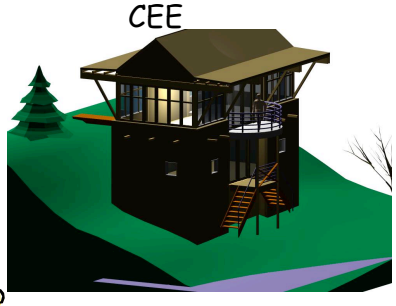
CEE 498A - AutoCAD Design

3D Rendering in AutoCAD?

Week 6 Lecture 2

Creating 3D Models

Computer Aided Engineering in CEE



The "Decatur Cabin". Rendering by Philip Stephenson, Autumn, 1999

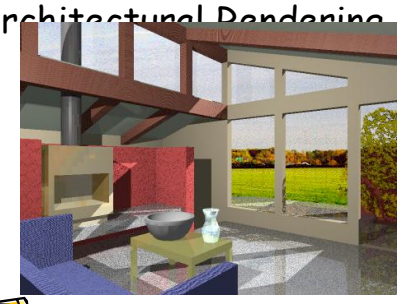
3D Models - Why?

Broadway Malyan, International Architects' visualization team meets customer demand for photo realistic renderings working with Autodesk Architectural Desktop designs further enhanced in Autodesk VIZ.

3D Models - Why?

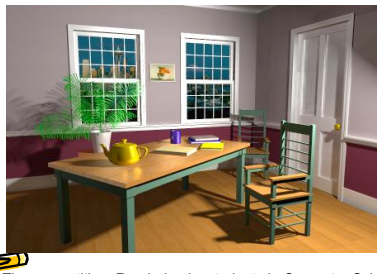
Rose-Hulman Institute of Technology is an educational facility in Terre Haute, Indiana designed by VOA Associates, Inc.using Autodesk Architectural Desktop and Autodesk® VIZ.

Architectural Rendering



The "House". Rendering by Vorapat Inkarojrit, Autumn, 1998

Architectural Rendering



The competition. Rendering by students in Computer Science.

The Data

WIREFRAME
An early, flexible, but less-often used modeling approach simply creates lines in space. No hidden-line removal is possible, so models are perpetually "see-through".

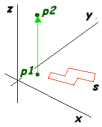
BOUNDARY REPRESENATION
Most 3D modeling is done to create models for rendering, and the history of rendering technology has (so far) focused on the development of "photorealistic" imagery--synthetic pictures which provide enough of the cues of the real world to fool our eyes. Such images require the existence of surfaces in the model, which boundary representation (or b-rep) data provides.

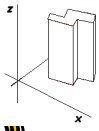
.

SOLID
Beyond simply providing a basis for photorealistic rendering, the model may need to provide answers to important questions, like,
"Does the crankshaft hit the engine block when it rotates?" or "What if we cut away the corner of the building to reveal an entry way?", or "Can we fabricate this part using a numerically controlled milling machine?" These involve processing or manipulating the existing data as if it were a true solid. It turns out that solid modeling depends on knowing the "inside" of the object from the rest of the universe

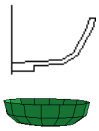
How do we do it?

Most 3D modeling consists of defining 2D "controls" for 3D information in various ways. Once we have some data, modifiers such as copy can be combined with basic transformations (move, rotate, scale, mirror) to make new data. Operations: **Extrusions, Revolutions, Sweeps, Boolean Operations**

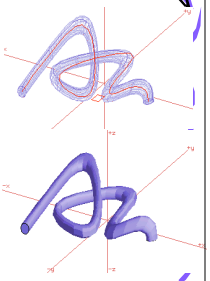




Extrusion



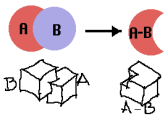
Revolusion

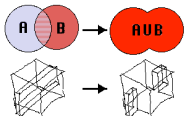


Sweep

How do we do it?

Most 3D modeling consists of defining 2D "controls" for 3D information in various ways. Once we have some data, modifiers such as copy can be combined with basic transformations (move, rotate, scale, mirror) to make new data. Operations: **Extrusions, Revolutions, Sweeps, Boolean Operations**





Boolean Operations

3D Axes

- Right hand rule
- Relative coordinates
  - Spherical (r, □, □)
  - Expressed: @ r < □ < □
- Create wire frames in XY plane
- 3DMESH: open / closed (M,N)
- Polyface mesh: PFACE

Commands

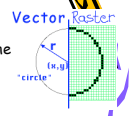
- BOX, CONE, CYLINDER, SPHERE, WEDGE
- EXTRUDE, REVOLVE
- Composite solids: UNION, SUBTRACT
- Editing: ROTATE3D
- Modifying: 3DARRAY, MIRROR3D
- Sectioning Solids: SECTION
- Hiding lines: DVIEW

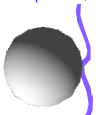
Rendering

- The challenge for the computer graphics programmer is to re-create the visual effect, without the actual materials and innumerable photons of the real world.
- We see objects because of the light reflected from or transmitted through their surfaces. Differences in surface quality, such as color, roughness, and transparency have an effect on the appearance of the shape. The light which falls on the surfaces is emitted by one or more light sources. Sometimes the light is blocked, causing a shadow to fall on the surface. Finally, if the light sources and objects are moved around, the surfaces of the object will change in appearance (getting darker or lighter, for example).
- What we see may depend on
  - The material/surface quality of the object
  - The light distribution qualities of the light source
  - The relationship between object, light source(s) and viewer
- Each of these issues is subject to a variety of approaches, ranging from [illegible]

Rendering – Things to Consider

- Data – is the image a raster or vector image. This determines how the 3D model is created
- View – from where will the image be viewed? Where's the camera / eye?
- How to deal with hidden lines / surfaces – different algorithms are used for different types of data
  - **CULLING:** This computationally inexpensive, approach does an imperfect job in most cases, but it can form an effective pre-processor to more complex algorithms. The results are object-data.
  - **DEPTH SORTING:** A technique dependent on the ubiquity of raster displays and plotters, the image is built by changing the order in which data is processed. Results in object data images.
  - **Z-BUFFERING:** A raster approach. Whenever a polygon is rasterized into the display memory, the algorithm asks (for each pixel), whether the new polygon surface is closer to the eye than the old one.
- Shading: *color intensity = ambient + shading + highlight*
  - Shading term contributes most of the light variation. Flat shading gives the same level of shading to each surface, smoothed takes into account curved surfaces
- Light source: distant light, point source, spot
- Shadows:
  - no connection with shading / here the issue is what is in the way of the light
- Textures: vary the intensity of the light to account for texture





Shading

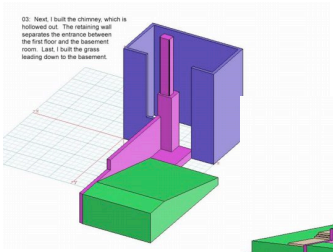
distant light

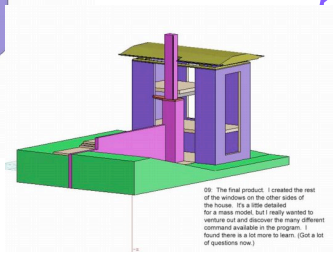
near field point source

Software

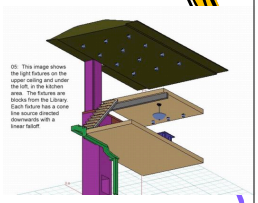
- form•Z – generate 3D surfaces & accomplish the rendering
- Autodesk
  - AutoCAD – very limited 3D capabilities
  - Inventor – generate 3D surfaces
  - Autodesk Architectural Desktop (3D Studio Max) rendering
- Microstation – similar to Autodesk
- Maya
- Renderman

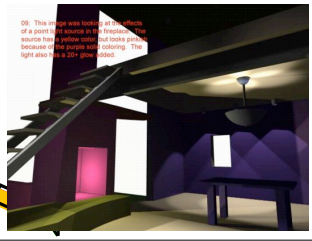
The Results – basic models



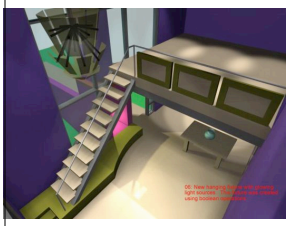


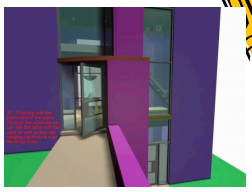
The Results - lighting

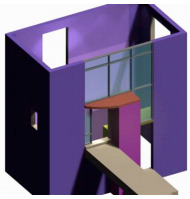




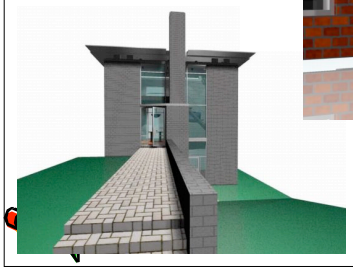
The Results - details

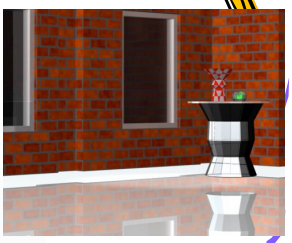




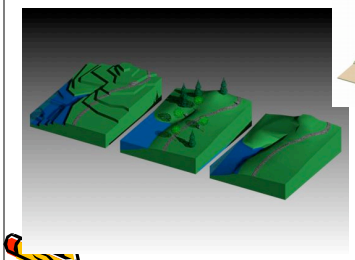


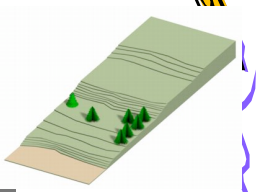
The Results - texture



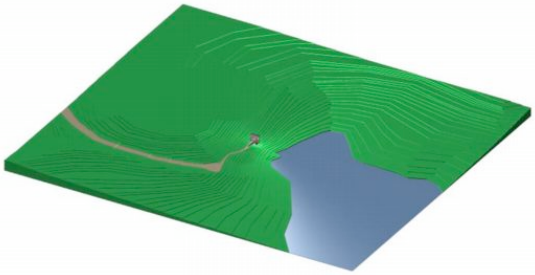


The Results – irregular forms

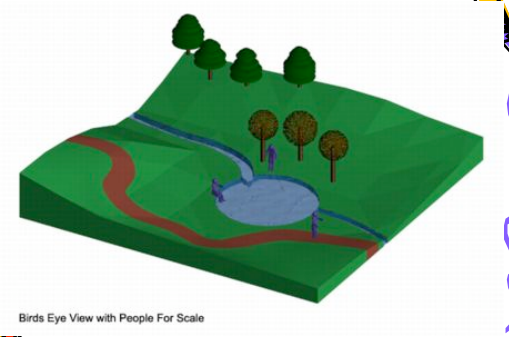




The Results – irregular forms



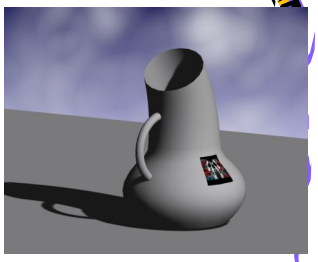
The Results – irregular forms



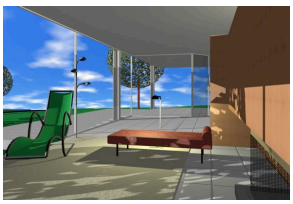
Birds Eye View with People For Scale

The Results – complex forms

07: Here is a Coke bottle with a decal lable off of a real bottle. The plaster bottle is filled with a liquid.



The Results



Animation

http://online.caup.washington.edu/courses/arch481/00HomePage/4.Gallery/D.Spring%202003/A.Final%20Video/0.default.html

Credits:

Brian Johnson and UW Arch 481 Students, Spring 2003

Maya / Renderman