가상훈련 컨텐츠 제작을 위한 웹 기반 3D 모델 관리 시스템

Web-Based 3D Model Management System for Virtual Training Contents Production

김 연 우*, 윤 지 연*, 강 형 주*

Yeon-U Kim*, Ji-Yean Yoon*, Hyeong-Ju Kang*

요 약

컴퓨터 관련 산업의 발전과 함께 컴퓨터를 이용한 교육의 향상도 같이 이루어지고 있으며, 그 중에서 가상 훈련(Virtual Training, VT)은 가장 주목 받고 있는 미래형 교육 매체 중 하나이다. 이러한 추세에 따라 많은 양의 가상 훈련용 3D 모델들이 개발되고 있으나, 이들을 통합적으로 관리하여 재사용을 편리하게 하는 시스템에 대한 연구는 부족하다. 본 논문에서는 3D 모델들을 통합적으로 관리할 수 있는 시스템을 제안한다. 컨텐츠 저작도구들에서 편리하게 이용할 수 있는 모델 형식에 대해 조사하며, 중앙 서버에 저장하는 방법에 대해 연구한다. 3D 모델을 저장하기에 알맞은 DB 구조를 설계하고, 모델을 검색하고 인출하기 편한 방법에 대해 제안한다. 본 논문에서 개발된 관리 시스템 소프트웨어를 통해 3D 모델의 재사용이 촉진될 것이 기대된다.

Key Words : Virtual Training, 3D Model, Collada, Web-GL

ABSTRACT

Recently, the growth of the computer-related industries lead to the growth of the computer-aided education. Virtual training (VT) is one of the most interesting education methods. Many 3D models for the VT contents have been produced, but little research has been performed for the unified management system to promote the reuse of 3D models. This paper proposes the unified 3D model management system. The 3D model file formats are surveyed to find a format that can be easily used by many VT content production softwares. The 3D model files will be stored in a central DB server. The structure of the DB system will be designed for 3D models. The system provides an easy way to search and download a 3D model. With the proposed system, the reuse of 3D models is expected to be promoted.

* 한국기술교육대학교 컴퓨터공학과 (sizku1210@korea.ac.kr, jiyoun17@koreatech.ac.kr, hjkang@koreatech.ac.kr)
제1저자 (First Author) : 강형주
교신저자: 강형주
접수일자 : 2012년 4월 20일
수정일자 : 2012년 5월 23일
확정일자 : 2012년 6월 07일

Ⅰ. 서 론

컴퓨터 산업의 발전과 더불어 컴퓨터를 교육에 이용하려는 많은 시도가 이루어지고 있다. 이 중에서 가상현실을 이용한 교육(Virtual Traning, VT)은 가장 주목받고 있는 미래형 교육 매체 중 하나이며, 다양한 분야에 걸쳐 많은 양의 VT 컨텐츠들이 개발되고 있다. VT 컨텐츠들은 기본적으로 3D 모델들의 배치와 움직임 등을 통해 구성된다. 따라서 VT 컨텐츠의 양산은 필연적으로 많은 양의 3D 모델들의 제작을 동반하게 된다.

이렇게 VT 컨텐츠의 제작을 위해 개발된 3D 모델들의 양은 급격히 늘어나고 있는 추세이다. 한국기술교육대학교 능력개발교육원의 예를 들면, 2010년까지 10종 이상의 VT 컨텐츠들을 제작하였으며, 각 컨텐츠마다 적게는 수 개에서 많게는 십여 개의 3D 모델 파일들을 가지고 있다. 이와 같이 3D 모델의 수가 늘어남에 따라 이 모델들을 관리하는 시스템에 대한 요구도 커지고 있다

VT 컨텐츠 개발을 위한 3D 모델 관리 시스템에 있어서 가장 큰 목적은 재사용성이다. 3D 모델을 제작하는 것은 많은 노력과 자본이 들어가는 작업이다. 이러한 작업을 VT 컨텐츠마다 모두 새로 하는 것은 낭비일 것이고, 이전에 제작했던 3D 모델들 중 새로운 VT 컨텐츠에 적합한 것들을 최대한 많이 찾아내어 재사용하는 것이 효율적일 것이다.

재사용성을 높이기 위한 3D 모델 관리 시스템은 3D 모델들을 알아보기 쉽도록 보여 주어야 하며, 객체 정보 등을 통해 검색이 쉬워야 하고, 원하는 부분을 사용자가 보고 받을 수 있어야 한다. 그리고 다양한 지리적 위치에서 접근이 용이해야 할 것이다. 뿐만 아니라 다양한 VT 컨텐츠 제작 도구에서 이용할 수 있는 형식으로 3D 모델들을 저장해야 한다.

이렇게 VT 컨텐츠 개발을 위한 3D 모델 관리 시스템이 필요함에도 불구하고 그 간에 3D 모델 관리 시스템에 대한 관심이 적었다. 몇몇 소프트웨어들이 관리 시스템을 지원하기는 하나 대부분 상업적인 소프트웨어들이다.

이 논문에서는 VT 컨텐츠 개발을 위한 3D 모델 관리 시스템을 제안한다. 이 관리시스템에는 공통 3D 모델 저장 형식, 입력 소프트웨어, 저장 방법, 출력 소프트웨어 등이 포함된다. 제작된 3D 모델들을 공개된 형식으로 바꾸어 저장하며, 모델들을 프로젝트 별로 정렬하여 보여준다. 각 모델들은 하나의 파일에 들어 있는 객체 전체뿐만 아니라 그 내부의 개별 객체 단위로도 보거나 다운 받을 수 있도록 하였다. 지리적인 제한 없이 접근성을 높이기 위해서 웹 기반으로 작성하였다. 그리고 상업적인 용도의 기존 소프트웨어들과는 달리, 사용성을 높이기 위해 최대한 무료 소프트웨어들을 이용하여 제작하였다.

이 논문의 구성은 다음과 같다. Ⅱ절에서는 기존의 3D 모델 관리 시스템과 3D 모델 형식들에 대해 설명한다. Ⅲ절에서는 이 논문에서 제안하는 3D 모델 관리시스템에 대해 설명하고 그 작동 예를 보여 줄 것이다. 마지막으로 Ⅳ에서는 결과를 정리할 것이다.

Ⅱ. 기존 연구 분석

이 절에서는 기존에 연구되거나 개발된 3D 모델 관리 시스템에 대해 살펴 보고 많이 이용되는 3D 모델 형식에 대해 설명한다.

1. 기존의 3D 모델 관리 시스템

그 중요성에 비해 3D 모델 관리 시스템에 대한 연구는 그리 많이 이루어지지 않았다. D. Koller 등은 문화재 등의 3D 모델들을 관리하는 서버의 필요성 및 요구되는 기능 등에 제안하였다[1]. 이 논문에서 대상으로 하는 3D 모델은 항아리나 가구 소품에서부터 건축물, 마을, 도시 등까지도 포함한다. 저자들은 이러한 3D 모델들이 많은 나라와 단체들에 의해 개발되고 있으나, 검색 등이 용이하지 않고 모델의 품질 등에 대한 보장이 이루어지지 않고 있음을 지적하면서, 중앙집중식의 저장 장소와 관리 체계가 필요함을 말하고 있다. 그리고 이와 더불어 이러한 체계는 저작권 보호 방법, 모델 내용에 대한 메타데이터,

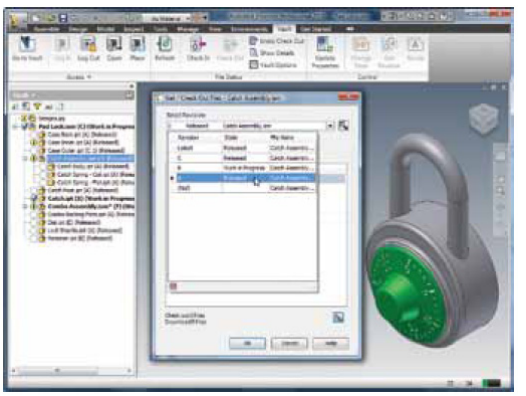


그림 1. Autodesk사의 Vault[2]
Fig. 1. Vault of Autodesk

표 1. 대표적인 3D 모델 형식들[5]
Table 1. 3D model formats

	Collada	ASE	X3D	3DS	OBJ
Basics	Format: XML Extension: .dae Usage: Open	Format: ASCII (no standard) Extension: .ase Usage: Proprietary	Format: XML Extension: Usage: Open, ISO-Standard (formerly VRML)	Format: Binary Extension: .3ds Usage: Proprietary	Format: ASCII Extension: .obj Usage: Open
Support	Many softwares	3D Studio Max Papervision Maya (via plugin) Doom 3 engine Unreal Editor	Blender (via plugin) 3DS Max (via plugin)	3D Studio (MAX) Various tools/repositories	SoftImage MeshLab 3DS Max Lightwave Art of Illusion Blender
Notes	Very extensive Input files can become very complex Widespread Java support only in beginning stages	Tied to the internal 3DS representation Basic but covers most required attributes	Very extensive Files can be complex ISO-standard but little prominent support Good library support Focus on 3D browsers	Proprietary, not fully explored Many 3D repositories use this format	Very widespread Very basic but sufficient, material information in external .mtl file No output libraries but simple enough to implement by oneself

버전 컨트롤 등에 대한 대책이 포함되어야 한다고 발표하였다. 3D 모델의 관리에 대해 다루었다는 측면에서 본 논문과 많은 연관이 있으나, 관리 시스템의 필요성을 말하는 것에 그치고 있다. 그리고 VT 컨텐츠 개발에 있어서는 모델을 재사용하는 사용자가 제한된 집단 내부에 속해 있으므로 저작권 보호 등의 이슈는 맞지 않는 면이 있다.

3D 모델 제작 소프트웨어를 판매하는 회사가 3D 모델을 관리하는 소프트웨어도 판매하는 경우도 있다. 그림 1이 대표적인 경우로, 3D Studio MAX를 판매하는 Autodesk 사의 Vault이다[2]. Vault는 제작된 3D 모델 데이터를 중앙 서버에 저장하고 관리하여, 3D 모델을 같이 사용하는 프로젝트 팀이 모델 정보를 서로 공유하고 재사용하기에 편리하도록 한다. 중앙 서버에 저장된 데이터는 적절한 접근 제한 정책을 통해 안전하게 관리할 수 있다. 그리고 저장된 모델들을 검색하고 읽은 뒤 수정하는 과정을 용이하게 하며, 수정과정에 있어서의 버전을 관리해 준다. Vault는 많은 이점을 가진 소프트웨어이나, 상업적으로 판매하는 소프트웨어이므로 비용이 많이 들고 특정 회사에 제약된다는 단점이 있다.

이 외에도 여러 사람이 공동 작업을 할 때 데이터 공유를 편리하게 해 주는 플러그인의 개발[3]이나 컨텍스트(context)에 기반하여 3D 모델을 검색하는 기법에 대한 연구[4]도 있었으나 본 논문에서 다루는 주제와는 조금 차이가 있다.

2. 3D 모델 형식

3D 모델의 데이터를 저장하는 모델 파일은 그 형식에 따라 파일의 크기도 달라지고 저장하는 데이터의 종류도 달라지는 등 형식에 따라 매우 다양하게 저장될 수 있다. 표 1은 3D 모델 파일의 형식들 중 대표적인 것들을 정리한 것으로써[5], 이 외에도 MAX, U3D, DWF, DXF, OBJ, STL 등 매우 다양한 종류의 형식이 존재한다.

이렇게 많은 형식들이 존재하나 그 들 사이의 호환성은 매우 적다. 대부분의 형식들이 특정 회사의 특정 소프트웨어를 위해서 만들어지다 보니, 특정 소프트웨어에서 제작되어 그에 따른 형식으로 저장된 3D 모델을 다른 소프트웨어에서 읽어 들이지 못하는 문제가 자주 발생하며, 이를 해결하기 위해 형식 변환기를 제작하면서 많은 시간과 노력을 소모하고 있다[6].

이와 같은 모델 형식의 비호환성을 해결하기 위해 표준 형식을 만들려는 노력이 있어 왔으며 그 중 대표적인 것이 Collada 형식이다[6][7]. Collada는 3D 어플리케이션, 게임, 플랫폼 등을 개발하는 여러 관련 회사들이 모여 만들어낸 형식이다. XML 기반의 파일 형식을 가지고 있으며 3D 모델의 버티스(vertices), 폴리곤(polygons), 텍스쳐(textures), 쉐이더(shaders), 광원(lights), 카메라(cameras) 등과 같은 정보들을, 어플리케이션 소프트웨어들 사이에 서로 교환할 수 있도록 만들어져 있다.

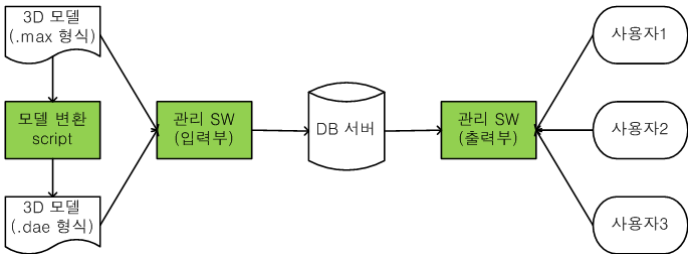


그림 2. 3D 모델 관리 시스템
Fig. 2. 3D model management system

III. 3D 모델 관리 시스템

본 논문에서 제안하는 3D 모델 관리 시스템은 그림 2와 같다. 모델을 저장해야 하므로 저장 DB 서버가 필요하고 이 서버에 입력하는 부분과 모델을 인출하는 부분이 필요하다. 그리고 저장 형식으로 변환하는 부분이 필요하다.

그림 2에서 음영으로 표시된 부분이 제작된 소프트웨어들이다. 그림에서 모델 변환 부분은 특정 형식으로 제작된 3D 모델을 Collada 형식으로 변환하는 부분이고, 관리 SW의 입력부에서는 원본 형식의 3D 모델과 Collada 형식의 3D 모델을 서버에 저장한다. 이와 함께 Collada 형식을 파싱하여 DB 테이블의 형태로 DB 서버에 저장한다. 관리 SW의 출력부에서는 사용자에게 3D 모델의 정보를 보여 주고, 사용자가 검색한 뒤 모델을 인출할 수 있게 한다.

1. 모델 형식 변환

3D 모델은 다양한 경로를 통해 다양한 소프트웨어에 의해 제작되고 그 소프트웨어들은 고유의 형식으로 모델들을 저장한다. 이러한 다양한 형식 그대로 서버에 저장할 경우, 재사용하는 입장에서는 3D 모델을 받은 뒤 사용하려는 소프트웨어에 맞게 모델의 형식을 변환해야 하는 번거로움이 생긴다. 이러한 작업을 사용자가 모델을 받을 때 마다 해야 한다면 매우 비효율적일 것이다.

이러한 비효율성을 해결하는 방법 중의 하나가 서버에 모델을 저장하기 전에 표준 형식으로 변환하여 저장하는 것이다. 이러한 용도로 사용되는 표준 형식은 대부분의 소프트웨어에서 읽어 들일 수 있는 형식이어야 할 것이다. Collada 형식은 그 구조가 공개되어 있고 대부분의 소프트웨어에서 지원하고 있으므로 이러한 표준 형식의 요건을 충족한다. 그러므로 본 논문에서 제안하는 관리시스템에서는 Collada 형식으로 모델들을 변환하여 저장한다.

본 논문의 관리시스템에서 사용하는 변환 스크립트에서는 사용자가 지정한 폴더 내에서, 지정된 형식들을 찾아서 Collada 형식으로 변환한다. 현재는 3D Studio MAX의 형식과 Maya의 형식에 대해 작성되어 있다. MAX 형식을 예로 들면, 지정된 폴더에 대해 MAX 형식의 파일을 모두 검색한다. 그리고 각각의 MAX 형식 파일에 대해, 3D Studio MAX의 스크립트를 작성한다. 이 3D Studio MAX 스크립트에서는 해당 파일을 읽은 뒤 Collada 형식으로 저장하는 동작을 한다. 스크립트를 작성하면 이를 3D Studio MAX에서 수행하여 3D 모델을 Collada 형식으로 저장한다. 이 때 3D 모델에서 사용하는 텍스쳐 파일들을 한 곳으로 모아서 추후에 관리를 용이하게 한다.

모델 변환을 3D Studio MAX 등의 상업 프로그램을 사용하지 않고 수행하는 것이 더 좋을 것이나, MAX 형식 등이 공개되어 있지 않아서 3D Studio MAX 등을 거치지 않고서는 Collada 형식으로 변환할 수 없다.

2. 관리 SW의 입력부

관리 SW의 입력부에서 3D 모델은 원본 형식과 변환된 Collada 형식, 두 개의 형식으로 저장된다. Collada 형식으로 변환할 경우 잃어버리는 정보가 있을 수 있으므로 원본 형식을 같이 저장하여, 사용자에게 정보 손실이 없는 원본 형식도 같이 제공할 수 있도록 한다.

모델의 저장은 두 가지 경로로 이루어진다. 하나는 3D 모델 파일들을 그대로 서버에 저장하는 것이고 다른 하나는 Collada 형식을 테이블 형태로 바꾸어서 DB 서버에 저장하는 것이다. 파일들을 그대로 저장하는 것은 저장한 형태 그대로 사용하기를 원하는 사용자를 위한 것이고, 테이블 형태로 저장하는 것은

표 2. main_tbl 테이블의 구조
Table 2. Structure of main_tbl table

column name	type	description
num	int	인덱스용 번호. 1씩 증가한다.
prj_name	varchar(100)	모델을 저장할 때 입력했던 프로젝트 이름
level	int	프로젝트 행이면 0, 프로젝트 내의 각 모델 파일들(DB들)에 대한 행이면 1
db_name	varchar(100)	각 모델파일들에 대한 DB 이름
dir	varchar(100)	해당 프로젝트의 파일들이 들어 있는 경로
file_name	varchar(100)	프로젝트 행이면 그 프로젝트에 대한 Collada 압축파일명, DB 행이면 그 DB에 대한 Collada 파일 이름
org_file_name	varchar(100)	프로젝트 행이면 그 프로젝트에 대한 max 압축파일명,

모델 내부의 객체 단위로 인출하기를 원하는 사용자를 위한 것이다.

관리 SW의 입력부는 웹페이지 형태로 제작하여 웹브라우저로 인터넷을 접속할 수 있는 곳에서는 어디에서든 3D 모델들을 저장할 수 있도록 하였다. 그리고 웹페이지로 제작되었으므로 계정 설정 등을 통해 접근 제한을 할 수도 있다.

3. DB 설계

본 논문에서 제안하는 관리시스템에서는 3D 모델을 테이블 형태로 DB 서버에 저장하기도 한다. DB 서버는 무료인 MySQL을 사용하였으나 다른 DB 서버 프로그램을 사용할 수도 있다.

DB 서버 안에는 여러 개의 DB들이 저장되며 하나의 DB는 여러 개의 테이블로 이루어져 있다. 본 관리시스템에서는 main_db의 main_tbl 테이블이 중심이 되고 각 3D 모델 파일들이 서로 다른 DB에 저장되는 형태로 설계하였다.

중심이 되는 main_db의 main_tbl 테이블은 표2와 같은 구조를 가지고 있다. 테이블에서 하나의 행은 하나의 프로젝트 혹은 한 개의 모델 파일에 해당한다. num 항목은 테이블의 행들을 구별하는 번호이며, prj_name 항목은 그 행에 해당하는 모델 파일이 속한 프로젝트의 이름이다. level 항목은 그 행에 해당하는 것이 프로젝트인지 모델 파일인지를 가리키며, db_name은 모델 파일일 경우 그 모델 파일이 저장된 DB의 이름을 가리킨다. dir 항목은 서버 내에서 실제 파일이 저장된 경로이며, file_name은 해당 모델의 파일 명을 가리킨다. org_file_name은 프로젝트에 해당하는 행에서만 값을 가지며 그 프로젝트의 모델 파일들을 압축한 압축 파일 명이다.

각 모델 파일들을 저장하는 DB는 7개의 테이블(asset, lib_eff, lib_geo, lib_img, lib_mat, lib_vis_scene, scene)로 구성되어 있다. 각각의 테이

표 3. 모델 DB의 테이블의 구조
Table 3. Structure of table in a model DB

column name	type	description
num	int	인덱스용 번호. 1씩 증가한다.
level	int	해당 tag의 깊이를 기록한다.
tag	varchar(80)	tag의 종류를 지정한다.
id	varchar(80)	tag에 id attribute가 있으면 그 값을 저장한다.
name	varchar(80)	tag에 name attribute가 있으면 그 값을 저장한다.
url	varchar(80)	tag에 url attribute가 있으면 그 값을 저장한다.
source	varchar(80)	tag에 source attribute가 있으면 그 값을 저장한다.
count	int	tag에 count attribute가 있으면 그 값을 저장한다.
attr	varchar(100)	tag의 attribute에 해당하는 문자열을 저장한다.
value	text (lib_geo에서는 longtext)	tag에 value가 들어 있으면 저장한다.
pcnt	int	lib_vis_scene에서 node tag이면 해당 객체의 triangle 개수를 저장한다.
vcnt	int	lib_vis_scene에서 node tag이면 해당 객체의 vertex 개수를 저장한다.

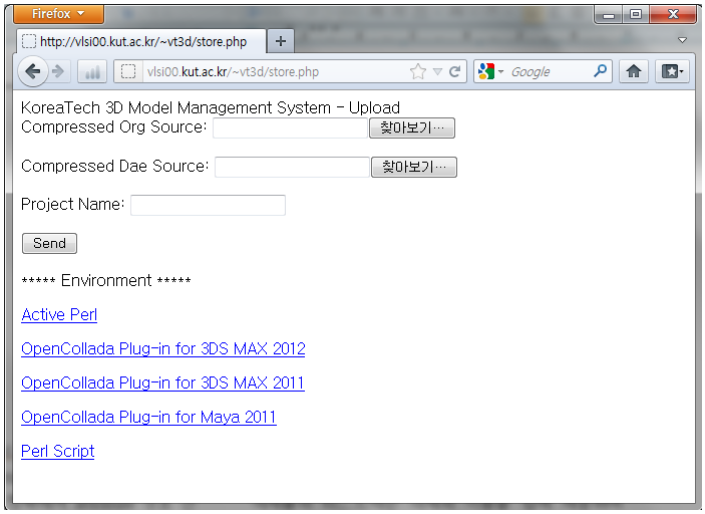


그림 3. 3D 모델 관리 시스템의 입력부
Fig. 3. Upload part of 3D model management system

블은 Collada 형식에서 asset, library_effect, library_image, library_material, library_visual_scene, scene에 해당하는 정보를 저장한다. 테이블에서 하나의 행에는 Collada 형식에서 한 tag에 있는 내용이 저장된다.

모델 파일을 저장하는 DB에서 테이블의 열은 표 3과 같이 구성되어 있다. num 항목은 테이블의 행들을 구별하는 번호이며, level은 해당하는 tag가 root로부터 얼마나 깊게 있는지, 즉 얼마나 많이 nest되어 있는지를 가리킨다. tag는 해당하는 tag의 종류를 저장한다. id와 name, url, source, count 항목들은 모두 tag에 들어 있는 attribute 정보들에서 해당되는 값들을 저장하며, attr 항목에서 attribute 정보 전체를 저장한다. attribute 정보들 중에서 id와 name 등을 따로 저장하는 이유는 이들 값들이 tag들을 검색하거나 타 정보를 추출할 때 유용하기 때문이다. value항목은 각 tag 안의 value값을 저장하며, pcnt와 vcnt는 해당 행이 3D 객체일 경우 그 객체 안의 triangle 개수와 vertex 개수를 저장한다.

4. 관리 SW의 출력부

관리 SW의 출력부는 접근성을 고려하여 입력부와 마찬가지로 웹 기반으로 개발되었다. 가상훈련 컨텐츠를 제작할 때는 상황에 맞추어서 적절한 모델을 검색해서 인출해야 한다. 이러한 과정을 원활히 지원하기 위해서는 어디서나 DB 서버에 접속하여 모델을 검색/인출할 수 있는 환경이 제공되어야 한다. 본 논문과 같이 웹 기반으로 개발하게 되면, 네트워크가 연결되고 웹 브라우저를 사용할 수 있는 곳에서는 어디서든지 DB 서버에 접속하여 모델을 검색/인출할 수 있는 장점이 생긴다.

웹 기반 출력부의 또 다른 장점은 접근 제약을 통일성 있게 할 수 있다는 것이다. 컨텐츠 개발을 용이하게 하기 위해 DB 서버에 대한 접근성을 높이는 것이 필요하나, 모델에 대한 저작권 등이 있으므로 DB 서버에 대한 접근은 권리를 가진 일부 이용자에게만 허용되어야 할 것이다. 웹 기반으로 출력부를 개발하면, 이를 이용하는 기관의 홈페이지 등과 연계해서 등록되고 허용된 사용자만 접속할 수 있도록 제어할 수 있게 된다.

본 관리 SW의 출력부에서는 웹 페이지를 통해 모델에 관련된 다양한 정보를 보여준다. 저장된 프로젝트 별로 어떤 모델 파일들이 있는지 보여주며, 각 파일 별로 그 안에 있는 객체들의 리스트를 보여준다. 객체들의 리스트에는 객체의 이름을 함께 제공하여 사용자가 이름을 통해 어떤 객체인지 알 수 있도록 한다. 그리고 각 객체 별로 vertex의 개수와 triangle의 개수를 같이 보여 줘서 객체가 얼마나 자세히 모델링 되어 있는지에 대한 정보도 제공한다.

이 외에도 본 논문에서 개발한 관리 SW의 출력부는 미리보기 기능을 제공한다. 검색을 통해 원하는 모델을 찾았다 하더라도 인출해서 사용하기 전에 원하는 모양인지 확인할 필요가 있을 것이다. 본 관리 SW에서는 Web-GL 기반의 3D 모델 뷰어[8]를 통해 선택한 3D 모델의 객체를 미리 확인할 수 있도록 하였다.

그림 4. 3D 모델 관리 시스템의 출력부
Fig. 4. Download part of 3D model management system

모델 및 객체의 검색은 웹 브라우저가 내장한 검색 기능을 활용하여 객체 이름을 찾는 방식으로 이루어진다.

5. 관리 SW의 동작 화면

관리 SW 입력부의 웹 페이지 화면은 그림 3과 같다. 프로젝트 별로 모델 파일들을 압축하여 입력하게 되어 있으며, 원본 형식과 Collada로 변환된 형식의 압축 파일 두 개를 동시에 저장한다. 프로젝트의 이름도 같이 입력한다. 그 아래에는 각종 환경 프로그램을 다운로드할 수 있도록 링크를 걸어 놓았다.

관리 SW 출력부의 웹페이지 화면은 그림 4와 같다. 프로젝트 별로 구별이 되며 각 프로젝트의 압축 파일을 다운로드할 수 있다. 프로젝트 아래에는 모델 파일별로 구별이 되어 있으며 해당 3D 모델을 미리 보기 할 수도 있다. 각 모델들 아래에는 속해 있는 객체들의 이름이 나열되어 있으며, 객체 별로 vertex의 개수와 triangle의 개수에 대한 정보를 같이 제공한다. 객체 별로 미리보기를 할 수도 있으며 그림 5는 한 객체를 선택하여 미리보기를 한 화면이다.

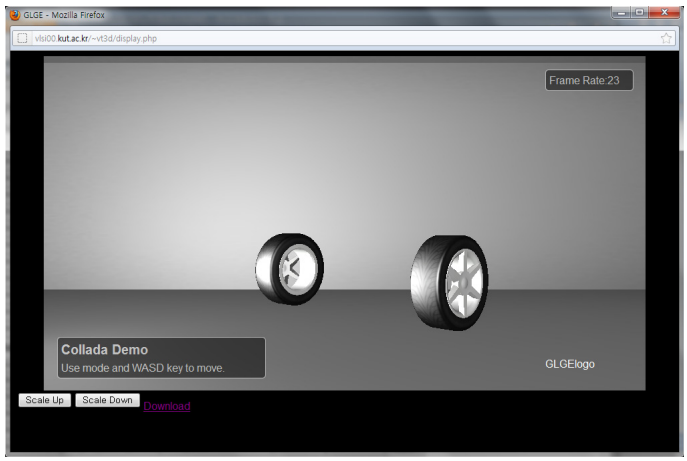


그림 5. 3D 모델 객체 뷰어
Fig. 5. 3D model instance viewer

V. 결 론

본 논문에서는 가상훈련 컨텐츠를 제작하기 위해 사용하는 3D 모델들의 통합관리 시스템을 제안하였다. 다양한 컨텐츠 저작도구들에서 이용하기 위해 3D 모델들을 공개된 표준 형식인 Collada 형식으로 변환하여 저장하며, 중앙 DB 서버에 저장함으로써 통일된 관리를 가능하게 하였다. 모델을 저장하거나 검색/인출하는 기능은 웹 기반으로 개발하여 접근성을 높였다. 모델 및 그 아래의 객체 단위로 다양한 정보를 제공함으로써 검색 등을 용이하게 하였고 미리보기 기능도 같이 제공한다. 무료 소프트웨어를 이용하여 시스템을 구성함으로써 사용성을 높였다. 본 관리 시스템을 통해 3D 모델들의 재사용을 촉진할 수 있으며 3D 모델을 사용하는 다른 어플리케이션에도 적용할 수 있을 것으로 기대된다.

감사의 글

이 논문은 2011년에 진행된 “3D 모델 통합 관리 방안 및 3D 모델 DB 관리 소프트웨어 연구” 과제를 통해 한국기술교육대학교 능력개발교육원으로부터 지원받았음.

참 고 문 헌

[1] David Koller, Bernard Frischer, and Greg Humphreys, “Research challenges for digital archives of 3D cultural heritage models,“ *ACM Journal on Computing and Cultural Heritage*, Vol. 2, No. 3, Article 7, Dec. 2009.

[2] Autodesck, Inc., available at: http://www.autodesk.co.kr.

[3] 권태숙, 이승룡, “협력시스템에서 3D 스튜디오 맥스 플러그인 설계 및 개발,” 정보과학회 논문지: 컴퓨팅의 실제, 제 7권, 제 5호, pp. 498-509, Dec. 2001.

[4] Matthew Fisher and Pat Hanrahan, “Context-based search for 3D models,“ *Proceedings of ACM SIGGRAPH Asia*, 2010

[5] Comparison of 3D file formats, available at: http://code.kuederle.com/3dformats.

[6] Remi Arnaud and Mark C. Barnes, *Collada: Sailing the gulf of 3D digital content creation*, AK Peters, Ltd., Aug. 2006.

[7] COLLADA - Digital asset and FX exchange schema, available at: http://www.Collada.org.

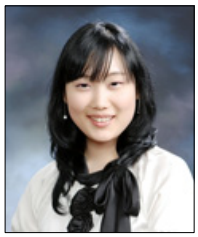
[8] GLGE: WebGL for the lazy, available at: http://www.glge.org.

김 연 우 (Yeon-U Kim)

2010년 2월 : 한국기술교육대학교 인터넷공학 졸업 (공학사)
2011년 3월~현재 : 한국기술교육대학교 컴퓨터공학과 (공학석사)

<관심분야> parallel programming, gpgpu, cuda, AR

윤 지 연 (Ji-Yean Yoon)



2010년 2월 : 한국기술교육대학교 인터넷공학 졸업 (공학사)
2011년 3월~현재 : 한국기술교육대학교 컴퓨터공학과 (공학석사)

<관심분야> 증강현실, 영상처리, 컴퓨터비전, 모바일 어플리케이션

강 형 주(Hyeong-Ju Kang) 정회원

1998년 2월 : KAIST 전기및전자공학과 졸업
2000년 2월 : KAIST 전자전산학과 석사
2005년 2월 : KAIST 전자전산학과 박사
2005년 3월~2006년 1월: (주)매그나칩반도체 선임연구원
2006년 2월~2009년 2월: (주)지씨티리써치 선임연구원
2009년 3월~현재: 한국기술교육대학교 컴퓨터공학부 조교수

<관심분야> VLSI 설계 및 CAD, 3D 모델링