

### Version 1.0b Introduction Guide



#### **Table of Contents**

Introduction3
Requirements4
Installation5
RandomBubbles toolset (Overview) 6
Quick Start tutorial7
Primitive creation tools8
Blobby Sphere8
Blobby Ellipsoid9
Blobby Segment 10
Plane DBO11
RandomBubbles - Emitter tool (Overview) 12
RandomBubbles - Emitter tool (Details) 13
Source types 13
CommandLine Options 14
Volume types 15
Custom Options 19
Output types 20
RiBlobby Exporter / Setup 21
Blobby object selection23
Blobby object visibility 23
Copyrights, Trademarks & Credits
FAQ

#### Introduction

RandomBubbles is a toolset of vbScripts written for *Rhinoceros 3.0.* It was initially developed to have a controlable emitter of random points or spheres in 3D space or user defined volumes.

The toolset contains scripts to create some simple geometry primitives like spheres, ellipsoids and sausage-like cylinders. Using an exporter script, objects can be written to a *RenderMan* based geometry archive (RIB), which can be recognized by *AIR* as a group of Ri\_Blobby objects.

Ri\_Blobby objects are implicit surfaces which are evaluated at rendertime as described in *RenderMan* ApplicationNote#31.

RandomBubbles needs *RhinoMan* to be integrated completely and provide a fast workflow and handling of material assignments and rendering with *AIR*.

#### Requirements



To use the emitter script, which is the main part of this toolset, at least *Rhinoceros* 3.0 Sr2 is required.

Since the toolset is oriented to the creation of blobby objects which are only evaluated upon rendertime by *AIR* you will need *AIR* version 2.6.0 o rhigher.

(It is also possible to render your emitted geometry primitives with other renderers, however only *AIR* is currently able to form an implicit surface from it.)

To provide a good workflow beetween *Rhinoceros*, *AIR* and the toolset, *RhinoMan* by Brian Perry is highly recommended. Using *RhinoMan*, it is also possible to quickly apply and change materials of your blobby geometry before rendering.

Attached are some links to the above packages:

Rhinoceros 3.0 :	www.Rhino3D.com
AIR:	www.Sitexgraphics.com
RhinoMan:	www.RhinoMan.com

#### Installation

Extract the content of the WinZip file to a directory of your choice. You should have the following files:

AddSingleBlobPlane.rvb AddSingleEllipsoid.rvb AddSingleSegment.rvb AddSingleSphere.rvb RandomBubbles.rvb RiBlobbyExporter.rvb RiBlobbySetupExport.rvb Ri\_ChangeInfluence.rvb Ri\_ChangePlaneHeight.rvb Ri\_SelectByInfluence.rvb Ri\_SelectByType.rvb Ri\_ShowBlobs.rvb Ri\_HideBlobs.rvb RandomBubbles.tb

Files with the extension \*.rvb are the Rhino scripts, the "RandomBubbles.tb" file is the Rhino workspace, with a set of toolbar buttons to call all the scripts.

To install the RandomBubbles scripts in Rhino go to: *RhinoOptions / Files / SearchPath / FileSearchPath* 

Document Properties		×
<ul> <li>Document Properties</li> <li>Render</li> <li>Mesh</li> <li>Units</li> <li>Dimensions</li> <li>Grid</li> <li>Notes</li> <li>Summary</li> <li>Web Browser</li> <li>Bhino Options</li> <li>View</li> <li>SpaceBall</li> <li>Alases</li> <li>Appearance</li> <li>Colors</li> <li>Files</li> <li>General</li> <li>Mouse</li> <li>Keyboard</li> <li>Display</li> <li>OpenGL</li> <li>Modeling Aids</li> <li>RhinoMan</li> <li>RhinoMan</li> <li>Alienter</li> </ul>	Template files Location: Default Autosave Save gevery: Autosave file: Always gave before: Save render and Search path File search paths E\\Rhino Daten\\R	eel\Rhinoceros\3.0\English\Template Files          C:\WINNT\Profiles\All Users\Anwendungsd          20

#### Document Properties / Rhino Options dialog

#### To install the RandomBubbles toolbar in Rhino go to: *Tools / ToolBar Layout / File / Open* open the file "RandomBubbles.tb" from your directory.

Toolbars dialog

😽 T	oolbars		×
<u>F</u> ile	Toolbar	<u>H</u> elp	
Ne	ew	Ctrl+N	
<u>0</u> p	ben	Ctrl+O	
<u>C</u> le	ose		
Clo	os <u>e</u> All		
Sa	ave	Ctrl+S	
Sa	ve <u>A</u> s		
Sa	ave All		ABonus Lools.tb
Pr	operties		
Ex	jit		
	Bonus Ed	dit tools	-
	Bonus M	esh Tools	
	Bonus Mi	scellaneou	is Tools
	Bonus Re	ender Tool	s
	Bonus So	olid Tools	
	Bonus Su	uface Too	s
	Bonus To	ols	
	Bonus Tr	ansform To	ools 💌

#### The RandomBubbles toolset - Overview

Once the RandomBubbles.tb workspace file has been opened inside Rhinoceros the following tools should be accessable from the RandomBubbles toolbar:





0

.....

The creation tools can be used to add new Blobby primitives like spheres, ellipsoids, segments and planes to the rhino document. Before a new primitive is added, the user will be prompted for its influence mode. (Positive or Negative)

These influence assignments can also be changed for existing objects after they have been created. You can use the Change Influence tool for this task.

The influence mode controls the type of operation with which the primitive will be rendered by *AIR*. Blobbys with positive influence are added together, then all Blobbys with negative influence are subtracted from the group of positive Blobbys.

The emitter tool can be used to create multiple new Blobby primitives with a predefined influence mode. There are several options to control the distribution.

The RIB exporter is a tool to export and render all Blobby Primitives and objects in the document with a single click.



To hide or show Blobby primitives, the Visibility tool can be used.

#### **Quick Start tutorial**

The following Quick Start tutorial will explain the necessary steps to render Blobby primitives with the RandomBubbles toolset, Rhinoceros, AIR and RhinoMan:

- 1. Create some Blobby primitives, eg. with the emitter tool
- 2. Create a "Loader" object
- 3. Open the RiBlobby exporter setup dialog, click "OK"
- 4. Export / Render using the RiBlobby exporter button
- To create some Blobby primitives, use the RandomBubbles 000 (emitter) script. Start the emitter and click on "Preview", then press ENTER.

The "Loader" object is a dummy object to read one geometry RIB archive which is created from the Blobby primitives. This "Loader" object is also used to apply a RhinoMan material to the group of Blobby primitives which is loaded from it. (It's not necessary to apply a material to the Blobby primitives !)

To create a "Loader" object, make a surface or polysurface, then open the object properties dialog in Rhino and choose "RhinoMan" from the dropdown list. Open the "Custom RIB Commands" dialog by clicking on the "User Options" button.

Custom RIB Commands	×
New Delete 🕈 🗲 File Path	ntrol for Value:
Enable Name	Value
ReadArchive	MyArchive.rib
OK	Cancel

Note:

Since the RIB archive is written to RhinoMan output folder, it is not neccessary to define a path to the archive. If the archive is written elsewhere, define the path either by entering or by clicking on the small browse button.

RhinoMan Custom RIB Commands dialog (geometry)

Set the "Interface Control for Value" to File Path, then add a new Custom RIB Command by clicking on the "New" button. Enable the new item, then add "ReadArchive" under "Name", and "MyArchive.rib" under "Value" as shown above.

**STEP 3+4** 

You can now export and render the Blobby primitives using the RiBlobby exporter. ("AutoRender" option must be activated to render automatically after export is done).

0



**Rhinoceros Properties** dialog (RhinoMan tab)

#### Adding Primitive Objects (Blobby Sphere)

To add single spherical Blobby primitives to your Rhino scene, left click on the "Add single RiBlobby spheres" button.

> The command will then prompt you for the influence mode of the object to create. By default, a positive mode is used.



Rhinoceros Command Prompt asking for the influence mode

Once the influence has been defined, the command will ask for the pick points of a sphere as usual. To complete adding single Blobby spheres, press ENTER.



two spherical blobby primitives in Rhinoceros

If both of the two spheres got a positive influence mode attached, the resulting image rendered with AIR should look like this after exporting the RIB archive (Conversion from Rhinoceros geometry to Ri\_Blobby geometry).



#### Adding Primitive Objects (Blobby Ellipsoid)

To add single elliptical Blobby primitives to your Rhino scene, left click on the "Add single RiBlobby ellipsoid" button.

> The command will then prompt you for the influence mode of the ellipsoids to create. By default, a positive mode is used.



Rhinoceros Command Prompt asking for the influence mode

Once the influence mode is defined, the command will ask for the pick points of an ellipsoid as usual. To complete adding single Blobby ellipsoids, press ENTER.



two elliptical blobby primitives in Rhinoceros™

Note:

Spheres and Ellipsoids are exported as objects with the opcode (1001). Apply eg. axial scaling to a sphere to form an ellipsoid. Do not shear, taper or otherwise deform these

primitive types !

If both of the two ellipsoids got a positive influence mode attached, the resulting image rendered with AIR should look like this after exporting the RIB archive (Conversion from Rhinoceros geometry to Ri\_Blobby geometry).



#### Adding Primitive Objects (Blobby Segment)

To add single segment Blobby primitives to your Rhino scene, left click on the "Add single RiBlobby segment" button.

The command will then prompt you for the radius and influence mode of the segments to create. By default, a positive value is used.



Ele Edit View Curve Surface Solid Iransform Tools Dimension Analyze Render Bonus RhinoMan He Script file to load (List): AddSingleSphere.rvb Choose RiBlobby Influence mode <Positive> (Positive Negative):

Rhinoceros Command Prompt asking for the influence mode

Once the radius and influence has been defined, pick the StartPoint / EndPoint, similar like drawing a line. A segment Blob primitive will be created accordingly.



Note:

Segment primitives are exported as Ri\_Blobby objects with the opcode (1002). You can move, rotate and use uniform scaling on them. Do not shear, taper or apply non uniform scale to this type of primitives !

six segment blobby primitives in Rhinoceros™

Segment Blobby primitives are blended together just like spheres and ellipsoids when rendered with AIR.



#### Adding Primitive Objects (Plane DBO)

To add a plane primitive to your Rhino scene, left click on the "Add single RiBlobby Plane" button.

The command will then prompt you for a plane strength and the influence mode of the plane to create. By default, a strength of 2 and a positive influence mode is used.

<u>F</u> ile	<u>E</u> dit	⊻iew	<u>C</u> urve	<u>S</u> urface	S <u>o</u> lid	<u>T</u> ransform	Tools	<u>D</u> imension	<u>A</u> nalyze	<u>R</u> ender	<u>B</u> onus	RhinoMan	<u>H</u> elp
S	Script	file to	load ( l	List ): Ad	dSing	leBlobPlar	ie.rvb						
Ē	nter	plane	streng	<b>,th &lt;2&gt;</b> :									
Rhinoceros Command Prompt asking for plane strength													
<u>F</u> ile	<u>E</u> dit	⊻iew	<u>C</u> urve	<u>S</u> urface	S <u>o</u> lid	<u>T</u> ransform	Tools	<u>D</u> imension	Analyze	<u>R</u> ender	<u>B</u> onus	RhinoMan	<u>H</u> elp
E	Enter plane strength <2>: 2												

Choose RiBlobby Influence mode <Positive> ( Positive Negative ):

Rhinoceros Command Prompt asking for the influence value

Once strength and influence mode are defined, pick the corner points like drawing a rectangular surface. A new blobby plane primitive will be created accordingly.

Plane primitives can be blended to any other geometry, like in the example below. A blobby sphere is blended to the plane DBO (dynamic blob op). To control the size of blending, select a plane primitive and right click on the button to "Change plane maximum influence height".



a sphere and a plane primitive in Rhinoceros

Note:

Plane primitives have the rib opcode (9000). The surface normal is the direction vector for the repulsion. Blobby planes must be planar to be exported properly. Do not deform etc, just like for all primitives.



a positive Ri\_Blobby sphere blended to a plane (DBO) with increasing strength / plane maximum height values, from left = 0.5 to right = 2.5

#### RandomBubbles - Emitter tool (Overview)

°°°°

The emitter tool can be used to create multiple new Blobby primitives with a predefined influence value. There are several options to control the distribution and the randomness of new created primitives.

Once the tool has been invoked, the options below are displayed and accessable from the Rhino CommandLine:

Elle Edit View Curve Surface Solid Iransform Tools Dimension Analyze Bender Bonus RhinoMan Help Script file to load (List.): RandomBubbles.rvb RandomBubbles (Source BubbleCount MinSize MaxSize Dimension Volume Collision Options Preview Keep.):

Rhinoceros Command Prompt asking for plane strength

Source	Defines the creation <i>Source</i> for the creation of new random blobby primitives. By default a single point <i>Source</i> located at the world origin is used.
BubbleCount	The number of primitives which will be created, once the <i>Preview</i> option has been invoked.
MinSize	A value to control the minimum bounding size of the new created primitive. The value must be smaller / <i>MaxSize</i> .
MaxSize	A value to control the maximum bounding size of the new created primitive. The value must be higher / <i>MinSize</i> .
Dimension	A value to control the overal bounding <i>Volume</i> size of the choosen <i>Volume</i> type.
Volume	Defines the bounding <i>Volume</i> type which will be used for the creation of randomly distributed blobby primitives.
Collision	A toggle to enable / disable <i>Collision</i> detection between new created objects. Useful to avoid intersections.
Options	Options and toggles to control collision detection offset, screen redraw, skip limiter, and to define the output type and influence of new created objects.
Preview	Start the creation of new random objects from the entered emitter settings and values. Provides StatusBar feedback.
Кеер	Option to Keep previewed objects and continue with the

command, without starting it again and again.

	<i>**</i> *	The emitter tool will create new randomly distributed primitives inside a pre-calculated volume which can be defined using the <i>Volume</i> option. To constrain this <i>Volume</i> and its centerpoint to a specified location, the <i>Source</i> option must be entered and changed.
	Source/	The Command Line will give access to these <i>Source</i> types. By default, a <i>Point</i> located at the world origin is used for the pre-calculated volume centroid.
		Eile       Edit       View       Qurve       Surface       Solid       Iransform       Tools       Dimension       Analyze       Bender       Bonus       RhinoMan       Help         RandomBubbles       (Source       BubbleCount       MinSize       MaxSize       Dimension       Volume       Collision       Options       Pr         Choose a source       object type < Point>       (Point       Qurve       Surface       CustomVolume       ):         Rhinoceros       Command       Prompt       asking       for       creation       source
	Point	The <i>Point</i> source is the fixed centerpoint of the volume type defined in the <i>Volume</i> option. Change the location of this point by entering the <i>Point</i> option, then pick the new location of your choice.
	Curve	The <i>Curve</i> source will ask for a Curve to emit primitives from. Actually, random points will be calculated on the curve and new primitives will be created around these points near the curve. The active <i>Volume</i> type will be "extruded" perpendicular to the creation source curve.
	Surface	To emit new primitives above or near a surface, change the source type to <i>Surface</i> and select a Nurbs surface. The distribution hight of new primitives around surfaces can be controled using the <i>Dimension</i> command option. The surface normal vector decides on which side of the surface the creation of new primitives will be performed, once the <i>AbovePlane Volume</i> option is used.
s e e.	CustomVolume	To create new primitives inside a custom volume object, enter the <i>CustomVolume</i> option and select one closed surface or polysurface. Note that you cannot access the regular <i>Volume</i> and <i>Distance</i> options, since the creation

of new primitives will be calculated from CustomVolume.



200 spherical primitives emitted around a curve, using the **Curve** Source and **Cubic** Volume type.

#### RandomBubbles - Emitter tool (CommandLine Options)

- Once a *Source* type has been defined, use *Preview* to create a defined number of new primitives. The amount of primitives to created is entered under *BubbleCount*.
- **BubbleCount** Controls the number of new created (emitted) primitives. To reduce filesize and export times, it is recommended to use small counts with eg. sphere primitives as output first, then change the *OutputType* to *Point* (under *Options*).
- MinSizeThe minimum allowed radius of the new created primitive.Note that this value cannot be higher than MaxSize value.
- MaxSizeThe maximum allowed radius of the new created primitive.Note that this value cannot be smaller than *MinSize value*.
- **Dimension** The *Dimension* value is an overal control for the bounding size of the current choosen *Volume* type. *Dimension* controls the distance of new created primitives from the source or the point on the source predefined under *Source*.
- VolumeThe Volume type defines the 3-dimensional space in or on<br/>which new primitives are randomly created. The number<br/>of available volumes depends on the current Source type.<br/>Pre-calculated volumes are explained on the next pages.
- **Collision** The *Collision* toggle is a mode to enable / disable collision detection for new created primitives. Note that this setting can drastically slow down the creation of new objects if the bounding volume *Dimension* was set to small. To speed up the creation and to avoid uneccessary skipped calculations, there is an *CollisionOffset* value, which can be modified at the *Options* CommandLine option.

 
 Preview
 To Preview command options and settings, click on the Preview CommandLine option. The StatusBar provides information of created and skipped () calculations while the emitter works...

Snap Ortho Planar Osnap 121/512 bubbles created.... (70)

Rhinoceros StatusBar



350 spherical primitives emitted around a NURBS Surface using a **Surface** Source and **AbovePlane** Volume type.

 To give maximum control over the 3-dimensional space in which new primitives are emitted randomly, the emitter tool offers several pre-calculated bounding *Volume* types.

 Volume\
 Once the *Volume* option is entered from within the Rhino CommandLine, these *Volume* types are accessable:

 Ele Edit View Durve Surface Solid Transform Tools Dimension Analyze Bender Borus RhinoMan Help

 RandomBubbles (Source BubbleCount MinSize MaxSize Dimension Volume Collision Options Preview Keep): Volume

 RandomBubbles (Source BubbleCount MinSize MaxSize Dimension Volume Collision Options Preview Keep): Volume

Rhinoceros Command Prompt asking for pre-calculated bounding volume type

The *Cubic* bounding *Volume* type will constrain the creation of new primitives to a cubic volume. To control the size of this pre-calculated cube, the *Dimension value can be used.* 

Cubic	Settings:	
	Source:	Point
Chine Pin all to Burke	BubbleCount:	1500
	MinSize:	0.25
	MaxSize:	0.75
	Dimension:	20
	Volume:	Cubic
	Collision:	On
	CollisionOffset:	-0.2
Dimension 1	Output:	Sphere

**Spherical** 

Cubic

*Spherical* bounding *Volume* type will constrain the creation of new primitives to a spheric volume. Control the radius of this pre-calculated sphere with the *Dimension value*.



Settings:	
Source:	Point
BubbleCount.	1200
MinSize:	0.25
MaxSize:	1.0
Dimension:	20
Volume:	Spherical
Collision:	On
CollisionOffset:	-0.2
Output:	Sphere

Note: Only the random emitted primitives are created from the script. The *Cubic / Spherical* bounding geometry above is pre-calculated and just shown to explain different volume types.

Volume Volume types can be combinied with most Source types accessable from the Source CommandLine option. The examples below are made with the Point Source option. A Volume type cannot be selected if the CustomVolume Source is used.

**Cylindrical** The *Cylindrical* bounding *Volume* type will constrain the creation of new primitives in a cylindrical volume. Control the cylinder radius and hight with the *Dimension value*.



Settings:	
Source:	Point
BubbleCount:	1000
MinSize:	0.25
MaxSize:	0.75
Dimension:	20
Volume: Cy	lindrical
Collision:	On
CollisionOffset:	-0.2
Output:	Sphere

OnSphere The OnSphere bounding Volume type will constrain the creation of new primitives to the surface of a sphere. All centerpoints of new emmited primitives will be constrained to the pre-calculated spherical surface. Control the radius of the bounding sphere with the Dimension value.



Settings:	
Source:	Point
BubbleCount.	300
MinSize:	0.25
MaxSize:	0.75
Dimension:	20
Volume:	OnSphere
Collision:	On
CollisionOffset	: 0
Output:	Sphere

Note: Only the random emitted primitives are created from the script. The *Cylindrical / OnSphere* bounding geometry above is pre-calculated and just shown to explain different volume types.

Volume Volume types can be combinied with most Source types accessable from the Source CommandLine option. The examples below are made with the Point Source option. A Volume type cannot be selected if the CustomVolume Source is used..

OnCircleThe OnCircle bounding Volume type will constrain the<br/>creation of new primitives to a circle in the active CPlane.<br/>Control the circle radius with the Dimension value.



Settings:		
Source:	Point	
BubbleCount.	50	
MinSize:	0.25	
MaxSize:	1.0	
Dimension:	20	
Volume:	OnCircle	
Collision:	On	
CollisionOffset:	-0.2	
Output:	Sphere	

OnPlane

The *OnPlane* bounding *Volume* type will constrain the creation of new primitives to a square surface in the active CPlane. All centerpoints of new emmited primitives will be constrained to the pre-calculated square. Control the size of the pre-calculated surface with the *Dimension value*.

![](_page_16_Picture_8.jpeg)

Settings:	
Source:	Point
BubbleCount.	100
MinSize:	0.25
MaxSize:	1.0
Dimension:	20
Volume:	OnPlane
Collision:	On
CollisionOffset:	0
Output:	Sphere

Note: Only the random emitted primitives are created from the script. The *OnCircle / OnPlane* bounding geometry above is pre-calculated and just shown to explain different volume types.

Volume Volume types can be combinied with most Source types accessable from the Source CommandLine option. The examples below are made with the Point Source option. A Volume type cannot be selected if the CustomVolume Source is used.

AbovePlane The *AbovePlane* bounding *Volume* type will constrain the creation of new primitives above a square in the active CPlane. Control the square size with the *Dimension value*.

![](_page_17_Picture_4.jpeg)

Settings:		
Source:	Point	
BubbleCount.	100	
MinSize:	0.25	
MaxSize:	1.0	
Dimension:	20	
Volume: Abo	AbovePlane	
Collision:	On	
CollisionOffset:	0	
Output:	Sphere	

OnSourceThe OnSource bounding Volume type will constrain the<br/>creation of new primitives to a curve or surface object.<br/>Centerpoints of new emmited primitives will be positioned<br/>on the selected Source object. (Curve or Surface)

![](_page_17_Picture_7.jpeg)

Settings:		
Source:	Point	
BubbleCount.	100	
MinSize:	0.25	
MaxSize:	1.0	
Dimension:	20	
Volume:	OnSource	
Collision:	On	
CollisionOffse	<i>t:</i> 0	
Output:	Sphere	

Note: Only the random emitted primitives are created from the script. The *AbovePlane* bounding geometry above is pre-calculated and shown to explain different volume types.

#### **RandomBubbles - Emitter tool (Custom Options)**

**Options\** To fine tune several features of the emitter script, these custom *Options* can be adjusted from the CommandLine, *Options* field:

 Eile
 Edit
 View
 Qurve
 Sulface
 Solid
 Iransform
 Tools
 Dimension
 Analyze
 Bender
 Bonus
 RhinoMan
 Help

 Info:
 CollisionOffset
 and
 SkipLimiter
 settings
 only
 apply
 if collision
 detection
 mode is
 enabled

 Custom
 Options
 (<a href="CollisionOffset\_SkipLimiter">CollisionOffset</a>
 SkipLimiter
 Redraw
 AbovePlaneOffset
 OutputType
 RiBlobby ): |

Rhinoceros Command Prompt asking for Custom Option to change

- **CollisionOffset** By default, a *Collision* between primitives will be found, if the distance between the primitives is less than zero. The *CollisionOffset* value can be used to fine tune the maximum allowed distance between primitives, before a *Collision* is detected. Values above zero will result in earlier *Collision*, to allow controlled intersections between primitives, negative values can be entered.
- SkipLimiterEnabled by default, the SkipLimiter stops the calculation of<br/>new primitives if the count of avoided (skipped) collisions is<br/>much higher than the value entered under BubbleCount.<br/>The creation of new primitives can also be stopped just by<br/>holding the ESC key for a short time.
- RedrawThe Redraw toggle controls the Rhinoceros screen redraw,<br/>while the creation of new primitives is running. To watch the<br/>creation process of the emitter, enable the Redraw toggle.
- AbovePlaneOffset The AbovePlaneOffset value controls the distance between the plane or source surface, once enabled. The AbovePlane Volume type has to be active, otherwise this option will have no effect.
- OutputTypeThe Option OutputType determines, which type of primitives<br/>will be created by the emitter script. A complete description<br/>of all OutputType modes can be found on the next page.
- **RiBlobby**The *RiBlobby Option* determines with which influence mode<br/>the new emitted primitives will be created. Once the *RiBlobby*<br/>*Option* is entered, define the mode of influence shown below:

### RandomBubbles - Emitter tool (Output types)

OutputType\	To control the type of primitives which will be emmited while the script runs, choose the desired OutputType from within the Rhinoceros CommandLine:	
	<u>File Edit View Curve Sufface Solid Iransform Tools Dimension Analyze Render Bonus RhinoMan Help</u>	
	Custom Options (CollisionOffset SkipLimiter Redraw AbovePlaneOffset OutputType RiBlobby): OutputType Choose an OutputType: <sphere> (Sphere Ellipsoid Drop Point):</sphere>	
	Rhinoceros Command Prompt asking for primitive OutputType	
Sphere	The <i>Sphere OutputType</i> will output spherical primitives. These primitives will be exported with RIB opcode 1001.	
Ellipsoid	The <i>Ellipsoid OutputType</i> will output elliptical primitives. These primitives will be exported with RIB opcode 1001.	
Drop	The <i>Drop OutputType</i> will output <i>Drop</i> shaped primitives. These primitives will be exported with RIB opcode 1001.	
Point	The <i>Point OutputType</i> will output <i>Point</i> objects. These objects will be exported with RIB opcode 1001 (Sphere).	

![](_page_19_Picture_3.jpeg)

OutputType = Sphere

![](_page_19_Picture_5.jpeg)

OutputType = Drop

![](_page_19_Picture_7.jpeg)

OutputType = Ellipsoid

![](_page_19_Picture_9.jpeg)

OutputType = Point

#### **RiBlobbyExporter (Setup)**

![](_page_20_Picture_2.jpeg)

To render the primitives created with the toolset as blobby objects with AIR, they have to be written to a RIB archive using the RiBlobbyExporter button.

Note: The RiBlobbyExporter is a "one shot" command, it will just create the RIB archive with the settings made in the RiBlobbyExporter Setup dialog shown below:

![](_page_20_Picture_5.jpeg)

RiBlobbyExporter Setup (Defaults)

"<u>Print export progress to CommandLine</u>" will print out info about object count, type, color assignment and operation mode (union or subtract) to the Rhinoceros CommandLine.

"Include RiBlobby annotations to RIB file" will include info about object type and index in the exported RIB archive.

"<u>Use RhinoMan`s default output directory</u>" will try to read the RhinoMan output directory from the registry. If this option is unchecked, you can manually define an output path in the dialog shown below:

OutputFolder	? X
Select a destination folder to create RiBlobby file "arch	ive.rib''
GIGA 1     GIGA 1     GIGA 1     G332-Diskette (A:)     G0ga1 (C:)     G0ga1 (D:)     G0ga1 (E:)     G	
OK. Abbre	chen

OutputFolder dialog

#### **RiBlobbyExporter (Setup)**

![](_page_21_Picture_2.jpeg)

"<u>Adjust Clipping before starting Rendering</u>" will add two points, to define the bounding box of all blobby primitives, in the document before starting a new rendering. Note: Setting only applies if the "Autorender" option is enabled.

"<u>Allocate Plane DBO inside output directory</u>" will search for the required file "Plane.dll" inside the output directory. Alternatively, a path to the procedurals can be defined: Rhino Options / RhinoMan / SearchPaths / Procedural

"<u>Assign vertex coloring to RiBlobby objects</u>" will add color information (derived from Rhino wireframe color) for each primitive to the RIB archive. Blending between all RiBlobby primitive types will be performed only once the scene is rendered with AIR.

"<u>Invert negative .....colors</u>" will invert the color for objects with negative influence mode. If disabled, the color of the primitive is negated eg. RGB(255,0,0) > RGB(0,255,255). This option only applies if the vertex coloring is enabled.

"*Positive Influence between negative blobs*" will export all negative objects as a group with smooth blending. If this option is disabled, all negative blobs will be subtracted individually from the group of positive blobs resulting in sharper edges in the area of subtraction.

"Include RIB: ShadingInterpolation smooth" is a custom RIB call which should be added to the RiBlobby archive if it will be rendered with PRMan. It is not necessary to use this option if you render with AIR.

"<u>AutoRender RiBlobby scene after RIB export</u>" will start a new Rendering once the geometry archive has been written. If enabled, this option does hide your blobby primitives inside Rhinoceros before starting the new Rendering. Once this Rendering has been started, all blobby primitives are shown again, like in the initial state before the rendering.

Note: You need a "loader" object to read the RIB archive ! (more info about "loader" objects in the QuickStart section)

![](_page_21_Picture_11.jpeg)

Blobby object rendered without vertex color

![](_page_21_Picture_13.jpeg)

Blobby object rendered with vertex color

#### **Selection tool**

![](_page_22_Picture_2.jpeg)

To select Blobby primitives by their influence mode or type of primitive, the selection tool can be used in large scenes.

Left click on the selection button to select by primitive type, Right click on the selection button to select by influence.

Note that only primitives created with the RandomBubbles toolset can be selected with the selection scripts.

![](_page_22_Picture_6.jpeg)

Rhinoceros Command Prompt asking for Blobby primitive type to select

![](_page_22_Picture_8.jpeg)

Rhinoceros Command Prompt asking for Blobby influence mode to select

#### Visibillity tool

**%** 

To hide or show all Blobby primitives the Visibillity tool can be used. Note: This works for unlocked Blobby primitives only.

Left click on the visibility button to hide all Blobby primitives. Right click on the visibility button shows all Blobby primitives.

Note that only primitives created with the RandomBubbles toolset can be hidden or shown with the visibility tool.

#### **Copyrights, Trademarks and Credits**

RANDOMBUBBLES IS PROVIDED BY THE COPYRIGHT HOLDER AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

RandomBubbles is Copyright 2004 Clement Greiner, CG3D. All rights reserved.

*AIR* and the Visual Shading and Lighting Tool Kit software and documentation are Copyright 2000-2004 SiTex Graphics, Inc. All rights reserved.

*Rhinoceros* (Rhino) is Copyright 1993-00 Robert T. McNeel & Associates. All rights reserved.

RhinoMan is Copyright 1999-2004 Brian Perry. All rights reserved.

The *RenderMan*® Interface Procedures and RIB Protocol are Copyright 1988, 1989, Pixar. All rights reserved. RenderMan® is a registered trademark of Pixar.

The *Blue Moon Rendering Tools* (BMRT) are Copyright 1990-1999 Larry Gritz. All rights reserved.

#### Credits

Thank you Brian Perry (RhinoMan) and Scott Iverson (AIR) for your help to write the toolset.

### FAQ - frequent answered questions

#### "All my tools work but if i use the RiBlobbyExporter i get a virus alert from my virus scanner ?"

The exporter uses the vbScript FileSystemObject to write a RIB file to your output directory. To avoid the virusscanner problem, exclude the RandomBubbles folder from the list of folders to scan permanently, or allow the script to write the rib file.

*"I can run all commands in the toolset but if i choose a new template, i sometimes get an "Cannot find the script file...." from Rhino and the command does not start / is not found ?"* 

Make shure the path has been defined under Rhino Options / Files / File search path. After reopening Rhinoceros this path should stay permanent.

# <u>"I have applied RhinoMan materials to my blobby primitives but they are not included in the ribfile or shown in the rendering ?"</u>

To assign a material to one set of interacting blobby primitives, assign it to the "loader" object. All blobby primitives loaded from this "loader" object have the material assigned to it.

#### "I can render materials but i want to render blending vertex colors as well. How does this work ?"

Enable the "Assign vertex colors..." option in the exporter setup and make shure different wireframe colors are assigned to your blobby primitives, then start the render with the exporter.

### "How can i render groups of blobby objects with different materials assigned to each group ?

To render multiple groups of interacting blobby primitives, export each group with the exporter and rename the RIB archive file from "MyArchive.rib" to eg.: "MyArchive\_01.rib" etc. Make shure to use multiple "Loader" objects for each group of interacting blobby primitives together with the correct path/filename in the ReadArchive call for each "Loader" object.

# "If i use the regular Rhinoceros Render button, i get my simple blobby primitives AND the blobby surface rendered. Why does this happen ?"

The regular render button can be used but the blobby primitives should be hidden after export is done and before the rendering is started. Alternatively, the AutoRender option in the exporter setup can be enabled, the exporter then writes the archive, hides all primitives, starts the render and shows all hidden primitives with a single click.

#### "Why do i have to use a loader object to render a set of blobby primitives with material ....?"

The "loader" object is written in the rib file which is generated via RhinoMan. The method was choosen to change the material of a group without rewriting the geometry archive.