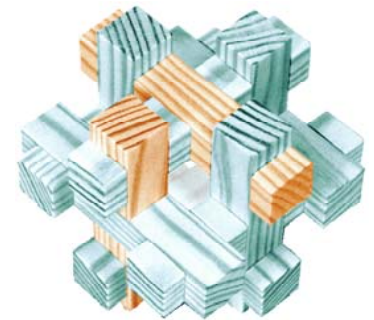


# CONTIGRA:

*An XML-Based Architecture  
for Component-Oriented  
3D Applications*



**Raimund Dachsel, Michael Hinz, Klaus Meißner**  
Dresden University of Technology, Multimedia Technology Group

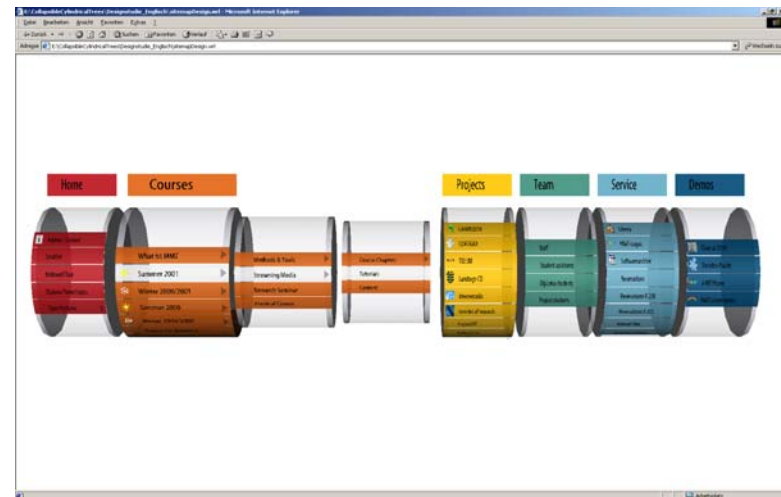
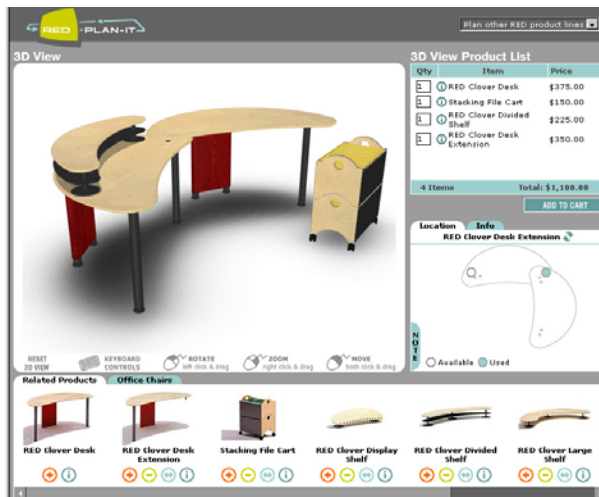
# Outline

- Motivation and Vision
- Related Work
- The CONTIGRA Architecture
  - Component Development Levels & Tasks
  - Three Levels & their Markup Languages
  - Example 3D Application
- Conclusion & Future Work

# Motivation

## ■ Current Situation

- Improvements in 3D graphics hardware & fast-evolving (3D) Internet technologies
- Increase of Web-based 3D applications
  - E-Commerce, Entertainment, Education, InfoVis...



# Motivation

## ☹ Problems

- Multitude of proprietary Web3d-formats | X3D
- Lack of 3D design standards, guidelines
- Authoring tools proprietary, not multi-disciplinary
- Too much programming, time-consuming | Reuse?

## 😊 Future Vision

- Repertoire of adaptable 3D Widgets, Metaphors  
→ Standards for 3D user interfaces & desktop VE
- Reuse of high-level 3D components

## Component

### WidgetComponent

SelectionComponent

...

NavigationComponent

...

ManipulationComponent

...

### ApplicationControlComponent

ButtonComponent

...

ValuatorComponent

...

TextInputComponent

WindowComponent

### MenuComponent

#### RingMenuComponent

...

ToolTipComponent

ColorChooserComponent

...

VisualizationComponent

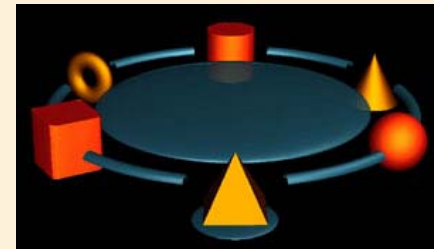
...

OrientationComponent

...

### AvatarComponent

### AnimationComponent



ary  
euse?

ors  
p VE

# Related Work

- Few 3D Component Approaches
  - NPSNET-V, Bamboo, i4D, 3D Beans, ...
  - Jamal (BML), 3dml, VRML Prototypes

→ *3D format dependency, code-centered, low-level*
- X3D - extensibility concepts
  - Prototypes, Profiles, Components
  - Basically Scene Graph Extensibility
  - No SG abstractions, no higher-level assembly



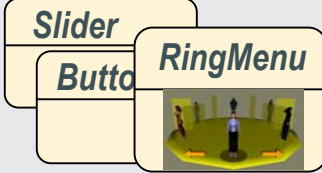


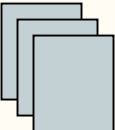
# CONTIGRA Architecture

*Component OriENted Three-dimensional  
Interactive GRaphical Applications*

## ■ Characteristics

- Document-centered 3D component architecture
- Documents describe component interfaces, implementation, configuration, and assembly
- Multi-layered, declarative approach (basis: XML)
- High-level view, hides scene graph details
- Abstraction to existing 3D toolkits, formats, APIs largely independent of implementation issues

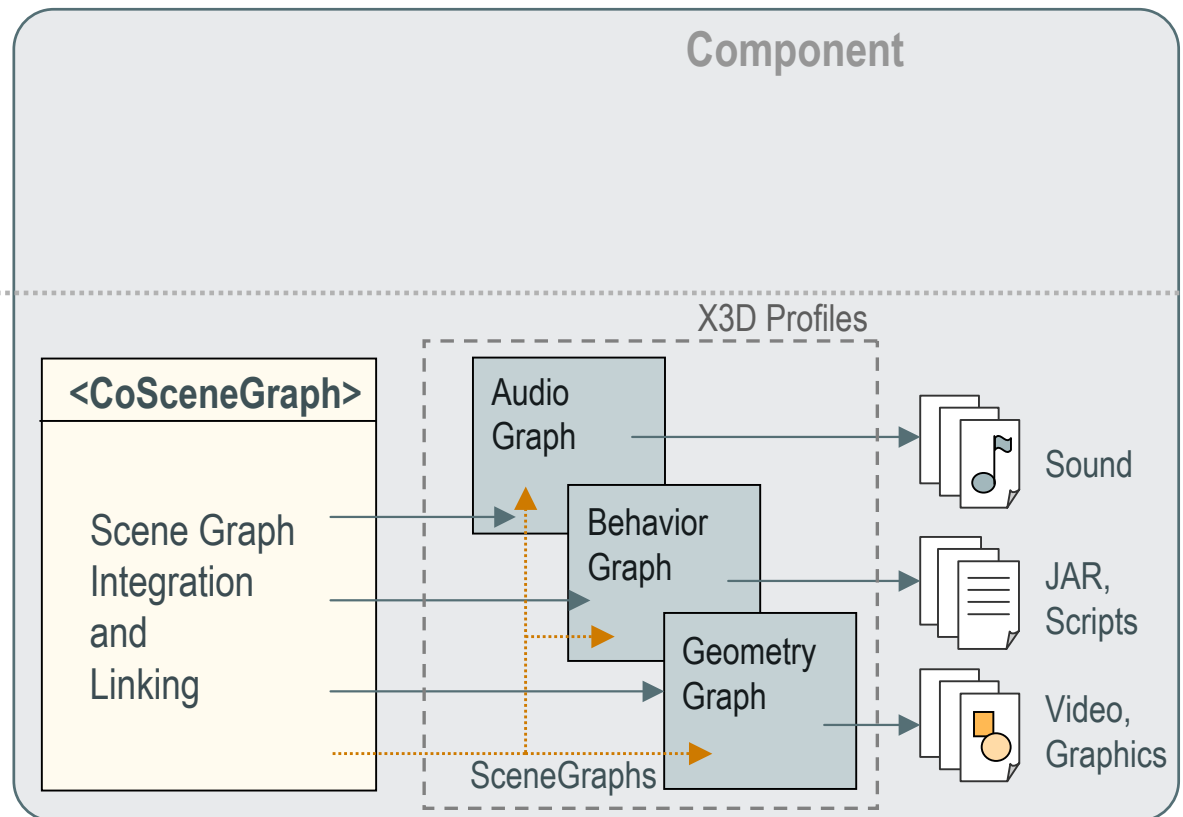
# Component Levels & Tasks

Level	Tasks	Result (Documents)	Tools
Runtime	<i>Usage Adaptation</i>	 <p>Executable 3D Application (Web / stand alone)</p>	<b>Specific 3D Viewer</b> (X3D Applet, VRML-, Viewpoint PlugIn ...)
Configuration & Assembly	<i>Configuration Assembly Linking</i>	 <p>Assembled 3D Application (format independent)</p>	<b>CONTIGRA SceneBuilder</b>
		 <p>Configured Components</p>	
Distribution	<i>Selection Retrieval</i>	Packaged 3D-Components	Component Database, Web Interface 
Development	<i>Description</i>	 <p>Component Interface</p>	<b>CONTIGRA ComponentBuilder</b>
	<i>Implementation</i>	 <p>Implementation Files (Media Standards)</p>	3D-Modelling, Media & Programming Tools



# CONTIGRA Markup Levels

Task	XML Schema	CONTIGRA Documents	Other Resources
------	------------	--------------------	-----------------



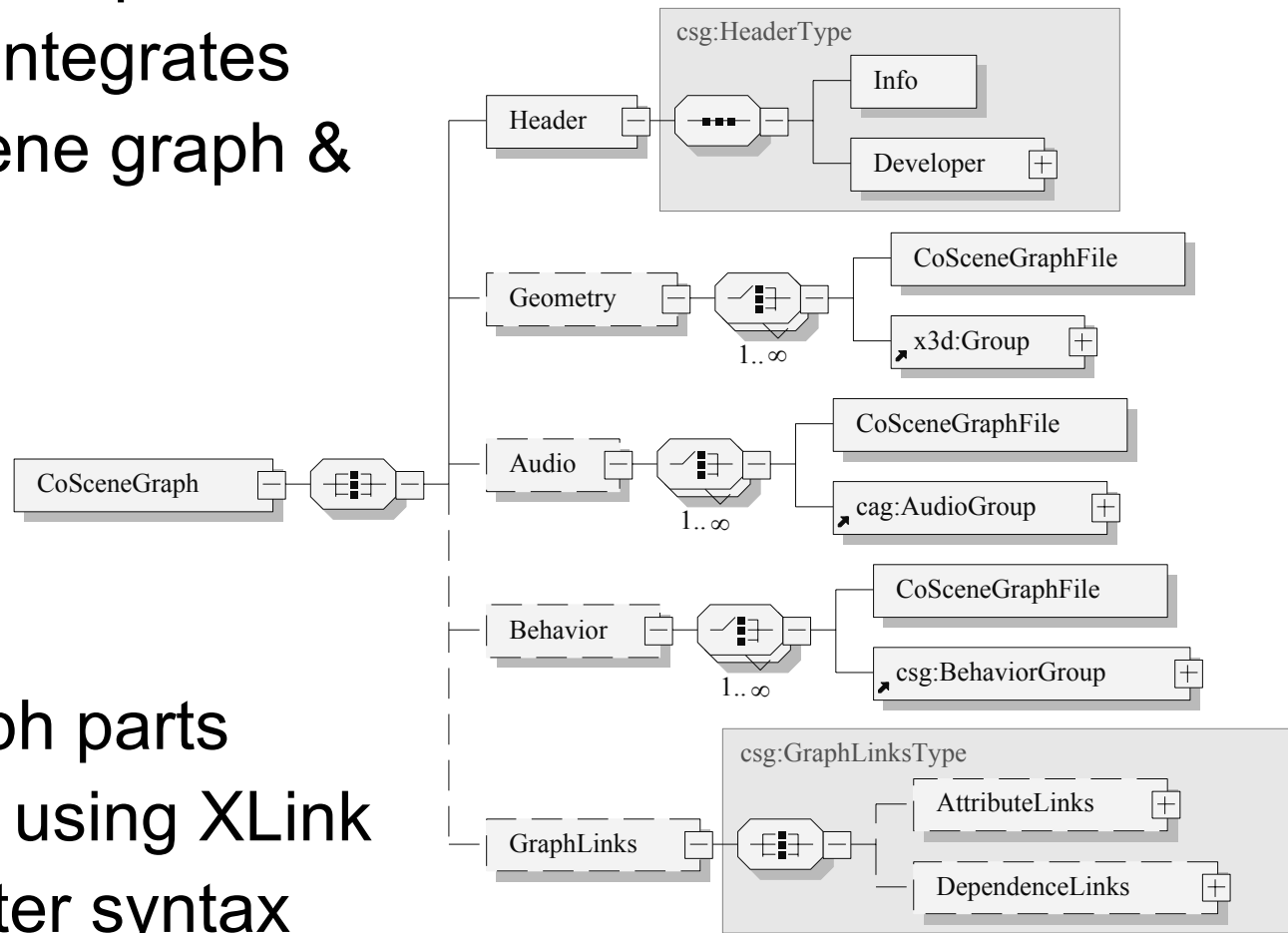
**CONTIGRA  
SceneGraph**

**X3D,  
Audio3D,  
Behavior3D**

*Implementation*

# Level 1: SceneGraph

- Component implementation language, integrates various scene graph & media files



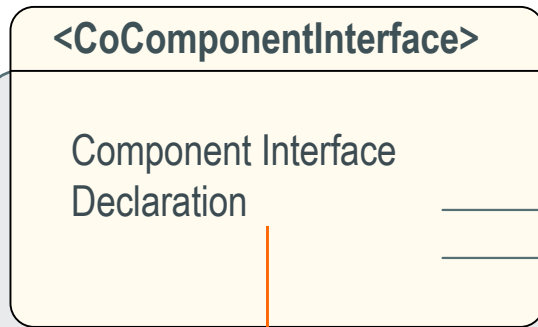
- Scene graph parts referenced using XLink with XPointer syntax

# CONTIGRA Markup Levels

Task XML Schema CONTIGRA Documents Other Resources

*Description  
Configuration  
Assembly  
Linking*

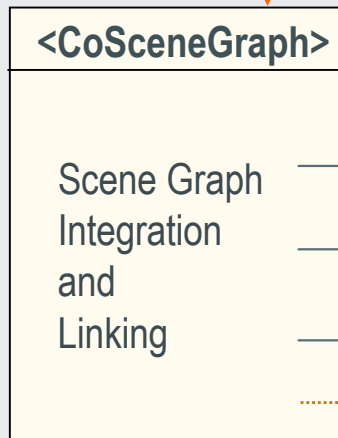
**CONTIGRA  
SceneComponent**



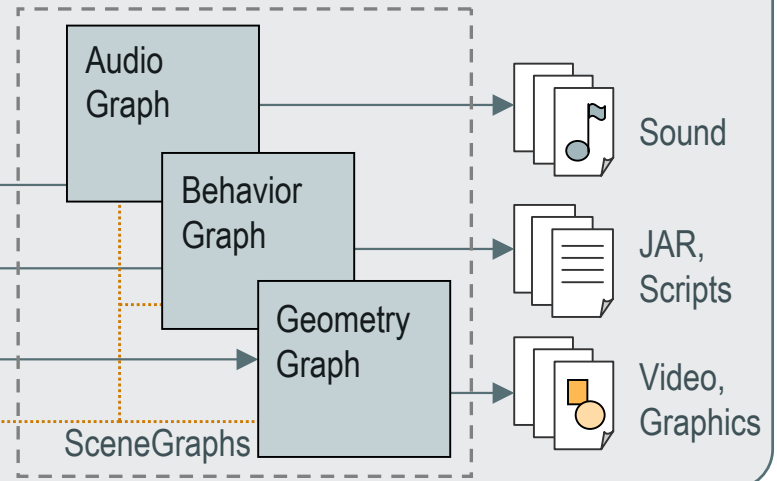
**Component**



**CONTIGRA  
SceneGraph**



X3D Profiles

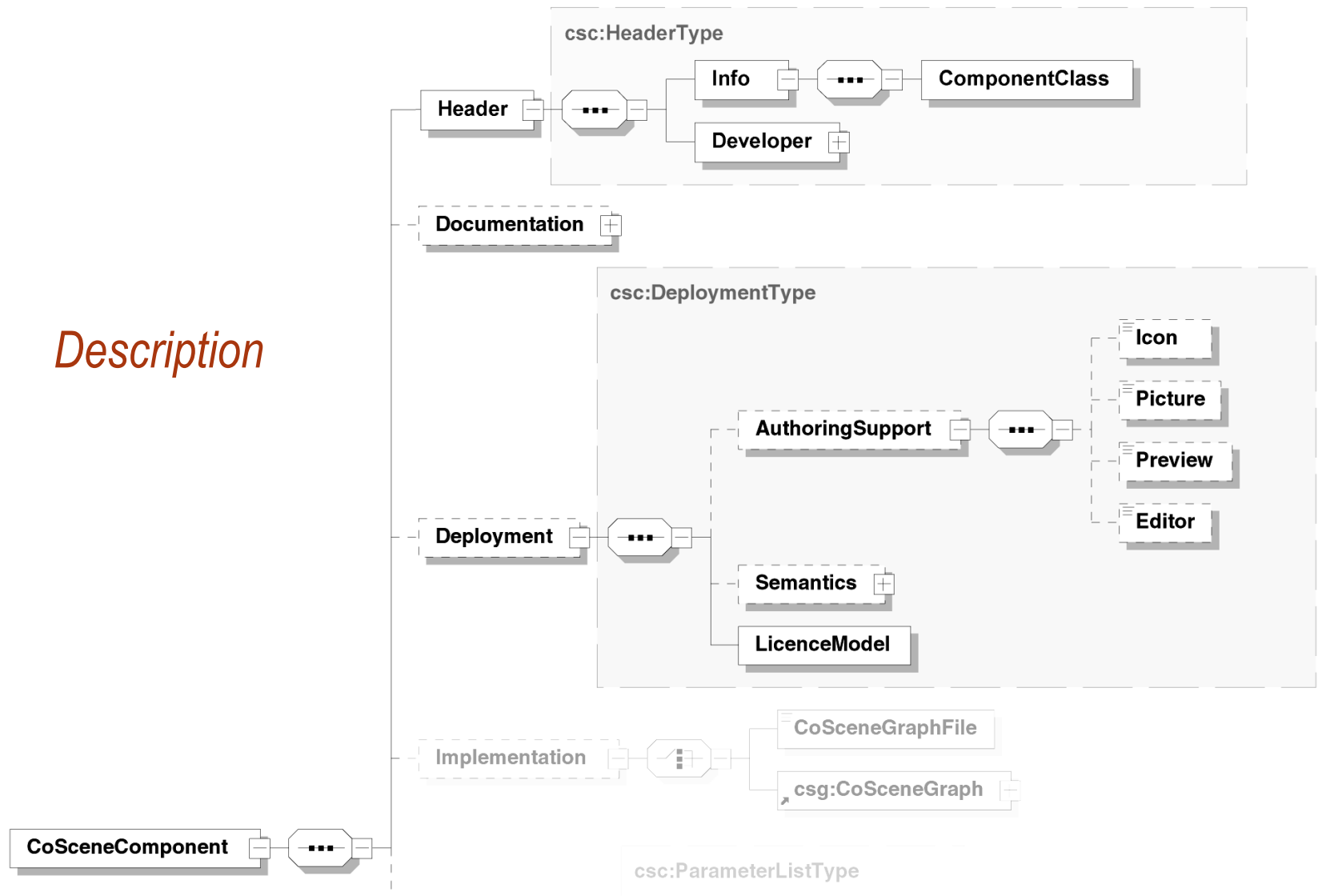


*Implementation*

X3D,  
Audio3D,  
Behavior3D

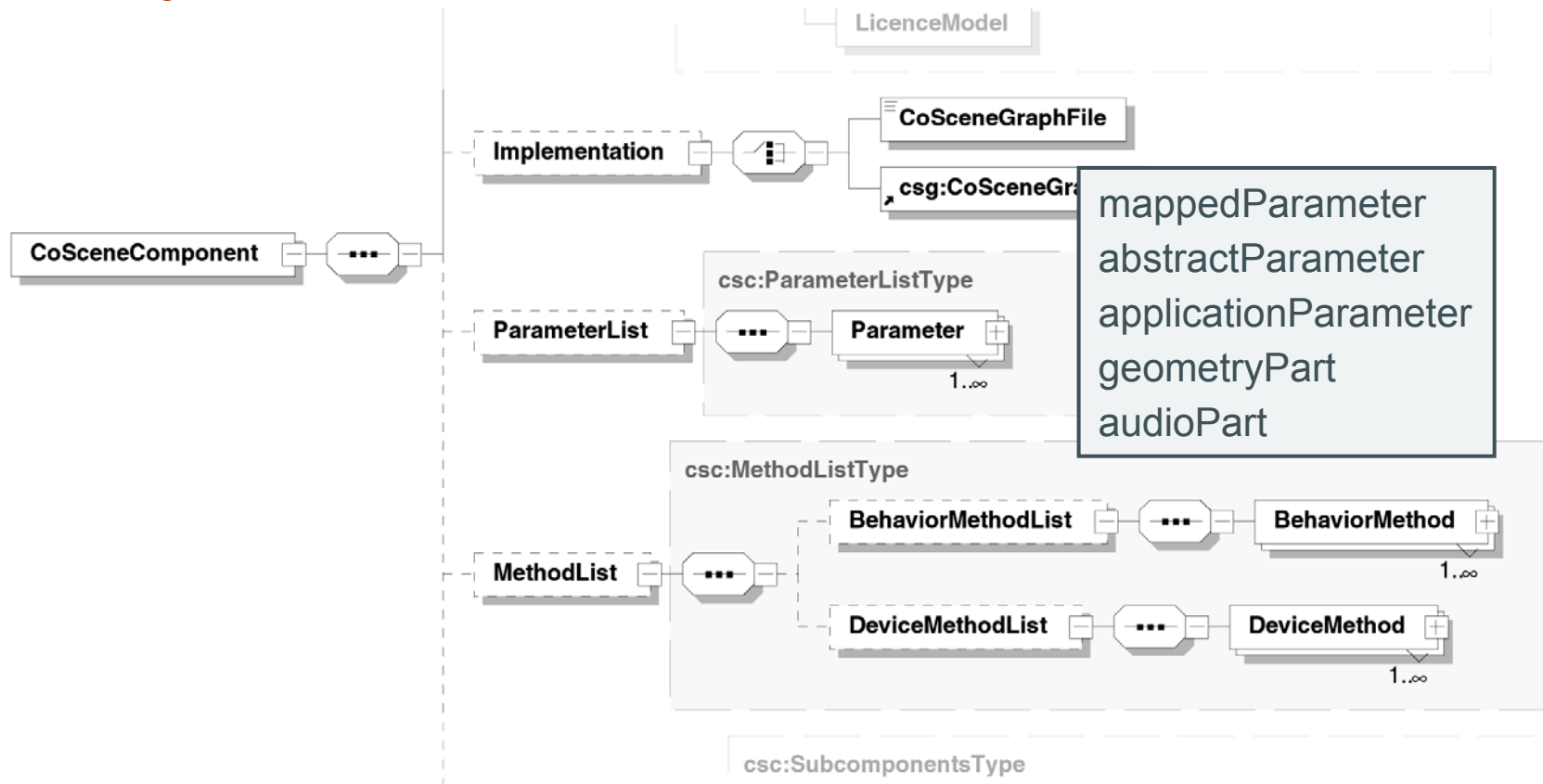
# Level 2: SceneComponent

*Description*



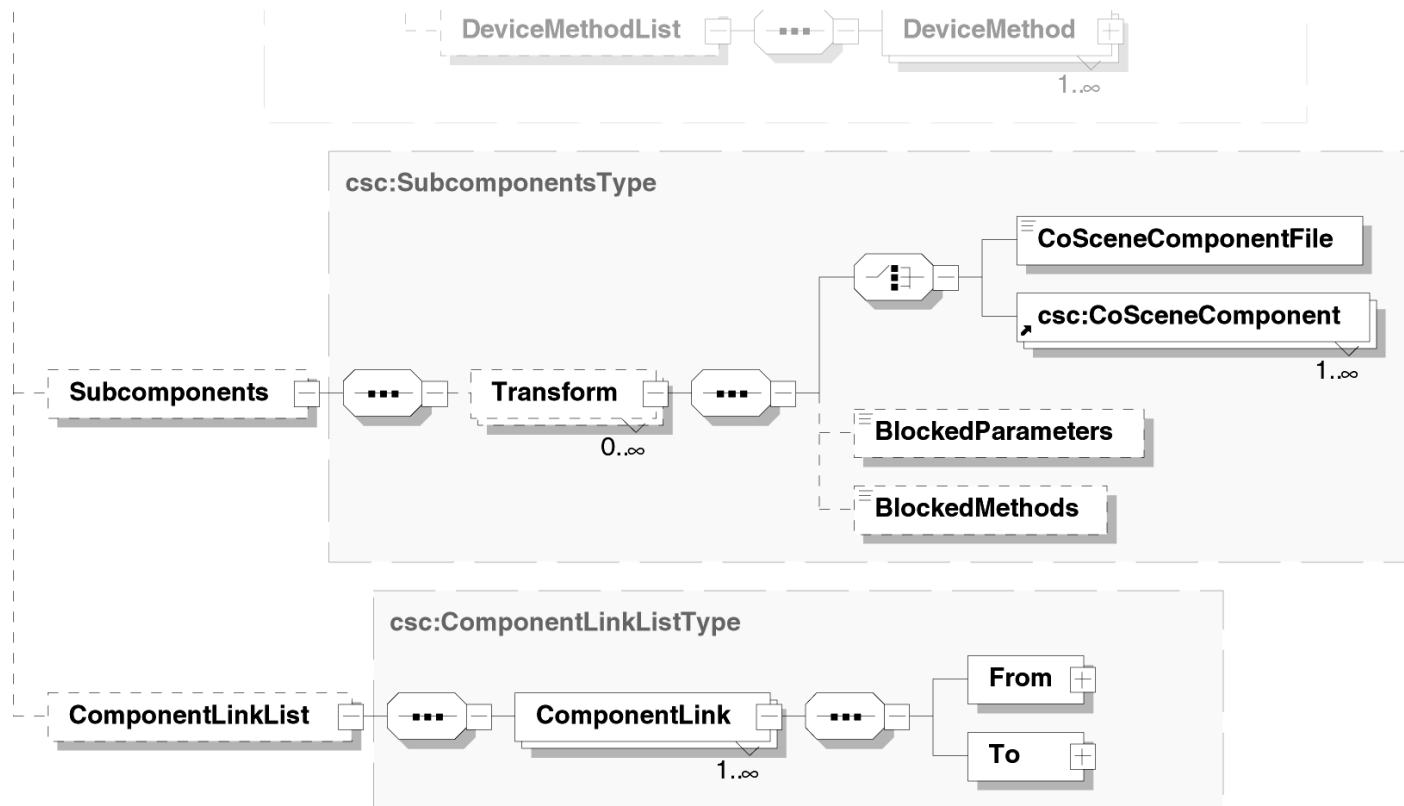
# Level 2: SceneComponent

## Interface & Configuration

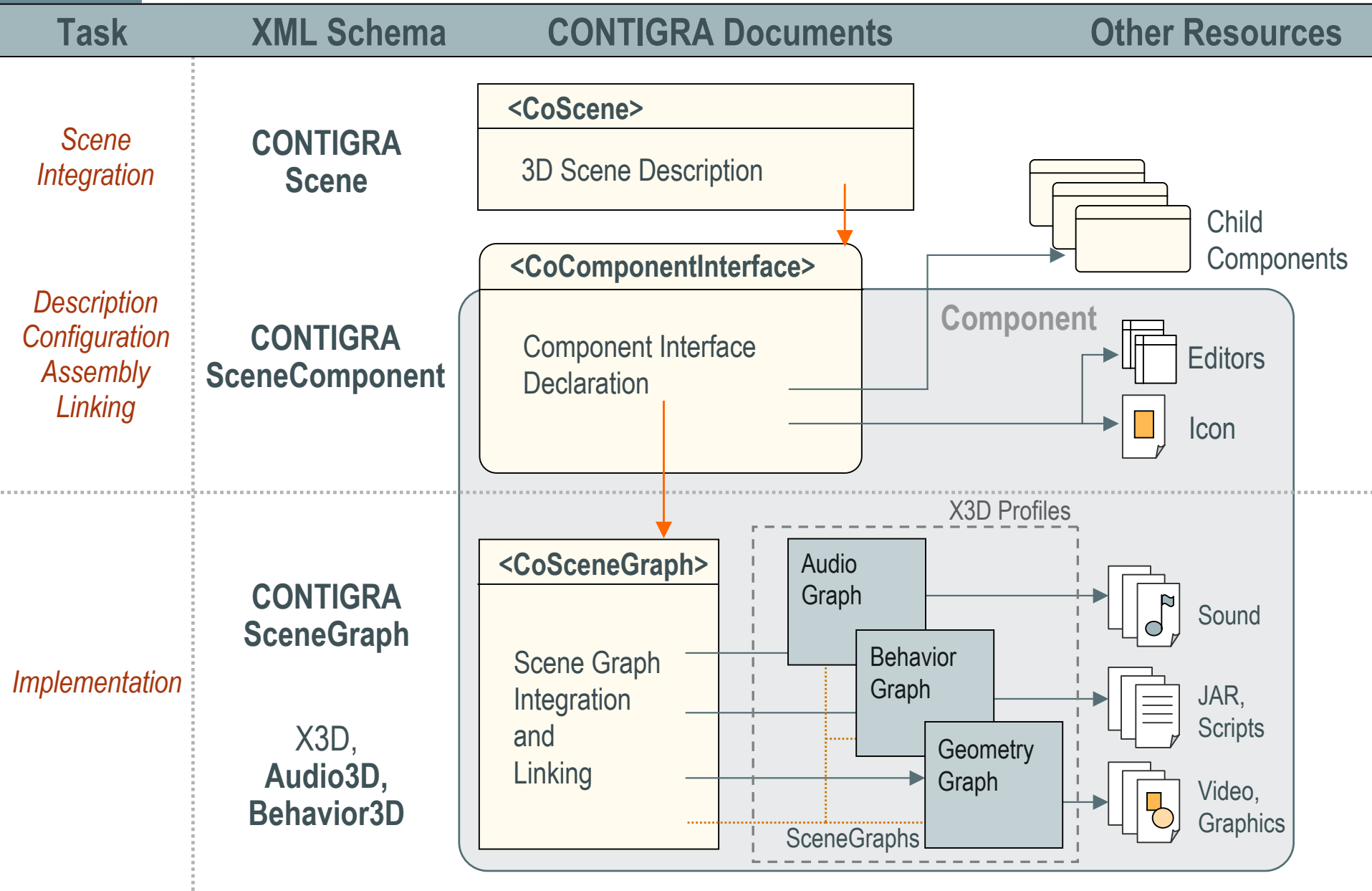


# Level 2: SceneComponent

## Assembly & Linking

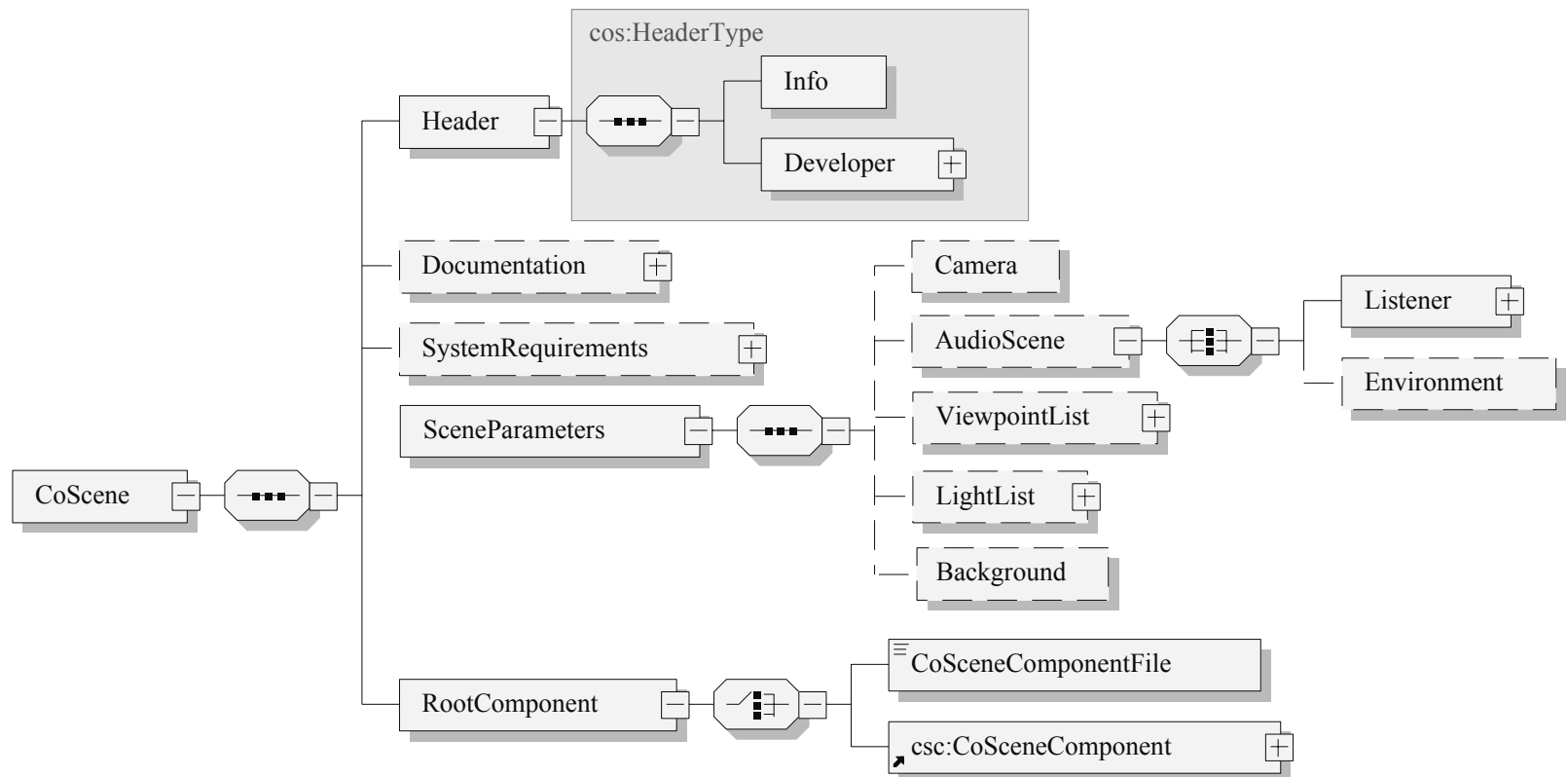


# CONTIGRA Markup Levels



# Level 3: Scene

- High-level component integration language
- Requirements for specific 3D web environment





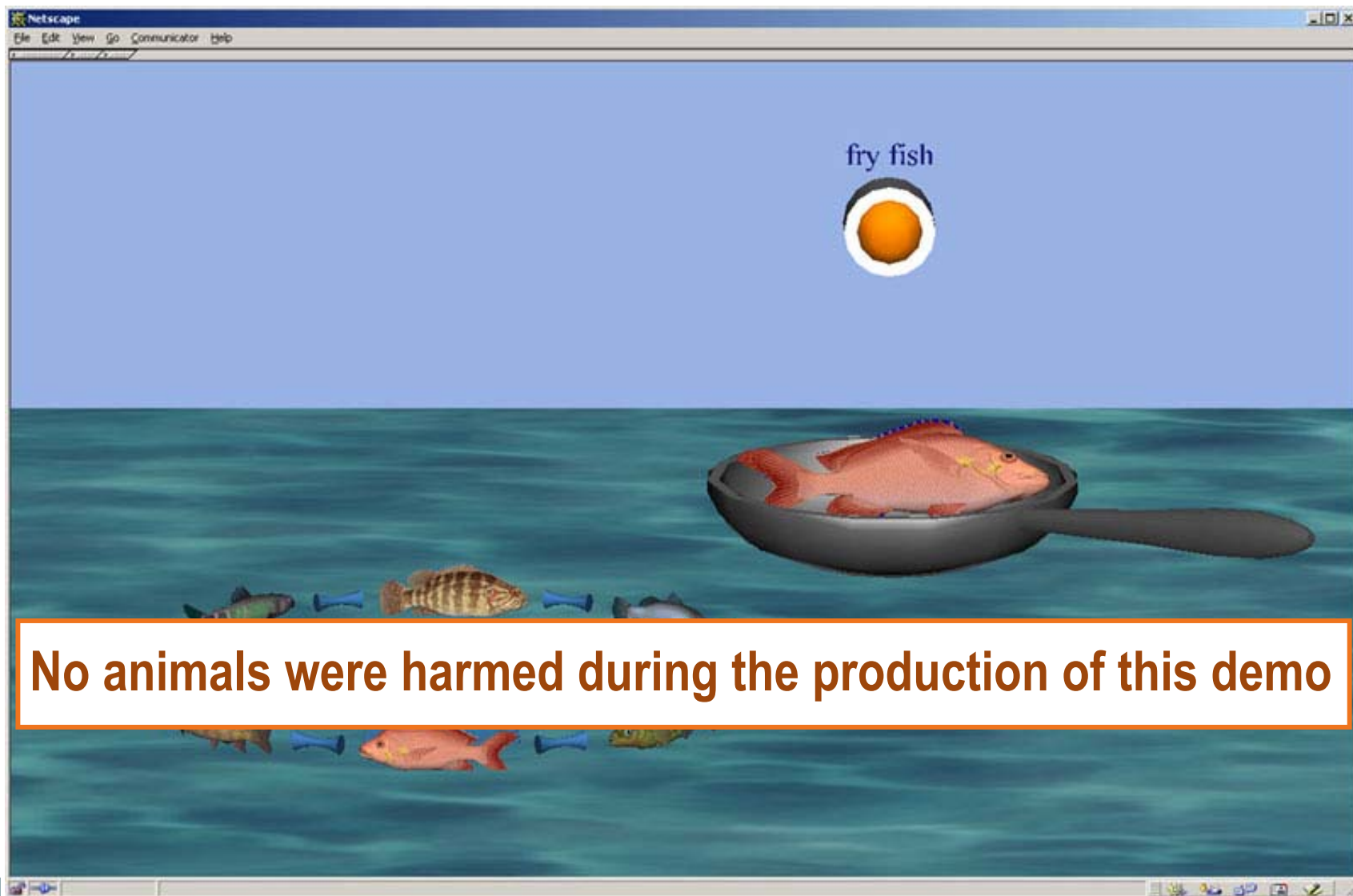
# 3D Application Example



Demo

Next

# 3D Application Example



**No animals were harmed during the production of this demo**

# Conclusion & Future Work

## ■ Major Features

- Componentization (design | deployment)
- Reuse on different abstraction levels
- Platform independence, abstraction to 3D formats
- Declarative approach, well suited for tool support

## ■ Next Steps

- CONTIGRA ComponentBuilder | SceneBuilder
- Development of runtime-framework (translators)
- More 3D widgets and example applications

# Discussion

*Thank you for your attention!*

**[www.CONTIGRA.com](http://www.CONTIGRA.com)**

