be used during anisotropic filtering.
- Most child elements are now valid in all **`<sampler*>`** elements – although some might not be relevant to a particular element.
- The **`<wrap_*>`** mode NONE has been removed; the mode MIRROR_ONCE has been added, which takes the absolute value of the texture coordinate and then clamps to the maximum value.
- Removed **`<*filter>`** (**`<minfiler>`**, **`<magfilter>`**, **`<mipfilter>`**) values NEAREST_MIPMAP_NEAREST, LINEAR_MIPMAP_NEAREST, NEAREST_MIPMAP_LINEAR, LINEAR_MIPMAP_LINEAR.

  Added **`<*filter>`** value ANISOTROPIC (for **`<minfilter>`** only), which compensates for distortion caused by the difference in angle between the polygon and the plane of the screen.

  The **`<*filter>`** value NONE is now valid only for **`<mipfilter>`**.

(FX) <sampler_image> added

Resolved: No report filed.

Instantiates an image targeted for samplers. This is not a sampler type but is, instead, an element used to modify an existing sampler. The sampler **`<newparam>`** identified by the parent **`<setparam>`** receives the instantiated image. This type, derived from **`<instance_image>`**, has no specific extension but was renamed for clarity in this situation.

(FX) <sampler_state> has been redesigned as <sampler_states>

Resolved: No report filed.

The changes are:

- **`<sampler_state>`** has been removed from **`<newparam>`**.
- **`<sampler_state>`** is now **`<sampler_states>`** under **`<setparam>`**. It allows all the same child elements (except **`<instance_image>`**) as the redesigned **`<sampler*>`** elements; see the **`<sampler*>`** note.

(FX) <setparam> removed from several elements

Resolved: No report filed.

`<setparam>` is no longer valid under **`<effect>`**, any of the **`<profile_*>`** elements, or **`<profile_*>/<technique>`**.

(FX) <shader> has been redesigned

Resolved: No report filed.

The **<shader>** element has the following changes. Refer to the *Specification* for details:

- Valid values for the **stage** attribute are now FRAGMENT, GEOMETRY, TESSELATION, and VERTEX. All are valid in all contexts where **<shader>** appears. FRAGMENTPROGRAM and VERTEXPROGRAM have been removed.
- The **<annotate>** child element has been deleted.
- The **<compiler_target>** and **<compiler_options>** child elements have been deleted and replaced with **<sources>** and **<compiler>**. See notes for **<sources>** and **<program>**.
- The **<name>** child element has been deleted; instead, use the **entry** attribute in the **<sources>** element.
- The **<bind>** child element has been renamed **<bind_uniform>**.
- **<extra>** is now a valid child element.

The basic syntax is now:

```
<shader stage="VERTEX">
   <sources entry="main">
      <inline .../>
      <import ref="..."/>
   </sources>
   <compiler platform="..." target = "..." />
   <bind_uniform symbol="...">
      <param ref="..."/>
   </bind_uniform>
   <extra ... />
</shader>
```

(Core) <SIDREF_array> added

Resolves report K-3282 (Spec: K-3283).

The **<SIDREF_array>** element now exists as a child of **<source>**. It contains one or more values of type **sidref_type** and has the standard array attributes of **id**, **name**, and **count**. For example:

```
<SIDREF_array id="SIDREFlist" name="My SIDREF list" count="3">
    item1  item2  item3
</SIDREF_array>
```

(Core, Physics) <source> now allows arrays of tokens and SIDREFs

Resolves report K-3282 (Spec: K-3283).

<source> now allows child elements **<SIDREF_array>** and **<token_array>**. See notes for **<SIDREF_array>** and **<token_array>**.

(FX) <sources> has been added to concatenate shader source code

Resolved: No report filed.

<shader>/<sources> allows you to specify multiple sources for shader code, combining inline code (**<inline>**) with included files (**<import>**, which references **<code>** or **<include>** elements). **<sources>** has the following basic syntax:

```
<sources entry="main">
   <inline .../>
   <import ref="..."/>
</sources>
```

See also the note for **<shader>**.

(FX) <surface> has been removed

Resolves report K-338.

(Core) <tapered_capsule> and <tapered_cylinder> have been removed

Resolves report K-3170 (Spec: K-3202).

(Core) <technique> now avoids conflicts with COLLADA element names

Resolves report K-213.

<technique> child elements can now safely use any element names other than **<COLLADA>** without the risk that they conflict with COLLADA element names during validation.

(FX) <texenv>'s `unit` attribute changed name and type

Resolved: No report filed.

The **unit** attribute of **<texenv>** has been renamed to **sampler** and is now of type **sidref_type**, which points to a sampler.

(FX) <texture>'s <extra> now uses maxOccurs=unbounded

Resolves report K-448 (Spec: K-2890).

<texture>/<extra> now allows one or more occurrences.

(FX) <texture_pipeline> now occurs only in <states>

Resolved: No report filed.

<texture_pipeline> can no longer be created as a **<newparam>**; it must be used only as a render state. This means that it is no longer valid under **<pass>**, **<newparam>**, or **<setparam>**.

(FX) <texture_unit> has been removed

Resolved: No report filed.

<texture_unit> no longer exists. Its three primary child elements:

- **<surface>** no longer exists; see **<surface>** note.
- **<texcoord>** now is valid only in GLES scope under **<newparam>/<sampler2D>**.
- **<sampler_state>** was a reference to a parameter.

(Core) <token_array> added

Resolves report K-3282.

The **<token_array>** element now exists as a child of **<source>**. It contains one or more values of type **xs:token** and has the standard array attributes of **id**, **name**, and **count**. For example:

```
<token_array id="tokenlist" name="My token list" count="3">
    item1  item2  item3
</token_array>
```

(FX) <transparent> supports additional opaque modes

Resolves report K-622 (Spec: K-3118).

The **<transparent>** element's **opaque** attribute now allows, in addition to **A_ONE** and **RGB_ZERO**, the following values:

- **A_ZERO** (the default): Takes the transparency information from the color's alpha channel, where the value 0.0 is opaque.
- **RGB_ONE**: Takes the transparency information from the color's red, green, and blue channels, where the value 1.0 is opaque, with each channel modulated independently.

(FX) <usertype> has been redesigned

Resolved: No report filed.

The following changes have been made:

- The **name** attribute has been renamed to **typename**.
- The only valid child element is **<setparam>**. All other child elements have been removed.

Specification Changes Since 1.4.1 (2nd Edition)

(General) Made a variety of improvements to the text

No reports filed.

- Updated text to match 1.5.0 schema (see previous section).
- Fixed 1.4.1 *Specification* (2nd Edition) bugs and omissions (as detailed in this section).
- Clarified text throughout, in particular to **<controller>** in the "Core Elements Reference" chapter.
- Expanded the list of "Other Sources of Information" and corrected outdated links.
- Removed the "Development Methods" section from the chapter on COLLADA's history.
- In the "Schema Concepts" chapter, updated and expanded the "Address Syntax" section.
- In the "Schema Concepts" chapter, expanded the "Example and Discussion on Techniques" section.
- Added a "Parameters" discussion to the "Programming Guide" chapter.
- Updated the GLSL example appendix and added a GLES2 example to the appendix.
- In the "Core Elements Reference" chapter, relabeled "**<param>** (core)" to "**<param>** (data flow)" to better describe its functionality; relabeled "**<input>** (indexed)" to "**<input>** (shared)" and "**<input>** (unindexed)" to "**<input>** (unshared)."

- Rewrote the introduction to the “Physics Reference” chapter and clarified and expanded text in many of the Physics elements, in particular, <morph> and <physics_model>.
- Moved the <setparam>, <newparam>, and <param> (reference) elements to the “Core Elements Reference” chapter to reflect that they are now also used in some locations other than FX.

(Index) Elements without main entries added to the “Index of COLLADA Elements”

Resolves reports P-18, P-19.

All elements, whether or not they have full reference entries (for example, <h> and <v>), are now indexed.

(Physics) <convex_mesh> corrected the occurrences of its child elements

Resolves report K-3155.

Corrected the “Child Elements” section of <convex_mesh> to state that no child elements need to be specified, but if any is specified, then <source> can occur 1 or more times and <vertices> must occur exactly once.

(FX) <color_target> attribute type corrected

Resolved: No report filed.

The **face** attribute of <color_target> is now correctly documented as an enumeration.

(Core) <evaluate_scene> is now documented

Resolves report K-366.

<visual_scene>/<evaluate_scene> is now documented.

(Core) <morph> attribute name is source, not sid

Resolves report K-3352.

<morph source=... > is now correctly documented.

(Core) <orthographic> child elements corrected

Resolved: No report filed.

<extra>, which was incorrectly listed as a child element of <orthographic>, has been removed from the *Specification*.

(Core) <point> child elements corrected

Resolved: No report filed.

<zfar>, which was incorrectly listed as a child element of <point>, has been removed from the *Specification*.

Specification 1.5.0 Existing Issues

(About This Manual) Other Sources of Information correction

Open report: K-3500.

“Softimage dotXSI FTK” should be “Softimage Crosswalk SDK”.

(Core) <token_array> isn’t documented

Open report: K-3473.

See the **`<token_array>`** description in the preceding “Schema Changes Since Version 1.4.1” section.

(FX) type `fx_sampler_type` is incorrectly referred to as `fx_sampler_common`

Open report: K-3490.

This type is used in the *Specification* only as a convenience; it is not expected that any COLLADA documents or applications would use this type name explicitly.

(Core, FX) Reference to “Parameter-Type Elements” section should point to Chapter 11

Open report: K-3491.

Cross-reference to “Parameter-Type Elements” section should point to Chapter 11; currently it says "at the end of the chapter" in 3 places:

- **`<array>`**
- **`<bind_uniform>`**
- **`<setparam>`**