KribiBench v1.0

User Guide

Revision 1.0

www.adeptdevelopment.com

END-USER LICENSE AGREEMENT

The software product is protected by copyright and other intellectual property laws and treaties. Adept Development or its suppliers own the title, copyright, and other intellectual property rights in the Product. The Product is licensed, not sold.

Rights and Restrictions. Adept Development grants you a worldwide, nonexclusive, nontransferable, royalty-free right to use and install an unlimited number of copies of the software solely for internal use, such copies shall be used only for internal purposes for demonstration, test, or evaluation and are not to be republished or distributed (either in hard copy or electronic form) without a priori written permission from Adept Development. You may not reverse engineer, decompile, or disassemble the Software Product.

Termination. Without prejudice to any other rights, Adept Development may cancel this end user license agreement if you do not abide by the terms and conditions of this end user license agreement, in which case you must destroy all copies of the software product and all of its component parts.

You agree that Adept Development may collect and use technical information you provide as a part of support services related to the Software Product.

Disclaimer of warranties. to the maximum extent permitted by applicable law, Adept Development and its suppliers provide the product and support services (if any) as is and with all faults, and hereby disclaim all other warranties and conditions, either express, implied, or statutory, including, but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, and of lack of negligence, all with regard to the product, and the provision of or failure to provide support services. also, there is no warranty or condition of title, quiet enjoyment, quiet possession, correspondence to description or non-infringement with regard to the product.

Exclusion of incidental, consequential, and certain other damages. To the maximum extent permitted by applicable law, in no event shall Adept Development or its suppliers be liable for any special, incidental, indirect, or consequential damages whatsoever (including, but not limited to, damages for loss of profits or confidential or other information, for business interruption, for personal injury, for loss of privacy, for failure to meet any duty including of good faith or of reasonable care, for negligence, and for any other pecuniary or other loss whatsoever) arising out of or in any way related to the use of or inability to use the product, the provision of or failure to provide support services, or otherwise under or in connection with any provision of this end user license agreement, even in the event of the fault, tort (including negligence), strict liability, breach of contract, or breach of warranty of Adept Development or any supplier, and even if Adept Development or any supplier has been advised of the possibility of such damages.

IMPORTANT READ CAREFULLY: This End-User License Agreement is a legal agreement between you (either an individual or a single entity) and Adept Development for the Adept Development software product identified above, which includes computer software and may include associated media, printed materials, and "online" or electronic documentation, and they supersede all prior or contemporaneous oral or written communications, proposals, and representations with respect to the Software Product or any other subject matter covered by this end user license agreement.

YOU AGREE TO BE BOUND BY THE TERMS OF THIS END USER LICENSE AGREEMENT BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE PRODUCT. IF YOU DO NOT AGREE, DO NOT INSTALL OR USE THE PRODUCT.

1. Trademarks & Copyrights

Adept Development, the Adept Development Logo, Kribi 3D Engine are registered trademarks of Adept Development S.r.l. and Ocean Side Corp.

Windows 98, Windows Me, Windows 2000, Windows XP and Microsoft are registered trademarks of Microsoft Corporation.

Celeron, Pentium and Xeon are registered trademarks of Intel Corporation

Duron and Athlon are registered trademarks of Advanced Micro Devices, inc.

All other trademarks are the property of their respective owners.

Some photos used as textures are Copyright © COREL Corporation

2. Introduction

2.1 The Kribi Project

Since more than 10 years we have developed and constantly refined a set of 3D technologies used in diverse commercial products. All these technologies are now grouped together in our ***Kribi*** project (to be pronounced « kreebee »). A key part of the project is the Kribi rendering engine. It is a ***pure software engine*** because we strongly believe that such solutions will be very important in the near future. We are convinced that they should replace entirely engines based on specialized “graphic processors” in the forthcoming years.

For more information on our Kribi project or to hear about the availability of new versions of the KribiBench check regularly www.adeptdevelopment.com/kribi

2.2 Why KribiBench ?

An essential part of the development of a 3D rendering engine, much like for a race-car engine, is **tuning** for the best possible performances on actual machines. A common task is to measure timings after each change in the program code to be sure that speedups are achieved. This is typically done rendering a sequence of images of some 3D model. We have added for this purpose a special notation in the Kribi file format. This way, every model can be used as a particular test. We have already dozens of such models, some are realworld scenes from our commercial packs some are custom-made to focus on a specific area of the renderer.

For a given version of the Kribi engine, the time it takes to render the sequence of frames can also be used to compare the performance of diverse computers, i.e. it can be used as a ***benchmark*** application. Our marketing team has identified here a good opportunity to catch public eyes on Kribi by releasing a useful freeware, KribiBench was born this way.

2.3 What is KribiBench actually testing ?

CPU(s) :	Heavily
Memory :	Yes
Chipset :	Yes
Graphic Card :	Marginally
Harddisk :	No

Like explained above Kribi is a pure software renderer, it means that all the computations required to generate pictures are done on the CPU(s). Faster CPUs show clearly better scores (unlike most « 3D benchmarks ») so KribiBench is a good CPU test. The Kribi engine use all CPUs present in a system and thus KribiBench is also an invaluable multi-processors PC benchmark.

Since Kribi use the graphic card merely as a framebuffer, KribiBench can’t be used to test all the capabilities of your graphic card. There is numerous benchmarks available for this purpose already. They are used routinely by computer hardware magazines and web sites and you should find them easily.

3. Installation

3.1 System Requirements

- Microsoft Windows XP / 2000 / ME or 98 operating system
- SSE (or compatible) enabled CPU like Celeron, Pentium III, Pentium III m, Pentium 4, Pentium 4 m, Xeon, Duron (new models only), mobile Athlon 4, Athlon XP, Athlon MP
- At least 20 MB of free hard disk space
- 256 MB system memory or more
- Graphic card with 1024 x 768 resolution and 24-bit colors or better

3.2 Install

To install KribiBench on your system :

- Exit all running applications (if any)
- Open *setup.exe*
- Follow the instructions from the installer dialog. For a standard installation simply click two times on the ***Next >*** button.
- Click on ***OK*** in the very last dialog and you are done with the install

After the installation KribiBench is available on the ***Start → Programs*** menu. You can then create a desktop shortcut as usual with the context menu ***Send To → Desktop***.

3.3 Uninstall

To completely remove KribiBench from your system do the following operations :

- From the ***Control Panel*** open the ***Add or Remove Programs*** utility
- Select KribiBench in the list
- Click on the ***Change/Remove*** button
- Confirm the deletion with the ***Yes*** button
- In the ***Remove Shared File ?*** dialog click on the ***Yes to All*** button and then ***Yes*** in the next dialog
- Click on ***OK*** in the very last dialog and you are done with the uninstall

3.4 Updates

Please note that when you update KribiBench it is recommended to uninstall the previous version before to install the new one.

4. Quick Start

4.1 Launching KribiBench

- Set your main display to 1024 x 768 / 24 bit colors or better
- Launch KribiBench with ***start*** → ***Programs*** → ***KribiBench***

4.2 Loading a 3D Model

- Click on the ***Open*** label at the upper left of the workspace
- Select a *.d* file in the dialog (the models are located in Local Disk\Program Files\Adept Development\KribiBench\3D models)

When the file is loaded a scene appears in the central window which we call the *view*. Its pane size is 640 x 480 pixels.

4.3 Information Area

Below the view at right you can found some basic informations about the current 3D model. You can see there how many polygons the model contains. Important data about the light sources in the scene are also available.

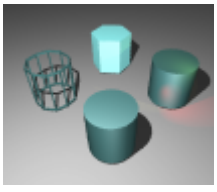
Some early users of KribiBench have asked us « what are those K, M, G letters » in the polygon counts. Their meaning is as follows :

K	**Kilo**	**1 000**
M	**Mega**	**1 000 000**
G	**Giga**	**1 000 000 000**

Examples :

12.72 K polygons :	12 720 polygons (more than 12 thousands)
3.45 M polygons :	3 450 000 polygons (more than 3 millions)
4.87 G polygons :	4 870 000 000 polygons (more than 4 billions)

4.4 Rendering Mode



The icon at the top left of the workspace allows you to select the rendering mode of the view. Simply click on the corresponding symbol to select a mode. The tool tips are here to help you. These modes are available :

- Wireframe.
- Flat Shading.
- Textured Shading. Textures, smooth surfaces, specular highlights,…
- Realistic Shading. Shadows casting, point and spot colored lights,…

5. Using KribiBench for Benchmark Tests

5.1 Running a Test

Using KribiBench as a benchmark tool is very simple. All the required commands are grouped together in the upper left corner of the workspace. When a model is loaded and the rendering mode selected, you just have to click on the ***Test*** label to run the benchmark test. A sequence of images is then rendered and the average frame rate is displayed at the end of the shot. The results in frames/sec are the official KribiBench scores, higher values mean better system performance.

5.2 Publishing Scores

It is recommended to run more than once the same benchmark test and to average the most representative values for published scores. If you publish results, please always mention the **name of the 3D model** the **rendering mode** used and the **version** of the Kribi engine. All these data are available in the information area below the view. A right mouse click in this area opens a context menu with edit features. It will help you to copy the desired content to the clipboard for further use in your favorite text editor.

6. Using KribiBench for Discovering the Kribi Engine

6.1 Overview of the 3D Models

The benchmark feature is also an easy way to discover the example 3D models, particularly the most complex ones. In the test sequences, the viewpoints are selected in order to show interesting parts of the models. To have an overview of Kribi capabilities, for each model loaded with the ***Open*** label, do the following operations :

- Select the ***Realistic*** rendering mode by clicking on the upper left icon
- Click on the ***Test*** label
- Watch the sequence of images

6.2 Playing with Kribi

Some early users of KribiBench have asked us if we use some tricks like pre-rendered images to achieve such a rendering speed, notably faster than the rendering packages they are accustomed to. To make it clear that it is not the case we have included commands that let the user interact with the 3D model. You are allowed for example to change the viewpoint, move or rotate the full model or only part of it and even to duplicate parts.

6.3 User Interface Philosophy

Most icons of the KribiBench user interface allow to select the current mode of interaction between the user and the view. We call the selected mode, the current **activity**. So it is typically a two steps process :

- You first click on the icon with the help of the tool tips, this will select the current activity.
- Then, you click in the view to actually use the activity.

For a given activity the user has two different usage models :

- **Clicks** : Left mouse button down and up at the same position. The operation achieved is dependant on the spot clicked.
- **Actions** : Left mouse button down (anywhere in the view) and move the mouse without releasing the button. You get this way a continuous update until the left mouse button goes up. Some actions are achieved with horizontal mouse movements, other with vertical movements and some use both directions. The action directions are detailed below with the **H**, **V** and **H+V** symbols for each activity.

6.4 Basic Activities

6.4.1 Introduction

Basic activities allow you to change the way a model is viewed. For example, you can rotate the full model (also called the **scene**) or move and rotate the virtual viewer who we call the **observer**. With these activities the model itself isn't modified and you can still use the benchmark feature after using them.

6.4.2 Panoramic Pictures

When you select some of the basic activities you have to wait a few seconds before to actually use it. A panoramic picture (a picture of the whole panorama at the current viewpoint) is computed in these cases and then you enjoy realtime update when you execute the action. The concerned activities are indicated with a **Panoramic** tag after their name.

6.4.3 Rotate Scene

Click : Reset the scene orientation

Action (H+V) : 2 axes rotation of the scene

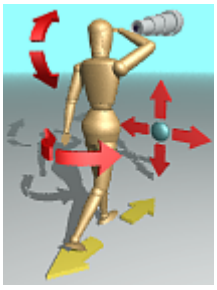
6.4.4 Rotate Scene Single Axis

Click : None

Action (H) : 1 axis scene rotation.

The actual axis is selected in the XYZ reference icon :

6.4.5 Move

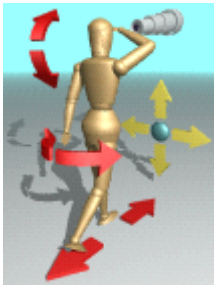


Click : The observer move in the direction of the spot clicked.

This is very useful to travel in complex models. You simply click in the 3D view where you want to go and the observer is moved near this place. With a few clicks you can come as close as you want to any detail of the scene.

Action (V) : Move the observer forward or backward

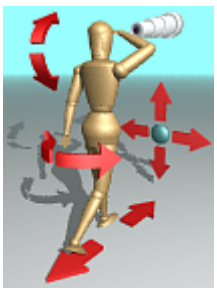
6.4.6 Pan



Click : The observer move in front of the spot clicked

Action (H + V) : Move the observer on a frontal plane

6.4.7 Zoom

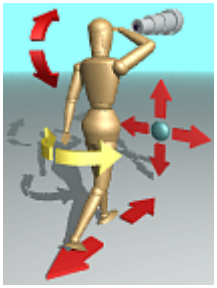


Click : None

Action (V) : Change the focal length

The scene appears bigger or smaller but the observer's position isn't modified. With small focal lengths, the perspective effect is accentuated.

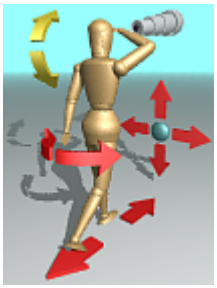
6.4.8 Turn Left / Right Panoramic



Click : The observer rotate horizontally to aim at the spot clicked

Action (H) : Rotate the observer horizontally

6.4.9 Look Down / Up Panoramic



Click : The observer rotate vertically to aim at the spot clicked

Action (V) : Rotate the observer vertically (elevation angle)

6.4.10 Look at Panoramic

Click : The observer rotate to aim at the spot clicked

Action (H + V) : 2 axes rotation of the observer

6.4.11 Tilt Panoramic

Click : Reset the tilt angle

Action (H) : Tilt the observer's head

6.4.12 Clip

Click : Set the frontal clipping plane at the spot clicked

Action (V) : Move the clipping plane forward or backward

6.5 Edit Activities

6.5.1 Introduction

Edit activities allow you to modify the model itself instead of only the way it is viewed like with previous activities. You can for example move or rotate independently some parts of the scene. We call **sub-objects** the independent parts of a given scene. A single sub-object is edited at a time. We call it the current sub-object or **current SO** in short. When a model is edited the benchmark results are altered, thus the benchmark scores are qualified as « non-official » in these cases. The benchmark feature is still useful with edited models though. Using edit activities and commands, it will allow you to discover Kribi very good scalability with more polygons and more light sources since you can watch precisely the evolution of the timings.

6.5.2 Discovering SO Activities

A good way to learn edit activities working on individual parts (SO) is to load the model *planes.d.* The parts that you can move independently in this model are :

- the floor
- the 5 planes
- the 2 light sources (white spheres)

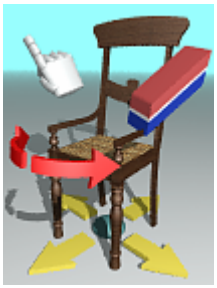
Provided that the current activity is one of the edit activities detailed below, when you click on one of these parts it is selected and the actual action (rotate, move, etc.) is then applied only on it when you move the mouse without releasing the button. You can keep this model loaded and have a quick try for every SO activity. Selecting the *Flat* rendering mode enable faster feedback. Some interesting effects are achieved in *Realistic* mode though. For example, when you move a light source or a plane, you see the shadows changing on the floor.

6.5.3 Rotate SO Default Axis

Click : Select the current SO

Action (H) : Rotate the SO around an axis that is object dependant

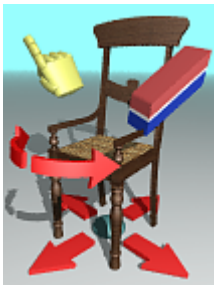
6.5.4 Move SO on Plane



Click : Select the current SO

Action (H+V) : Move the SO on a plane

6.5.5 SO Aim at



Click : The current SO is rotated to aim at the spot clicked.

Action : None

This one is particularly useful to specify the target of a spot light source. It is a bit more tricky to use than the other edit features because the current SO must be already selected prior to choose this activity. This example should clarify the usage :

- Set the Realistic rendering mode.
- Select the ***Move SO on Plane*** activity.
- Click on one light source. It is now the current SO.
- Select the ***SO Aim at*** activity.
- Click somewhere on the floor. The light source now point toward the spot clicked.

6.5.6 Rotate SO

Click : Select the current SO

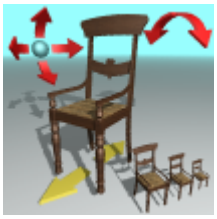
Action (H+V) : 2 axes rotation of the SO

6.5.7 Rotate SO Single Axis

Click : Select the current SO

Action (H) : 1 axis SO rotation.
The actual axis is selected in the XYZ reference icon

6.5.8 Move SO Single Axis



Click : Select the current SO

Action : Move the SO in one direction.
The actual axis is selected in the XYZ reference icon

6.5.9 SO Sizing

Click : Select the current SO

Action (H) : Change the size of the SO

6.6 Edit Commands

6.6.1 Introduction

Below the edit activities icons (at the bottom right of the workspace) you have acces to the edit commands. They are related to the current sub-object and thus a SO must be selected prior to use one of these commands. To select a SO simply click one part of a scene with a SO activity (like ***Move SO on Plane)*** set.

You must load the *planes.d* model for the examples below.

6.6.2 Copy SO

Create a new SO exactly like the current one but at a slightly different position. It allows you to create more complex scenes.

Example: Click on a plane with the ***Move SO on Plane*** activity set. It is now selected as the current SO. Click on the ***Copy SO*** label : a new plane appears and the polygons count goes up. You can now change the position of the duplicate: click on it and move the mouse without releasing the button until the wished position is reached.

6.6.3 Delete SO

Remove the current SO from the scene. It allows you to simplify the model.

Example: Click on a plane with the ***Move SO on Plane*** activity set. It is now selected as the current SO. Click on the ***Delete SO*** label : the plane disappears and the polygons count goes down.

6.7 Scene Graph Browser

A typical Kribi model is made of many objects. When a model is loaded, a default object is displayed in the view but it is possible to change later the focus to any other object thanks to the listbox below the view. The labels at right allow to display specific objects in this listbox. You then click on the object name in the list to use it as current scene.

- **Main** show the top-level objects of the model
- **Super** show the super-objects of the current scene (the objects using it as part)
- **Sub** show the sub-objects of the current scene (i.e. all its parts)
- **All** show all the objects of the model

6.8 Synchronization with Monitor

At the lower left of the workspace you find the***VSync*** toggle button. When it is pushed the display of the frames is synchronized with the refresh of your monitor. It helps to avoid diverse annoying artifacts. It is generally better to enable this mode when playing with KribiBench, particularly for watching panoramic pictures.

This feature provide less accurate timings when using the benchmark ***Test*** command and thus is disabled by default. Official KribiBench scores are obtained with VSync turned off.

6.9 Fullscreen Mode

The <F9> key let you enter the fullscreen mode. You have to hit <F9> again to go back to the normal mode. The fullscreen mode is provided only for playing with Kribi and not for benchmark purposes. The mouse isn't usable in fullscreen mode but KribiBench provide keyboard commands that let you travel into a model or watch panoramic pictures. These keys are usable in normal mode and full screen mode and are detailed in the next paragraph. It is generally better to set ***VSync*** ON for the fullscreen mode.

6.10 Keyboard Shortcuts

<F9>	Toggle between normal and fullscreen modes
<Up Arrow>	Move forward
<Down Arrow>	Move backward
<Left Arrow>	Turn Left
<Right Arrow>	Turn Right
<Delete>	Look down
<End>	Center view
<Page Down>	Look up
<Shift>	Temporary acceleration when moving or turning

<Space>	Temporary deceleration when moving or turning
<+> (Num Keypad)	Acceleration (moving only)
<-> (Num Keypad)	Deceleration (moving only)

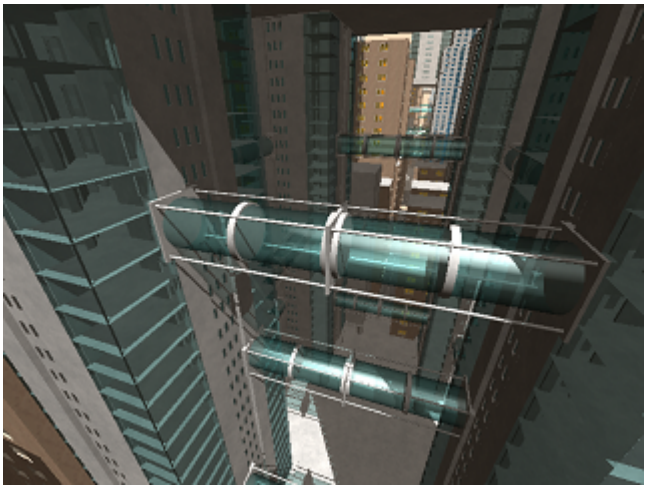
7. 3D Models

7.1 Introduction

The 3D models included in this release of KribiBench are the official ones to use for publishing scores. We have picked a dozen files among the huge collection of benchmark-ready models that we actually use in the labs for further progress of the Kribi engine. Some are more real-world like and others are typical « labs scenes » featuring airplanes with transparent missiles and such…

KribiBench purpose is twofold: providing an invaluable CPU benchmark and let its users discover the strength of the Kribi engine. We have thus selected models showing a wide range of the current Kribi capabilities. Every benchmark-ready model put more emphasis on some features of the Kribi engine. For each example model you will find below a short indication of the area of the renderer particularly involved.

7.2 City



- Deep Scene Graph
- Transparent Materials
- Complex Shadows

This modern city is our most ambitious model to date. It was modeled entirely with our own tools. The transparent tubular footbridges are particularly challenging for a renderer because they cast colored shadows on the buildings walls and windows.

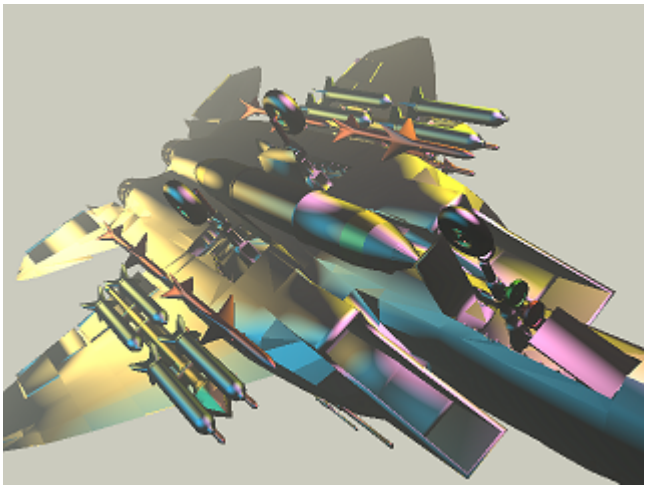
With the ***Move*** activity you can visit the city by clicking on the interesting places. When you like a viewpoint you can select the ***Look at*** activity to enjoy a realtime panorama.

7.3 Office

- Realworld Content
- Features-rich Lighting

This interior scene is an excerpt from one of our commercial packs. It shows complex shadows casting with many point and spot lights. Like the city it is a nice model for playing with the panoramic pictures feature.

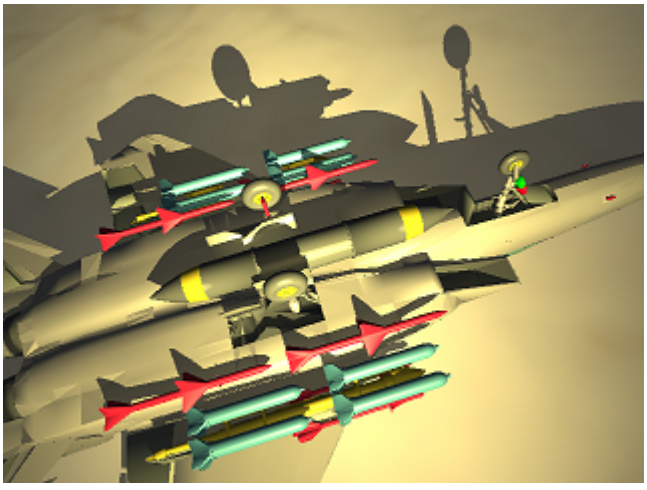
7.4 JetFog



- Smooth Surfaces
- Specular Materials
- Colored Lights
- Fog

A simple scene with 4 colored lights in realistic shading. Metal-like material for the plane and high quality smoothing.

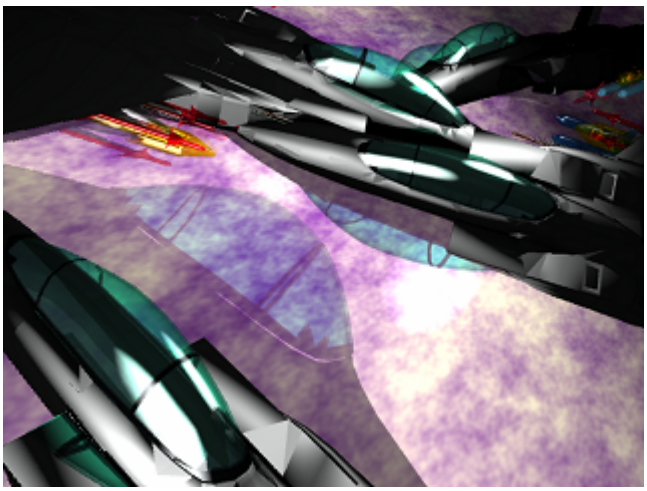
7.5 JetShadow



- Smooth Surfaces
- Specular Materials
- Shadows

A variant of *jet* with a single spot light and shadows casting. Shows realistic shadows at interactive frame rate.

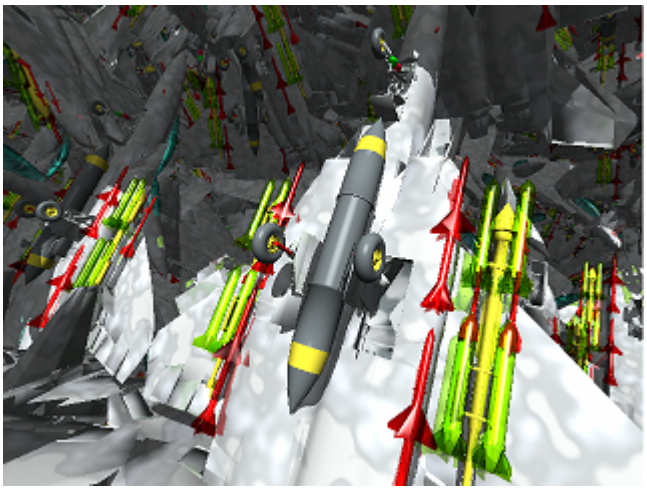
7.6 Planes



- Procedural Textures
- Transparent Materials
- Complex Shadows

Shows projected transparencies in realistic rendering and a procedural texture on the floor.

7.7 Squadron



- High Depth Complexity
- Transparent Materials
- Procedural Textures
- Complex Shadows

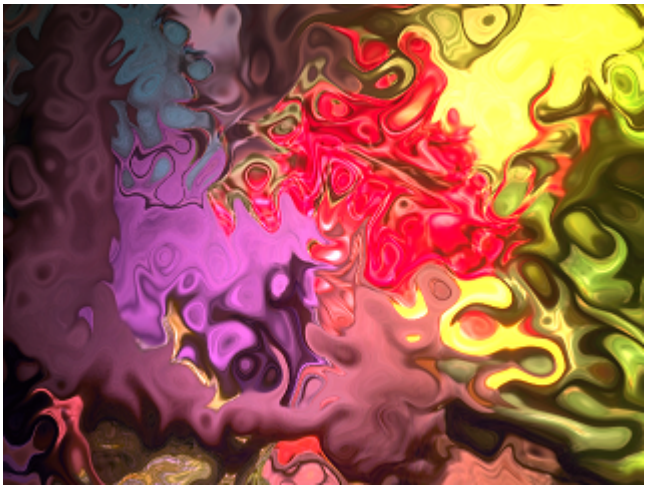
Another typical labs scene : a heap of airplanes ! It's an example of very high depth complexity scene. The complex mix of opaque and transparent surfaces allow us to stress and improve the renderer.

7.8 Base

- Simple Texture Mapping
- Colored Lights

Although classical texture mapping is a simple part of the full rendering problem, it is important to optimize its speed like for other areas. This simple scene with a clipped texture is one of our models of choice for tuning Kribi texture mapping code.

7.9 BaseTurbulent



- Complex Texture Mapping
- Colored Lights

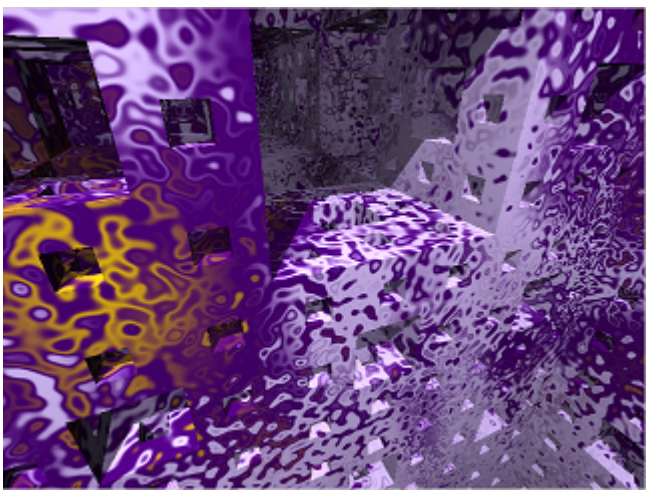
This model is a variation of the previous one with a more complex texture mapping scheme. Here the texture is warped according to a 3D noise function. This effect is also called *procedural turbulence*.

7.10 TubesMap

- Cylindrical Texture Mapping
- Colored Lights
- Specular Materials

Here is another important case of texture mapping : the planar source photo is mapped onto cylinders.

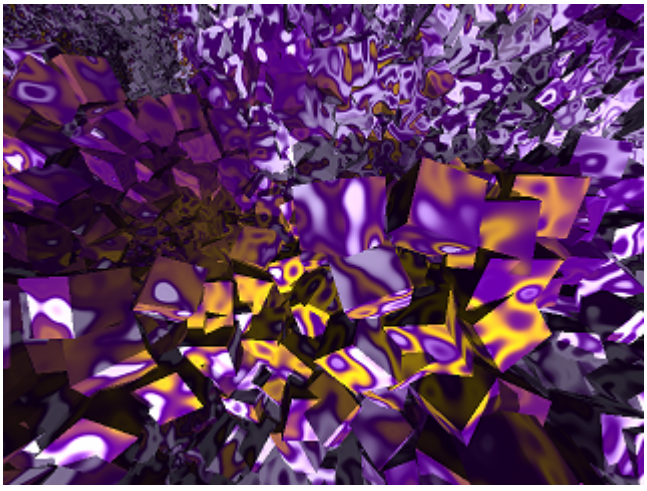
7.11 Sponge



- High Depth Complexity
- Procedural Textures

Computer generated fractal made of cubes. You can see the continuity between surfaces achieved with the use of a 3D procedural texture.

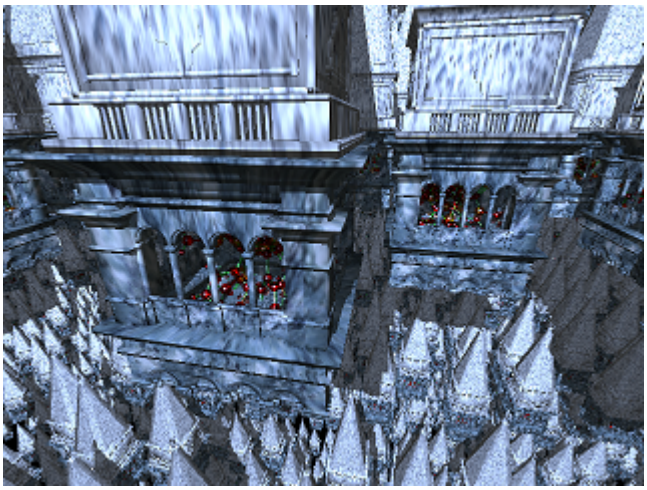
7.12 SpongeExplo



- High Depth Complexity
- Procedural Textures

A variation on *Sponge*. Accessing the low level objects with the scene graph browser the scene has been « exploded » with the move & rotate SO activities. Note that there is still a perfect aspect continuity between surfaces thanks to the 3D texture in use.

7.13 Ultra



- High Polygons Count
- High Depth Complexity
- Spherical Texture Mapping

Ultra stands for « ultra complex »…

For this scene we have modeled the top of a classical tower from our famous neighboring city *Venice*. It's also a typical labs scene: we have included in each monument a transparent molecule. The historic building is duplicated more than 150'000 times in the model for a grand total of more than 16 billions polygons !

In realistic shading the "moonlight shadows" effect is achieved with a central point light source. It stress the Kribi rendering engine since the light source cast shadows in every directions.

8. Acronyms

CPU **C**entral **P**rocessing **U**nit. Also known as *processor*. This is an essential part of your computer that do all the actual computations required by Kribi to generate pictures. Examples of well known modern CPUs : Pentium 4 and Athlon XP.

SO **S**ub-**O**bject. A part of an object. We call the whole object the *scene* and the currently selected sub-objet the *SO.*