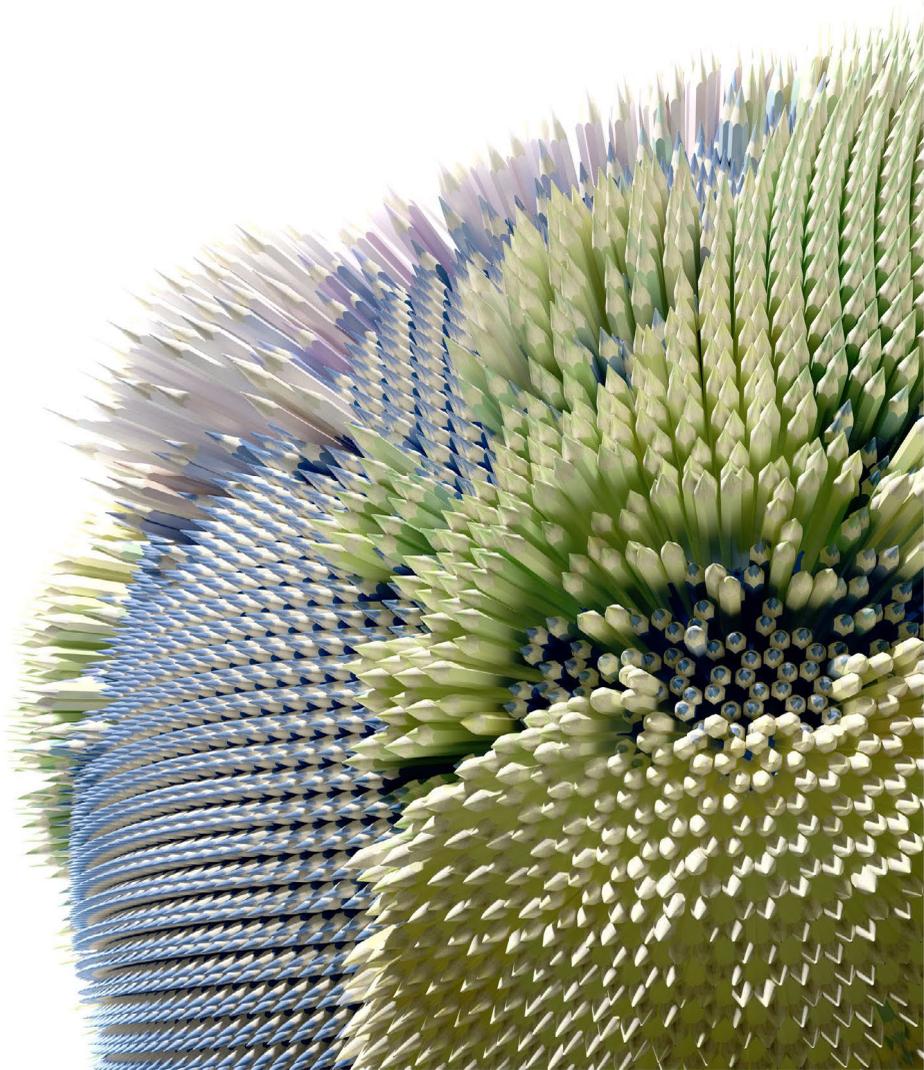


André Kutscherauer

# 3D 3ds Max tool ScatterFX







[www.ak3d.de](http://www.ak3d.de)



**André Kutscherauer**

**ScatterFX**  
3ds Max tool

[www.ak3d.de](http://www.ak3d.de)



## AK3D® ScatterFX

Published by  
André Kutscherauer (AK3D)  
Bartholomäer Strasse 4

81825 Munich / Germany  
<http://www.ak3d.de>

Copyright © 2011 André Kutscherauer (AK3D®). All rights reserved.

### Copyright:

Proprietary Rights. The Software and Documentation (= the work) are protected by copyright, trade secret and other proprietary rights of AK3D® (= publisher, software producer, author).

Limitations on Use. The right to use Documentation is contingent upon the right to use the applicable Software. AK3D® reserves all rights in the Software and Documentation not expressly granted to Customer. AK3D® grants a **1-user per-seat license** to Customer. Customer is permitted to make a reasonable number of copies of the Software and the Documentation for disaster recovery and back-up purposes but not for use of simultaneous users accessing the program. Unless otherwise stated Customer will not:

(a) reproduce the Software or Documentation;

(b) market, license, distribute, sublicense or otherwise commercially exploit the Software or Documentation or sell, lend, rent, give, assign or otherwise transfer or dispose of the Software or Documentation;

(c) permit the use of the Software or Documentation by others or otherwise operate the Software for third parties (e.g., as a service bureau or data processing service);

(d) modify or translate the Software or Documentation into any other computer or human language;

or (e) disassemble, reverse engineer or decompile the Software or otherwise attempt to discover any portion of the object or source code or trade secrets related to the Software.

No part of the Software or Documentation may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of AK3D® with the exception that the electronic formats may be entered, stored, and executed in a computer system for your private and/or business personal use, but they may not be reproduced for publication. Requests to AK3D® should be addressed to: [scatterfx@ak3d.de](mailto:scatterfx@ak3d.de)

### **Limit of Liability/Disclaimer of warranty:**

AK3D® makes no representations of warranties with respect to the accuracy or completeness of the contents of the Software or Documentation and specifically disclaims all warranties, including without limitation warranties of fitness for a particular purpose; of integrity, interface and/or structure of the Software and the use specifically concerning the operation in conjunction with computer programs, data or hardware supplied by a third party.

No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is licensed with the understanding that AK3D® is not engaged in rendering legal, accounting, or other professional services. If you require legal, accounting, or other expert assistance, you should consult a professional advisor. AK3D® shall not be liable for this work as a citation and/or a potential source of further information does not mean that the AK3D® endorses the information the organization or website may provide or recommendations it may take.

AK3D® makes no express or implied warranty of any kind and assumes no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or digital files contained or supplied with or for the use with this work.

Further, readers should be aware that internet websites listed in this work may have changed or disappeared between when this work was written and when it was published and read.

For any suggestions and information on errors AK3D® is grateful. These should be addressed to: [scatterfx@ak3d.de](mailto:scatterfx@ak3d.de)

### **Trademarks and other intellectual property:**

AK3D and the AK3D logo are the firm name and firm logo with AK3D® is a registered German Trademark of André Kutscherauer and protected by Commercial Law, Competition Law and Trademark Law in Germany and other countries, and may not be used without the prior written permission with the exception of private use or citation. “3ds Max” and all other brands, brand images and product names mentioned in this work are trademarks, service marks or intellectual property of their respective companies. The same applies to design rights and patents mentioned in this work. If mentioned without using signs like ® or ™ you can not be sure that they are free of rights of third parties. AK3D® is not associated and has no business relationship with any of those products or vendors mentioned in this work.

### **Privacy Policy:**

Any of the information collected from you may be used to personalize your item of the work by embedding your contact data and MAC-address, CPU-ID in the work file. Your information, whether public or private, will not be sold, exchanged, transferred, or given to any other company for any reason whatsoever, without your consent, other than for the express purpose of delivering the purchased product requested.

**Notice:**

AK3D® also publishes a variety of other products in print and in a variety of electronic formats. Some contents that appear in print may not be available in electronic formats. In case AK3D® published online supplementary information (eg tutorials), the above mentioned rules also apply. AK3D® does not assume any obligation to provide additional information.

## TABLE OF CONTENT

<b>table of content .....</b>	<b>6</b>
<b>1 Introduction .....</b>	<b>9</b>
<b>1.1 Preamble .....</b>	<b>10</b>
<b>1.2 Usages.....</b>	<b>11</b>
1.2.1 Architectural Purposes .....	11
1.2.2 Bringing live to city design or architectural projects ...	11
1.2.3 Marketing Projects.....	11
1.2.4 Free Art Projects.....	11
<b>1.3 Key Features: .....</b>	<b>12</b>
<b>1.4 Detailed Features:.....</b>	<b>13</b>
1.4.1 What you get.....	13
<b>2 Installation.....</b>	<b>15</b>
<b>2.1 Installation .....</b>	<b>16</b>
2.1.1 Running the installer .....	16
2.1.2 Place ScatterFX in your Toolbar.....	16
2.1.3 Activation process.....	17
<b>3 The User Interface .....</b>	<b>19</b>
<b>3.1 The rollouts .....</b>	<b>20</b>
3.1.1 Baseparameters .....	20
3.1.2 Distribution .....	21
3.1.3 Orientation rollout.....	22
3.1.4 Direction rollout .....	23
3.1.5 Modification rollout .....	24
3.1.6 Animation rollout.....	25
<b>3.2 The maps .....</b>	<b>26</b>
3.2.1 Scattermap.....	26
3.2.2 Densitymap .....	26
3.2.3 Highmap.....	26

---

---

3.2.4	Orientationmap .....	26
3.2.5	Animationmap .....	26
3.2.6	Objectmap.....	26
3.2.7	Colormap .....	26
3.2.8	Offsetmap.....	26
3.2.9	Headmap.....	26
<b>4</b>	<b>Quickstart Guide.....</b>	<b>27</b>
4.1.1	Basefunctions .....	28
<b>5</b>	<b>About the author .....</b>	<b>45</b>

---



# 1

# INTRODUCTION

## 1.1 Preamble

ScatterFX is Plug-in for 3dsMax written in Maxscript. It is written from a creative for creative's. It is a program to the place a large number of objects with maximum aesthetic impact. Basically it is a highly advanced scatter tool with functions to create huge amounts of copies from one object and scatter them around a "Baseobject". How these objects are scattered is fully controllable via different maps. The scatter objects can be placed on triangle, vertices or "Texels" on a Baseobject with any shape. Next to the endless usages of scattering objects to a base object controlled by maps, the unique biped feature will let your architectural projects come to a new level of "natural" look. Please send in your result images or animations. The main highlight of the program is the ability to place objects controlled by. This allows results that are currently not possible with base tools. Objects can be placed controlled by maps. Here, not the network structure of the base object is used as placement goal, but its texture coordinates. This allows very interesting possibilities. A second highlight of the program is the ability to place not only static objects, but to use also fully animated bipeds for placement. This allows a much more intuitive way to place a large number of people in a room, the bearing of a texture can be controlled. This is much easier and faster than, for example, a solution with the simulation of a crowd. Thus, it is the ideal complement for architectural applications to fill visualizations with life. It is a creative addition to the tools of 3D Studio Max, and can be used in very different situations. As the implementation of the scheme using Maxscript, the maximum compatibility to 3D Studio Max is guaranteed. Even with program updates. The program is completely designed from user's perspective and guarantees thus maximum intuitive operation. The full program is modular, so that all functions can be combined with each other. It allows you as a very flexible implementation of scatter effects. This documentation covers essentially the basic functionality. In addition, are but on the number of possible combinations applications possible can be that not all documented. The program stores all settings in the objects themselves, so that a backup of the settings is not necessary.

I will post every entry that was made with ScatterFX on [ak3d.de](http://ak3d.de)!

---

## 1.2 Usages

### 1.2.1 Architectural Purposes

Example: Scattering different Trees around a landscape with maximum control. You can setup maps for color changes, density maps realize areas without trees and setup a map for a high range where trees should be placed. Even animated Trees are possible. What about placing thousands of cars of different color and type on a parking house with absolute control what car will be placed where, and in what color...

### 1.2.2 Bringing live to city design or architectural projects

Example: The function of being able to scatter animated bipeds, also controlled by maps, is an absolute unique feature of ScatterFX. Furthermore you can setup a "Look At" Direction and separately can control where those entire bipeds should look at, while their body's still follows their existing animation from their heads. With ScatterFX you will be able to fill out a complete station with an animation of just one standing up biped, where every single biped will have a different pose, controlled by a map. Then we can setup, where all the bipeds should look at, combined and mix this with a second target. Finally we can setup, that some of the bipeds stands up and hold a banner in their hands, building together a text like on Olympia. Crazy, isn't it?

### 1.2.3 Marketing Projects

Example: What about taking thousand copies of one product to scatter them around a second object? All those copies can have different colors. So can easily show a second object just through thousands colored copies of your main product in different colors...

### 1.2.4 Free Art Projects

Example: Scatter 20000 pens around a simple globe sphere with two different maps. Use the Color map to colorize the pens like a real globe

---

---

and use a black and white High map profile of the earth to alter the Z Position of the pens to visualize the "real" high profile of earth...

### 1.3 Key Features:

- Easy to use
  - 10 maps to control the Scattering in detail:
  - Scattermap / Densitymap / Highmap / Orientationmap / Animationmap / Objectmap
  - Colormap / Offsetmap / Headmap / Headmap Weightmap
  - Use Bitmaps, procedurals or even mix maps for input
  - All maps can be modified by adding random numbers in ranges
  - Automatic UV channel number handling
  - shade thousands of objects with just one shader
  - Use of the new "Object Color" shader of mental ray.
  - Easy "Show Map in Viewport" buttons for knowing how every single map exactly is looking
  - UI with focus on creative parts
  - Every "module" setting is combinable with the other
  - Endless combinations are possible
  - Randomness addition to every setting for natural results
  - Special Undo / Scatter Function for quickly prove and fine-tune scattering
  - All settings stored in Base object so re-scattering is easily possible
  - As it is realized with Maxscript, it will work in all future 3ds max releases without needing to update it
-

## 1.4 Detailed Features:

- Map Controlled Scatter Tool with an extremely intuitive Interface
- Complete modular Structure for endless combinations of every feature
- Automatic procedurals rendering while scattering
- Stores any setting directly into the Base object. So even if no Objects are existent in the Scene, the Base Object will "know" the last settings you have made and can easily reproduce the scattering
- ScatterFX will take the color from the Color map and will setup the wire color for every single copy. The "Object Color" shader will be able to render this color out.
- Copy types "Instance", "Reference" or "Copy"
- Performance focused options like to show only a percentage of the final objects while the others will be on a hidden but render able layer
- Even more performance by "Show as box" feature for the results
- No need for maps, all values can be entered as number, too

### 1.4.1 What you get

- ScatterFX one seat license
  - Installation movie
  - PDF Documentation
  - Getting Started Guide (within the Documentation)
  - Tutorial support files for a quick start with ScatterFX
  - Support forum on [ak3d.de](http://ak3d.de)
-



# 2

## INSTALLATION

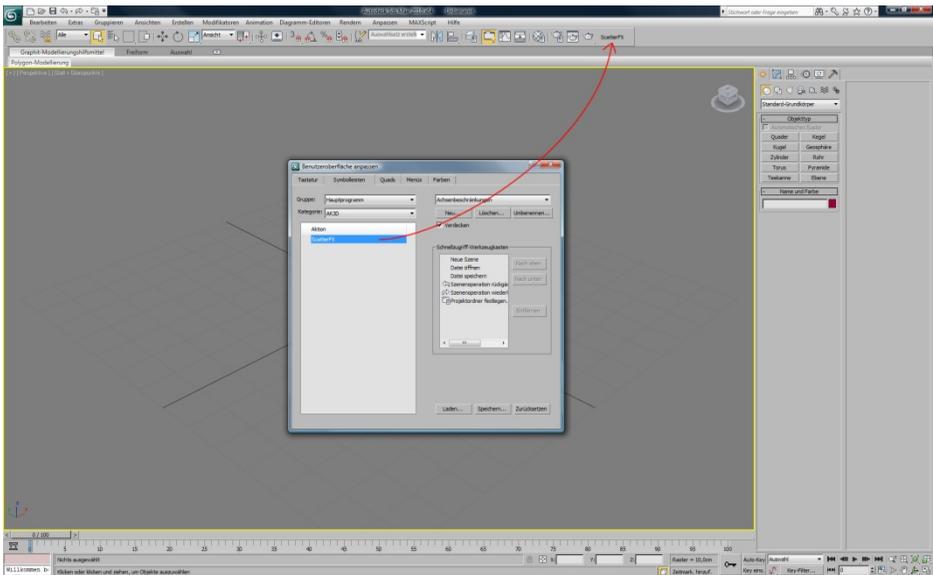
## 2.1 Installation

### 2.1.1 Running the installer

After receiving your download link, start your 3ds max. Go to “MaxScript” – “Run Script” and search for the file you have downloaded “AK3D\_ScatterFX\_Install.mzp”.

### 2.1.2 Place ScatterFX in your Toolbar

Go to “Customize – “Toolbars” and search for “AK3D” in the list. Here you will find ScatterFX. You can drag and drop it to a place in your toolbar where you like.



### 2.1.3 Activation process

After the first start of ScatterFX a License Activation Request have to be made. Please just fill out the text fields and click on “Send Activation Request”. I will review it as fast as possible and will send you your Activation Key back. For this step it’s necessary to enter a valid order number and valid information’s, that I can allocate your order. (Normally within the next two hours)

If you ever stuck at any step, please visit [www.ak3d.de](http://www.ak3d.de) as there will be a forum where help for ScatterFX may be found.

---



# 3

## THE USER INTERFACE

## 3.1 The rollouts

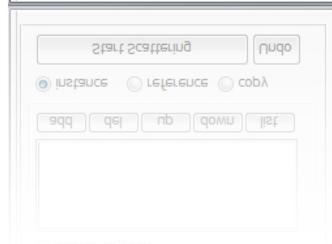
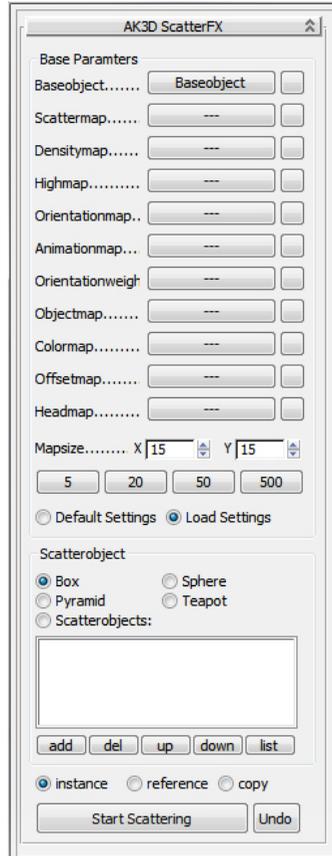
### 3.1.1 Baseparameters

This is the main rollout with the most important input. Here we can select a “Baseobject”, setup the control maps, the resolution and the objects we like to scatter around the Baseobject.

With a click on the button next to “Baseobject” we can select the object where we want to scatter around the objects. If the object is selected it ScatterFX will apply a new material to it, where all maps and settings are stored to.

With a click on the buttons next to the map names we can setup the maps that control the scatter process. With the right square buttons we can easily switch to the map that should be showed in the viewport. With the option “Mapsize” we can choose the “resolution” of the Scattering. This is basically the count of objects that will be created. Some count presets are reachable via the four buttons. In this category there is also a list field included, with that we can add one or even more objects to scatter as well as some primitives. We can also choose the kind of copy here. “Instance” is most useful for performance if there will be a lot of objects.

Finally with the button “Start Scattering” and “Undo” we can start the Scatter process or “Undo” it for changing some settings.



### 3.1.2 Distribution

This rollout is structured into five groups. In the “Basic Settings” we can setup where the objects should be placed.

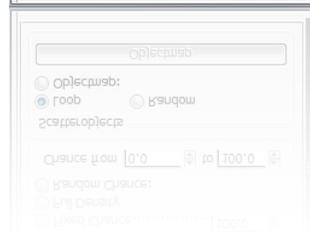
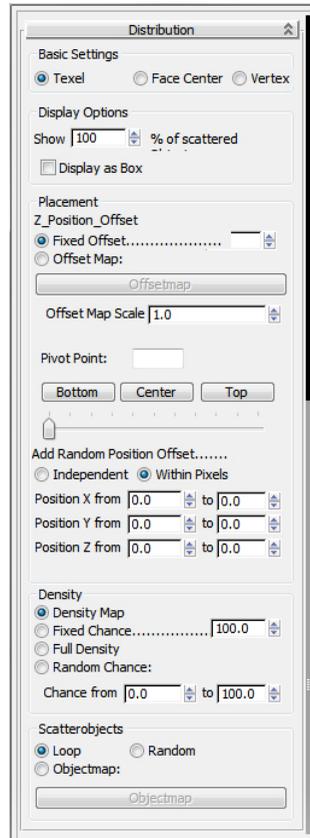
With “Texel, objects will be placed at the underlying pixel of the texture, with “Face Center” at the center of the triangle, with “Vertex” on every Point of the Baseobject. In “Display Options” we can how many of the resulting objects should be shown in present. The rest will be placed on a hidden layer that will be renderable. So we can keep viewport performance.

The “Display as Box” option furthermore raises viewport performance if needed. With the group “Placement” we can setup an fixed offset in Z or setup a “Offsetmap”. The scale parameter is a multiplier to raise or lower the effect of the map.

With the Pivot point options we can alter the rotation an position point of the objects. With the random Pos Offset we can add a “more natural” random placement factor.

The group “Density” contains four options to control the Density. Her we also can control it via a Density Map, a fixed Chance factor or a range.

With the options of the group “Scatterobjects” we can change the order of the objects, if we have setup more than one. With loop they will be place in the order of the list, with random they will placed randomly, with an Objectmap, the Brightness controls the order.

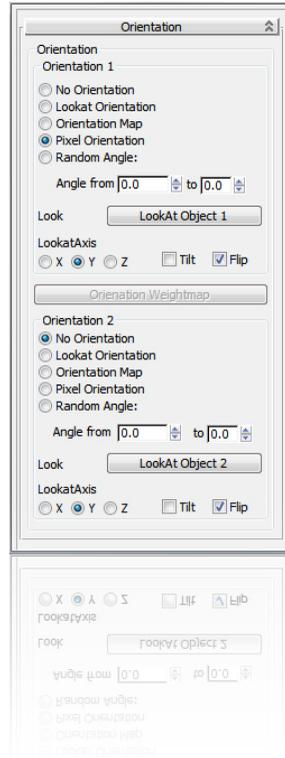


### 3.1.3 Orientation rollout

In that rollout we can setup up to two different Orientations for the scattered objects.

With the standard option “No Orientation” the objects will keep there initial Orientation. With the option “Lookat Orientation” we can select a Lookat object with a click on the button next to “Look”. With the options in “LookatAxis” we can control what axis should lookat the object and we can chose “flip” it it’s the opposite direction. The option “tilt” even alters the Z axis of the objects to look at the target object. Without “tilt” the objects are only rotated around their Z axis. With the option “Orientationmap” the Orientation is map controlled (black 0 degrees, white 360 degrees). With the option “Pixel Orientation” we can let the objects look in the pixel direction (when mapping is turned on or so) With random angle and the spinners we can manually setup the “Range of randomness” of the Orientation. 0 and 360 means here “fully random”, 0 and 0 would mean 0 and so on.

Finally the exact same settings are reachable for a second orientation. We can easily mix these two Orientations via an “Orientation Weightmap”. White is Orientation 1, black is Orientation 2, Grey values are mixes of both.

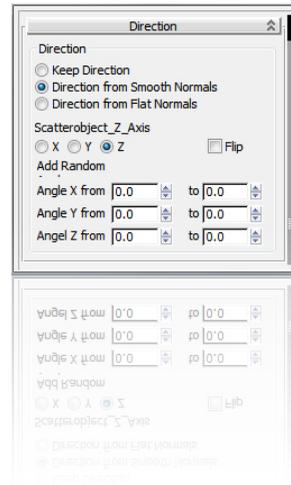


### 3.1.4 Direction rollout

In that rollout we can setup up the “Z” direction of the scattered objects. If you take a sphere as base object you can fast see what these options do.

With “Keep Direction” objects will keep their initial Z direction and will not follow the surface normal orientation. With “Direction from Smooth Normals”, the Z axis will be placed perpendicular to the Baseobjects surfaces. Even Smoothing groups will be calculated properly with this option. Objects will be placed smoothly around the baseobject while using the smoothgroup of it for orientation. With the option “Direction from Flat Normals” only one normal of one polygon of the baseobject will be used. So to get a smooth scattering around a sphere, you would need a lot of more objects with this option as with the “Smooth normals” option. In the group “Scatterobject\_Z\_Axis” we can chose Axis of the object should be used for the “Z\_Axis”. With “Flip” we can turn the object around 180 degrees if it was the opposite direction. Basically we can turn the objects with that option.

Finally with the “Add Random” spinners, we can add random values for each axis. “0” and “0” here means add no random values. The most random value for example would be “0” to “360” but we can also setup just a range “from” “to”. This is very useful to get a more “natural” look into the result.



### 3.1.5 Modification rollout

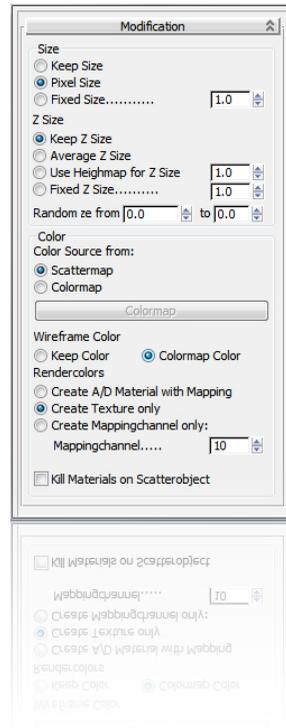
In that rollout we can take influence of the Scaling and Coloring of the scattered objects.

In the group “Size” we can choose between three modis: “Keep Size” takes the original scale of the objects and doesn’t alter it. The setting “Pixel Size” scales the object to fit into the current mapped pixel size as good as possible. This works best with rectangular objects and unscratched mappings. It’s the algorithm for creating “3D Pixelblocks”. “Fixed Size” is just basic “Multiplier”.

In the group “Z Size” we have control over the height of the object separately. “Keep Z Size” keeps the original Height of the Object. “Average Z Size” Takes the modification of the “X” and “Y” scale of the object and sets the “Z” height to the average of the values. With “Use Heighmap for Z size” we can use the “Heighmap” for controlling the Height. Black is “0”, white is “100”. With the factor on the right, we can multiply this with a number. “Fixed Z Size” is a fixed multplicator. With “Randomize” we can setup a random value that makes the Height changes more “natural”.

In the group “Color” we can setup where the object get their color from. With “Color source” we can chose to take the color form the “Scattermap” or from a second “Colormap”. In “Wireframe Color” we can set “keep color”, the colors will not be changed or “Colormap Color” to colorize the wireframes.

In the group “Rendercolors” we can setup what to do with the colors. “Create A/D Material with Mapping” renders out a bitmap, creates a material with that bitmap as diffuse. With “Create Texture only” we just will render out a texture. This is the fastest option. With “Create Mapping channel only” we can create a special Channel for the objects. “Kill Materials” deletes all applied materials on the scatter objects.



### 3.1.6 Animation rollout

In that rollout we can setup how to deal with object that are animated and how to deal if the animated objects are bipeds. We even can give up to three “requisites” to the figures and can control on what frames they should be visible. This way we easily can make a station with figures that Holds up banners etc...

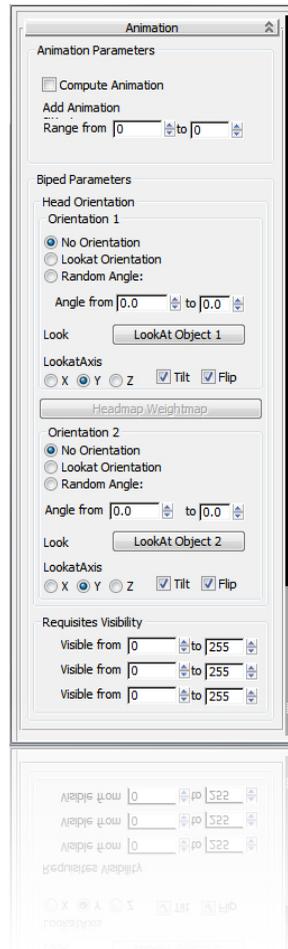
The first checkbox en or disables the computation of the animation. The computation of an animation means: On a black pixel, we see the first frame of the animation of that object and on a white Pixel, we see the last frame of that animation. With an animation map we can exactly setup on what pixel we want to see what frame of the base object.

With the spinners we even can add a “randomness” to the resulting frames of the animation. This brings more “natural” mass movement feeling.

In the group “Biped Parameters” we can setup how to deal with animated bipeds. The group “Head Orientation” works exactly like in the rollout “Orientation” except that it only alters the bipeds head position in this case. We can choose up to two “lookAt” objects and mix the together via a map.

In the group “Requisites Visibility” we can exactly setup on what animation frame of the single objects when the requisite should be visible.

With these settings, we will be able to fill out a complete stadion with an animation of a standing up biped, where every single biped will have a different pose, controlled by a map. Then we can setup where all the bipeds should look at, combined and mixed with a second target. Finally we can setup that some of the bipeds stands up and hold a banner in their hands, building together a text like on Olympia. Crazy, isn't it?



## 3.2 The maps

### 3.2.1 Scattermap

Controls the basic positions, if ScatterFX is in “Texel” mode. The map, we setup here will be rendered in the size we setup with “Mapsize”. Objects will then be placed on every projected position of that map.

### 3.2.2 Densitymap

White means “100 % chance” of placing. Black means “0% chance” of placing. All grey values will raise or lower the chance that an object will be there.

### 3.2.3 Highmap

The Highmap alters the “Z-Position” or “Z-Scale” of the objects.

### 3.2.4 Orientationmap

Controls the Z rotation. White means “360 degrees”, Black means “0 degrees”

### 3.2.5 Animationmap

Controls the animation frame to be scattered. “White” is the “first frame”, “Black” is the last frame.

### 3.2.6 Objectmap

Controls the order in what objects from the list will be scattered. (Black to white – Top Down in the list)

### 3.2.7 Colormap

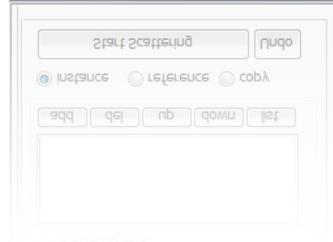
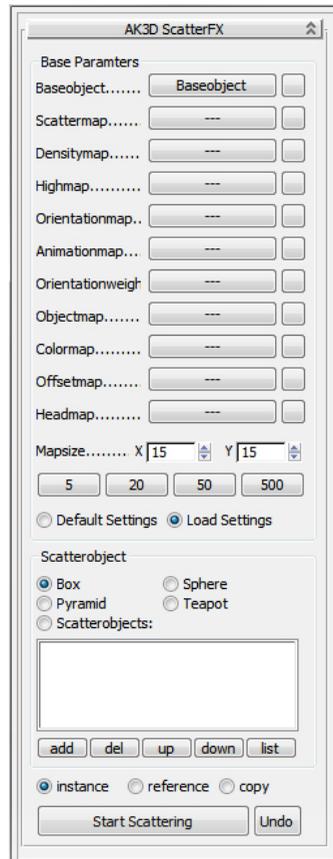
Separate Map for the color if Color from Scattermap should not be used.

### 3.2.8 Offsetmap

Alters also the “Z-Position”. If Heightmap should control just “Z-scaling”.

### 3.2.9 Headmap

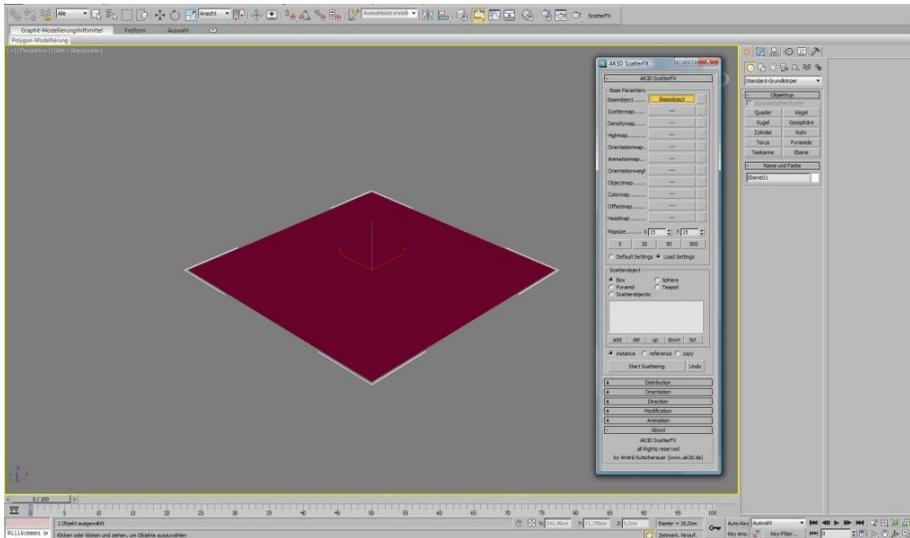
Like the Orientation Map but just for biped heads.



# 4

# QUICKSTART GUIDE

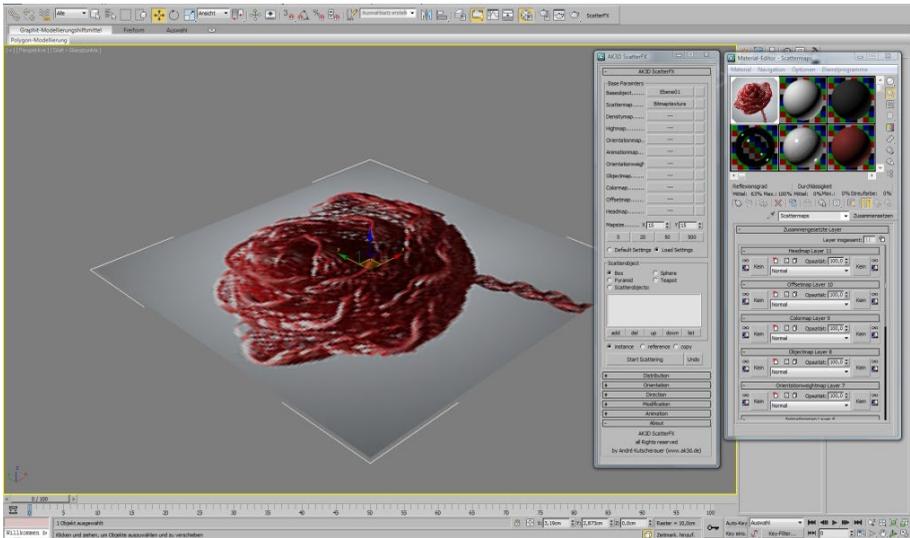
### 4.1.1 Basefunctions



Start 3ds Max and create a new plane with 1 x 1 length / width segments and remove the check mark at "Original size of the map" and hide the grid with the Shortcut "g". Click on ScatterFX. Then open the main dialog of the program.

First, click on the button “Baseobject” and then click on the plane in the viewport. The caption of the button changes to the name of the selected object. The object changes the color.

Then click on the button next to "Scattermap...". The standard mapdialog opens. In this dialog you can select every supported map types. In addition to parametric maps like "Gradient" or similar, you can use also 3d maps or bitmaps. Choose "Bitmap" and browse to the included thumbnail of “Rose\_of\_Ring.jpg”.

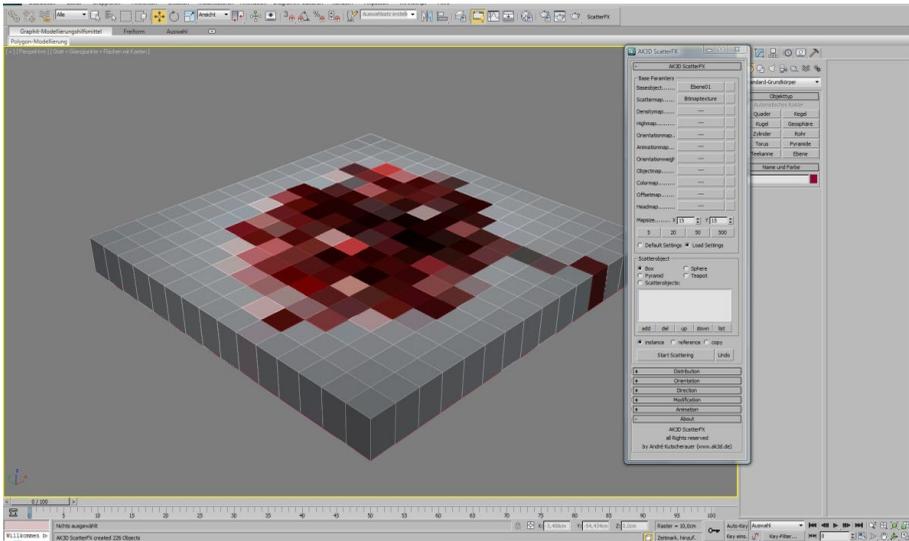


After you select the bitmap, the image is immediately displayed in the viewport and the material editor is automatically: note that the Baseobject got assigned a new material automatically, where all maps of ScatterFX will be stored. This material consists of a "Compositing" map. The slots are titled with the respective names in the dialog of ScatterFX. If you scroll down to the slot "Scattermap" you will see that the map that you have selected in the step before is assigned to this material, too.

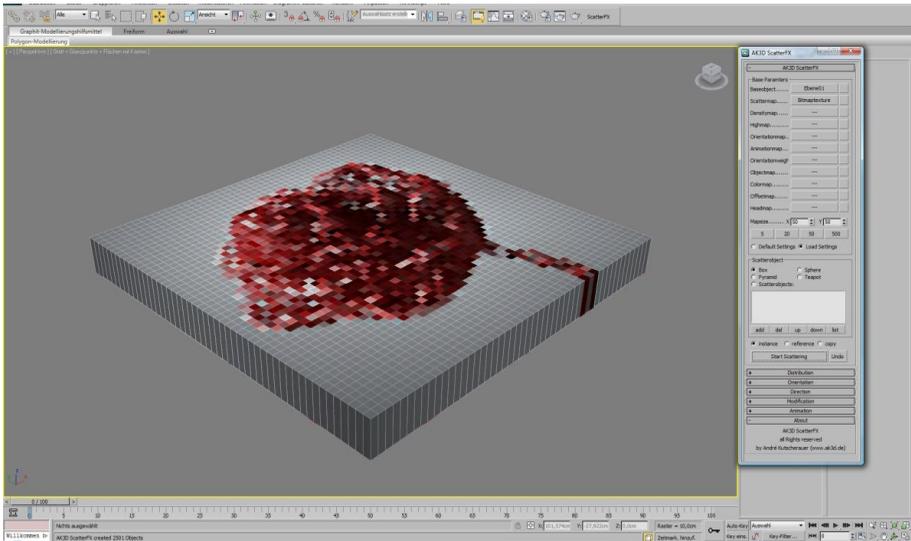
This was already the minimum preparation so that the scattering can work. Note that these two steps are always necessary to start:

1. Select one object with mapping coordinates as "Baseobject"
2. Choose a Scattermap (bitmap, 2D or 3D maps)

Now we click on "Start Scattering".



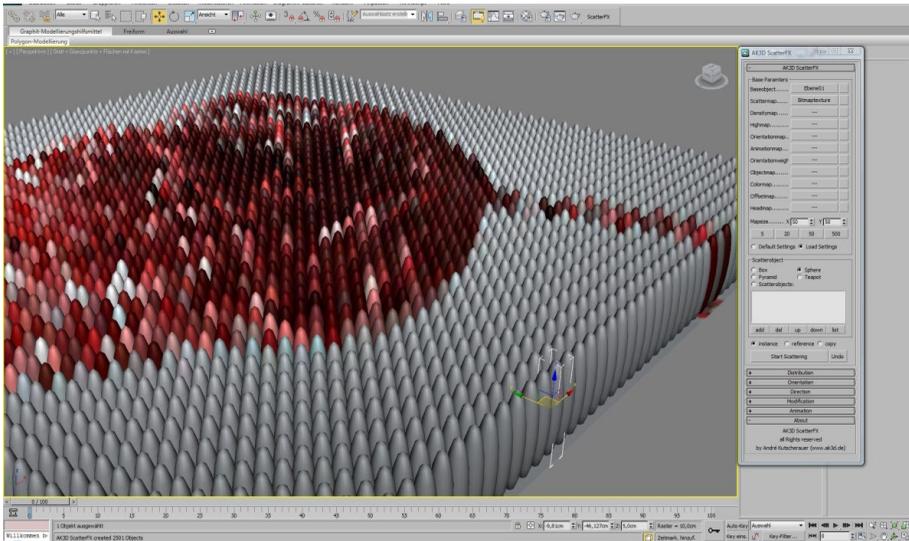
In accordance with the basic settings, the program has generated the 15 x 15 boxes. The boxes have been scaled to pixel size (more on that later) and the color of the texture is assigned to the boxes as "Wirecolor". All objects are instances and thus optimally in terms of memory consumption. The "Undo" button will always restore the State just before the start of the last Scattering. This function is very useful to try out different values. Now we click on "Undo".



Then we chose another Preset for the “Mapsize”. For example “50”. Mapsize X and Mapsize Y values are automatically set to 50. Click on “Start Scattering”. The result reflects the image much more accurately, as now are 2500 objects scattered instead of 225 before. We got true “3D pixels” here.

We click again on "Undo" and select the button "Sphere" in the "Scatterobjects" group.

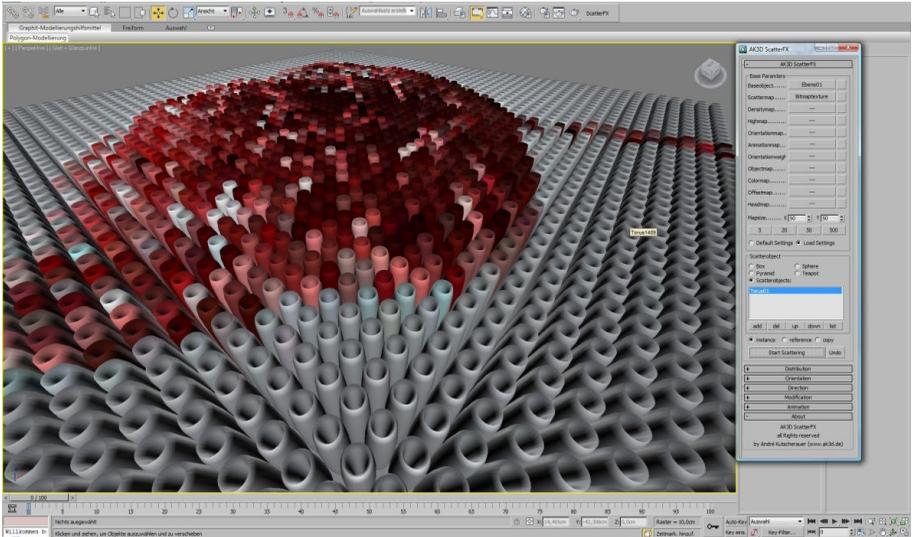
Then click "Start Scattering"



You have created 2500 spheres, colored with their underlying texture, where the spheres has got the size of the pixel! Within the group "Scatterobject" there are two more presets available. "Pyramid" and "Teapot". Of course you can scatter also any other objects. For this purpose there is the "list box" in this group.

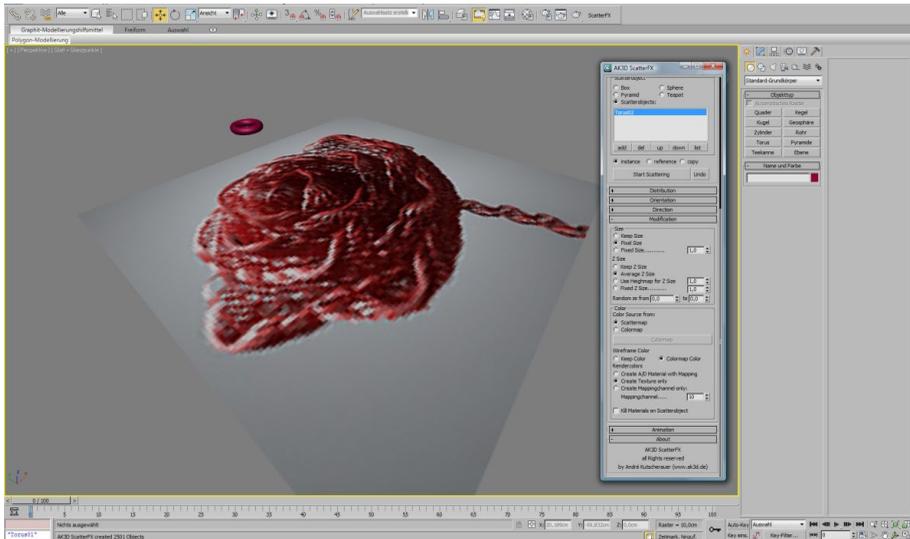
Click "Undo". Then we create a torus of any size in the viewport. In the dialog we click on "add" and select the torus in the viewport. The name of the selected object in the dialog box will appear.

We click on "Start Scattering".

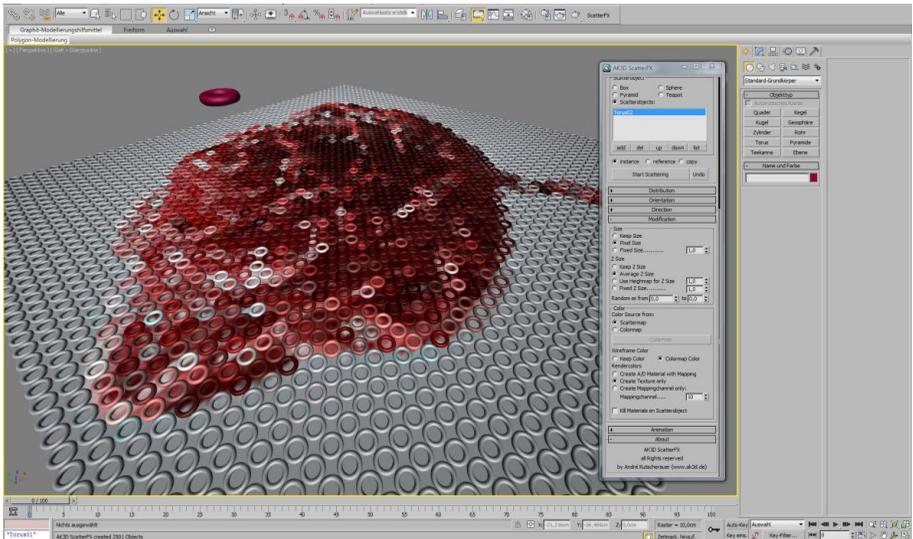


As a result, the “Baseobject” is covered with pixel scaled toruses. Please note that these toruses are highly distorted to meet the pixel size. Of course, we have full influence on this scale.

Then we click on "Undo" to reset the Scattering.

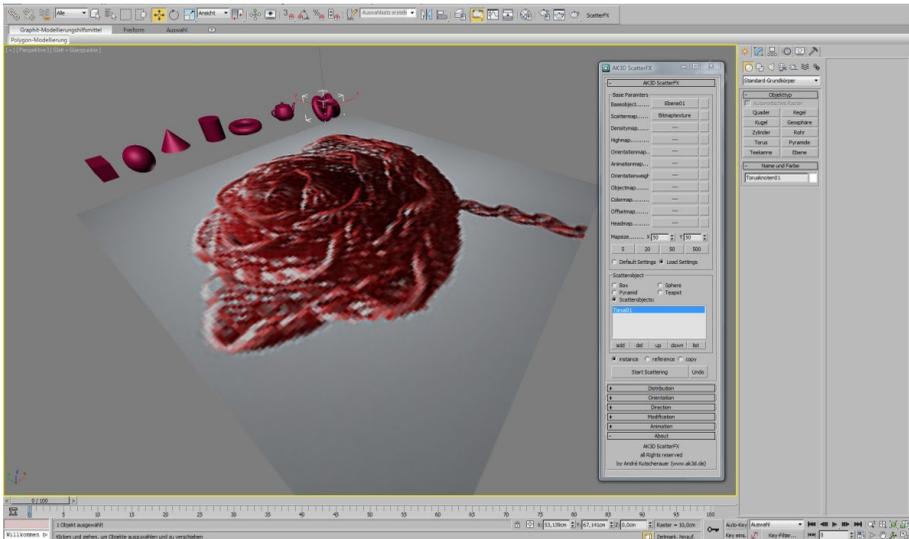


Now we click on the tab "Modification" in ScatterFX. By default "keep Z size" is selected. This takes the given size of objects and leaves its height unchanged. In popup menu, we click in the field of "Z size" and change the setting to "average Z size" and click on "Start Scattering".



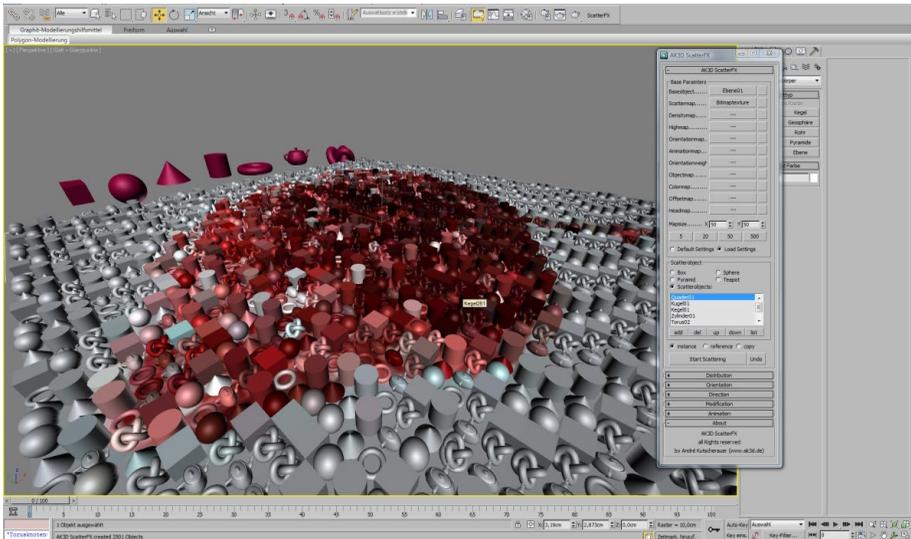
The result is already considerably more "pleasant". Z scaling of objects is now generated from an average of X and Y scaling of the object. This scaling is defined by the texture pixel size and the resolution.

We click "Undo" to reset the scene again.



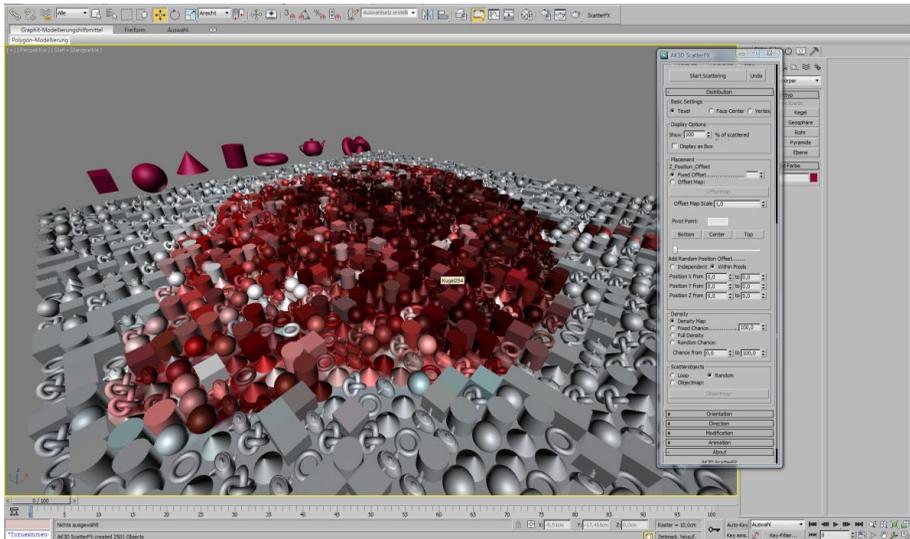
Now we create several basic objects next to the Baseobject. The size of the objects is not important. We now have two ways to get these objects into ScatterFX. Either we click the "add" button and click on each object individually, or, if it's a huge amount of objects, we click on "List" to bring up the "scene explorer" where we easily can select a lot of objects. In both cases, the objects finally appear in the list of ScatterFX. If we have an object in the list, which we want to remove, click on the object in the list and click the "Del" button.

We click on "Start Scatterung" to start the process.



All objects in the list are used as Scatterobjects now. The order in which those objects are placed is at “loop”. That means that they are placed one after another defined by the list. To change that, we click on the tab "Distribution". This is by default set to "loop". This means that the order in the list is used. If we want change the order, we can use the buttons “up” and “down” after selecting an object in the list. Now we want to make the order to “Random” so we click on “Random” in the Distribution group.

We click on "Start Scattering" to prove this new setting.

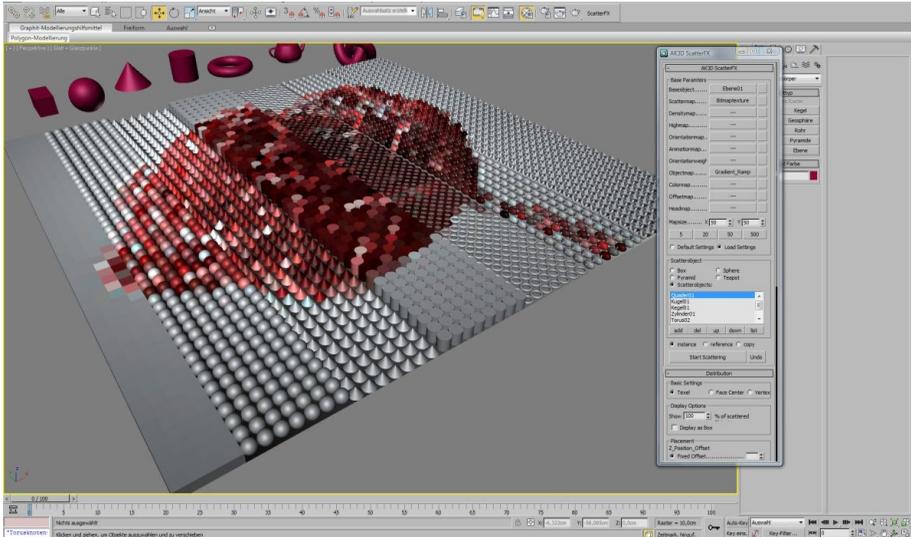


Now the objects are placed with a random order. The difference is marginal in this example. If we use fewer objects the effect will be more visible. Now we turn to "more control" of the order of the objects. For this purpose, ScatterFX has the "Objectmap".

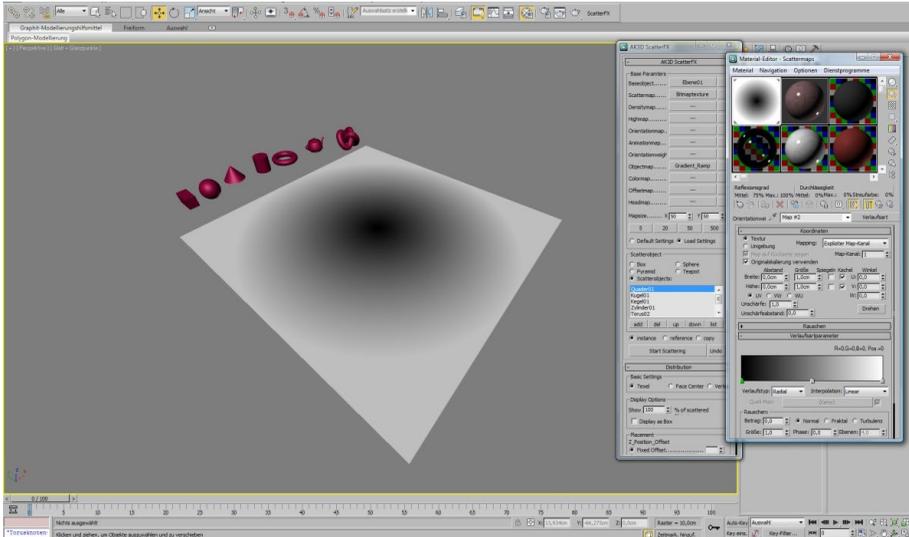
We click the "Undo" button.

Then we change the setting from "Random" to "Objectmap". Then we scroll up and click on "Objectmap". In the map dialog we select the map "Gradient Ramp". The map is automatically displayed in the view pane.

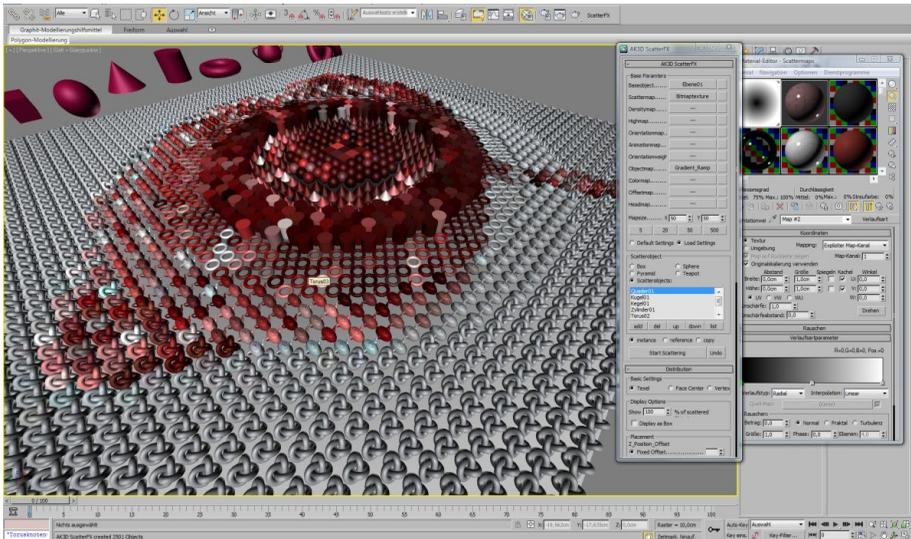
We click on "Start Scattering".



The objects are now distributed based on the brightness of the “Objectmap”. That means that the first Object in the list is placed on the brightest pixel of the objectmap and the last object in the list is placed on the darkest pixel. All objects in between are placed on their corresponding grey color. This function “scales” itself, so that we can distribute any number of objects with this method. This gives maximum control, where what object should be placed at!



To better represent this feature we click on "Undo". In the materials editor, click the "Objectmap" and adjust the Gradienttype to "Radial". We click on "Start Scattering" again.

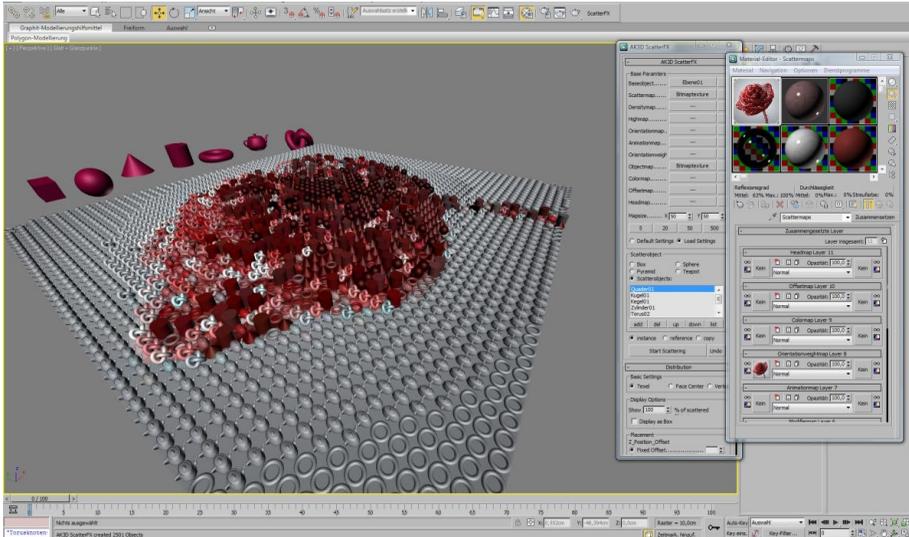


The objects are placed now on the basis of the radial gradient. In other words, the torus knot uses all white pixels of the "Objectmap" and all black pixels the box. All objects in between get their corresponding RGB value. We can use not only parametric maps but also bitmap maps with this function.

We click the "Undo" button. Then we click on the button "Gradientramp" and in the following window, we chose "Bitmap".

We locate the small image "Rose\_of\_Rings.jpg". The image will appear automatically in the view pane.

We click on "Start Scattering" again.

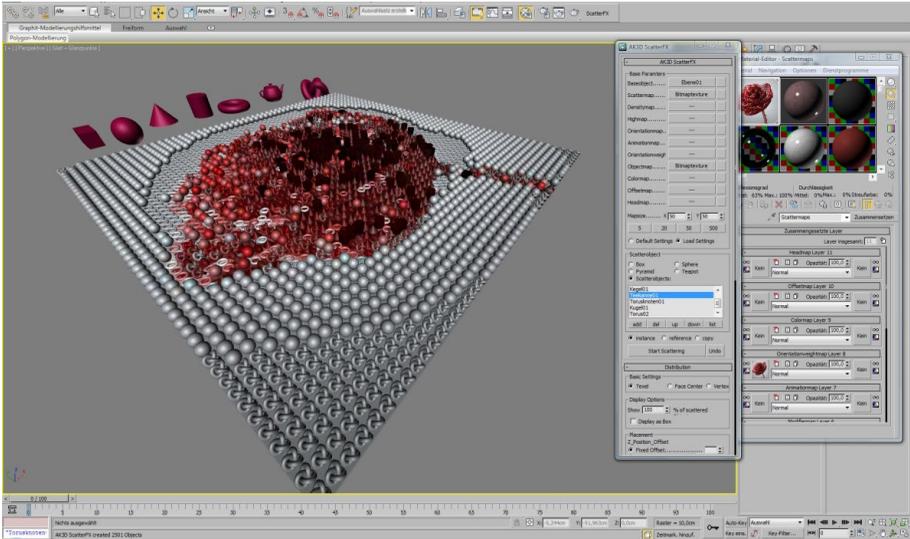


Now, the objects have the order of the brightness of the bitmap. The torusses and teapots are used at the brightest points of the image and the darkest again the pipe and cylinder.

We click "Undo".

If we want to change the order, what object should take what brightness, we can simple re-sort them in the list with the “Up” and “Down” buttons. In the following, I will change the list of objects according to its “visual size”. That means:

1. block; 2. Cylinders; 3. Cone; 4. Tea pot; 5. Torus node; 6. sphere; 7. torus.



As a result, the result is significantly changed. There are many ways to influence the result with the help of these options.

Now you got the basic features explained. Please feel free to try your own settings.

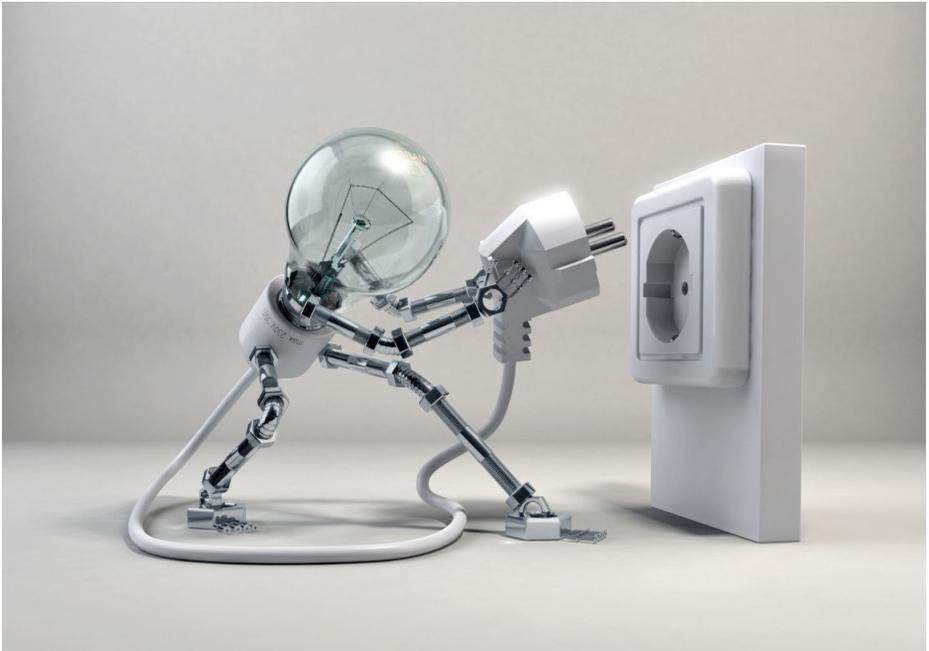


Dandelion of screws

# 5

## ABOUT THE AUTHOR

André Kutscherauer is 3D artist in munich, germany. He has worked for many years in the field of 3D visualization. He visualized thousands of designs for a major household appliance manufacturer and create 3D animations for marketing and advertising. In addition to its main activity, he realizes free art projects, is author and worked as a freelance lecturer at universities.



Selfillumination

