



## Introduction

MegaCache is an editor extension of for the Unity game engine, it allows you to import cached animated mesh geometry regardless of the topology, vertex count, material use etc of each mesh in the cached sequence as well as importing Particle systems exported from 3d packages. The system can accept a sequence of OBJ files where each file can have a number of objects all with different materials, MegaCache will import the sequence of files merging multiple objects and material sub meshes into a single mesh object for you, you can then play back the imported sequence either by asking the system to directly change the mesh using its stored list of generated meshes, or the sequence can be read from a generated cache file, or finally be read from a memory based image file. Play back of the animation is blazingly fast no matter which option you choose with the system making use of multiple threads where possible to further increase performance. You can also tell MegaCache to optimize the data that is imported reducing the memory use to under a 1/3rd of storing raw mesh data.

MegaCache is perfect for exporting complex animated meshes that are not possible to recreate using skin and bones, such as fluid simulations, fracturing objects, cloth simulations or anything where the vertex count, face count, materials etc change during the course of the animation. It is also perfect for playing back complex animations made using modifiers or deformers in 3d packages, all the animation can be baked down to a sequence of frames using standard OBJ sequence exporters which are available for all the 3d packages.



Included in the system are components to allow the import of a sequence of OBJ files and play those back as well as an OBJ reference component where multiple objects can playback a sequence from an already loaded source so dozens of objects can all be sharing the same data but playing back at different speeds or positions. Also included are components to playback imported particle simulations either on the Unity Shuriken particle system or the Legacy system.

## Particle Simulations

As well as importing sequences of Mesh geometry MegaCache can also import and playback particle systems such as Particle Flow simulations from 3ds max. Again the system will optimize the memory use for imported simulation and gives you complete control over the playback of the particles, selecting emit rate, scales, speeds etc and works alongside the existing Unity Shuriken and Legacy particle systems allowing you to add even more detail to the finished particle animation.

## Particle Exporters

At release we have available an exporter for 3ds max to exporter Particle Flow simulations, we also have a beta Maya exporter for Maya particle system exporting, please note the Maya plugin is still a work in progress. If you require the exporters please get in touch with the invoice number for your purchase and we will send the exporter of to you.

## Example Videos

This video shows the Object cache system playing back a fluids simulation as well as a fracturing object animation.





This video shows the workflow of exporting a sequence of OBJ files from a 3d package, in this case 3ds Max, and then importing them into Unity using the MegaCache OBJ system.



## MegaCache Object



The OBJ Cache system in MegaFiers is a basic version of the system found in the full MegaCache Asset. It allows you to import any number of OBJ files as a sequence and then play those back in Unity. Each OBJ file in the sequence can have any number of vertices (up to the limit of Unity of 65535) and the counts can change between files as well as the face count, texture coordinates etc. If the OBJ files have materials then those will be imported as well and the system will combine all the mesh and material data into a single object.

You can then select to use the mesh data as it was imported but that can take a lot of memory depending on how many files were imported, or you can save the data to an optimized Cache file which massively reduces the memory as the system will compress the data for you. If you use a cache file as the source the system will stream each frame as it is needed so the memory use comes down to a single frames worth of memory instead of all the data. Or you can choose to make a memory based image of the data for even faster streaming of the data, this option also supports the option for using another thread to pre fetch the next frame of data for you.

## How to use

Using the system is very simple, go to the GameObject/Create Other/MegaCache menu and select the OBJ Cache

Using the system is very simple, go to the GameObject/Create Other/MegaCache menu and select the OBJ Cache option, this will create a new game object in the scene for you with the required components attached. Next you need to set the frame range for OBJ files that you wish to import, so say for example you exported a sequence of 100 frames from your 3d package called Object\_0001.obj to Obj\_0100.obj then you would set the 'First' value to 0 and the 'Last' to 100. You have the option to only load every nth frame if you wish to reduce memory use, if you want every frame from the sequence then set skip to 1, otherwise set how many frames you want to skip.

Next you need to set the Format slider so that the system knows how long the sequence number is in the file name, so for our example the Format would be 4 as the numbers are 4 digits long. Next click the Load Frames button, and then select any frame from the sequence you wish to import and click Open. The system will now show a progress bar as it imports the data. When the data has finished importing you should see the first frame of your sequence in your scene.

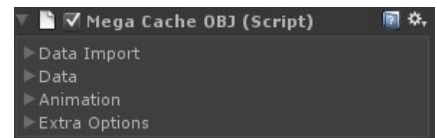
It may be the object is too big or small, each 3d package uses different scales for its scenes but you can compensate for that by changing the Import Scale value much as you would importing any object into Unity. Another thing to note is different 3d packages use different coordinate systems, if you find your object is flipped on the X axis then check the Adjust Coords option and reimport the frames.

One other thing to note is most OBJ exporters will use world coordinates for the vertex values, so it is good practice to centre your object at 0,0,0 in the 3d package before you export the sequence so that the pivot will be in the right place.

## Main Params

### Data Import

This will open up the params section that has the options for the importing of OBJ sequences.



### Data

This will open up the section where you can choose how the data is stored, depending on which Data source you have selected you will see different options in the Data section below.

### Animation

This will open up the Animation options for the imported data, where you can control the playback speed, looping etc.

### Extra Options

Opens up some extra options for the system.

## Data Import Params

### First

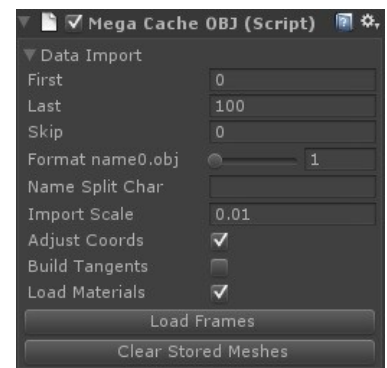
This will set the first frame the system will import from the OBJ sequence.

### Last

This will set the last frame the system will import from the OBJ sequence.

### Skip

You can tell the system to load every nth frame by setting this to a value other than 1, so if you want to load every 4th frame from the sequence then set this to 4. If you want every frame then set it to 1.



### Format

This will tell the system how many digits make up the OBJ sequence filenames, so if your filename is OBJ\_0000.obj then you will need to set the format to 4 to say the names use 4 digits.

### Name Split Char

Sometimes you may have numbers in the main part of the obj sequence filename, if so you can force the system to use the character you place here to split the name part from the frame number in the filename, for example if your sequence is say Anim01\_0001.obj then you would use the '\_' character to tell the system where to split the filename, if your sequence filenames only have digits to describe the frame number then no split char is needed.

### Import Scale

You can control the size of the meshes that are imported by adjusting this value.

### Adjust Coords

Some OBJ files have different coordinate systems, if you find your mesh is flipped on the x axis try changing this value and reimporting the files.

### Build Tangents

If you plan on using shaders which require tangents then you can ask the system to generate tangents for each OBJ file it imports by checking this box, this will obviously increase the memory used by the imported sequence so only turn it on if you really need tangents.

## Load Materials

Turn this on to tell the system to import the mtl files if they are present and to create the materials and textures in Unity, again be careful if you OBJ sequence uses unique materials and textures for every obj file then this could create a lot of assets. The system will do its best to check for duplicates and remove them.

## Load Frames

Click this button to open the file select dialog, select any file from the sequence you wish to import, it does matter which frame it is.

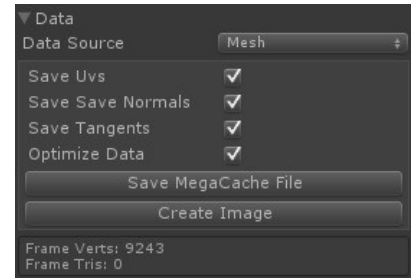
## Clear Stored Meshes

If you have imported meshes then this button will appear, it is an indication that you have imported meshes attached to the object, if you have created a cache file or an image then you no longer need the meshes so click the button to delete them all and free up the memory used.

## Data Params

### Data Source

Depending on the Data Source you select you will have different options in this param section. You have a choice of 3 data sources. Mesh, Image or File. Mesh is the raw Unity Mesh data as it was generated during the import process. Playback of Mesh data is amazingly fast as it is just changing a mesh pointer but you have the overhead of uncompressed mesh data, if your sequences are not long then there is no problem using Mesh data as your source but keep an eye on your memory usage. If it gets too high then you may want to either create a Cache File or an Image. An Image is basically just a cache file stored in memory so again playback is fast and you can use the pre load option if you have more than one thread to speed it up. Memory use for image and cache files is about 1/3 of that of mesh data. File data source will stream the data from a cache file as it is needed, this will result in virtually no memory use and is only a little slower than image for mesh update. Note cache files can not currently be used in Webplayer builds.



## Mesh Data Params

### Save Uvs

Check this to have the uv data saved to the cache or image files.

### Save Normals

Check this to have the normal data saved to the cache or image file.

### Save Tangents

Check this to have the tangent data saved to the cache or image file.

### Optimize Data

Checking this will tell the system to compress and optimize the mesh data in the cache files, this will result in a lot less memory being used at the cost of slightly slower playback and a little in accuracy.

### Save MegaCache File

Click this button to save the data to a MegaCache mgc file.

### Create Image

Click this button will make a memory image of the OBJ sequence data.

### Info

This section will show some stats on the current frame of data as well as the memory use.

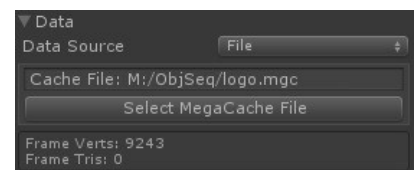
## File Data Params

### Filename

Shows the filename for the current cache file being used.

### Select MegaCache File

Clicking this allows you to select the cache file to be used by the system.



### Info

This section will show some stats on the current frame of data as well as the memory use.

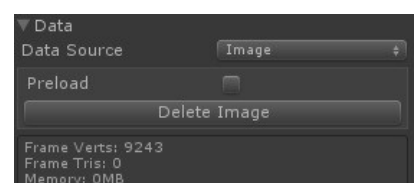
## Image Data Params

### Preload

If you are building for a system that supports threading then this option will appear, it allows you to use another thread to fetch the next frame of data making the playback faster.

### Delete Image

This will delete the image and free its memory.





## Info

This section will show some stats on the current frame of data as well as the memory use.

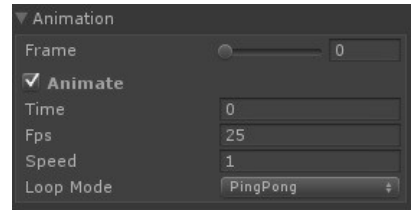
## Animation Params

### Frame

The current frame number of the sequence, if the Animate option is off then you can scrub this value to playback the sequence.

### Animate

If you want your sequence to animate in play mode check this box, the system will then update the frame number for you based on the time and fps settings and will loop the animation using the value set in the loop mode below.



### Time

The current time of the animation playback, if Animation is on you can scrub this value to playback the animation.

### Fps

The fps for the sequence, you should set this to the same value as the 3d editor you exported your sequence from to get the correct length of the animation playback

### Speed

You can alter the speed of the animation playback by changing this value.

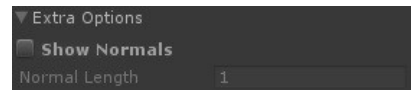
### Loop Mode

You can select the loop mode for the animation from loop, ping pong or clamp.

## Extra Params

### Show Normals

Checking this box will tell the system to display the normals for the mesh.



### Normal Length

The length of the normal that will be shown.

## Videos

This video shows the Object cache system playing back a fluids simulation as well as a fracturing object animation.



This video shows the workflow of exporting a sequence of OBJ files from a 3d package, in this case 3ds Max, and then importing them into Unity using the MegaCache OBJ system.





## MegaCache Obj Ref

The OBJ Ref system allows you to make use of the object cache data that is already being used by a Cache object in the scene, this allows to to have multiple copies of the animated mesh all with different playback options and current frames but without any of the extra memory overhead of storing the cache data for each image, this helps make the Cache system much more useable in your game projects as you could have say one building collapsing or levelling up animation but use it dozens of times in your scene with no extra memory use.

### How to use

Using this system is very easy, just got to the GameObject/Create Other/MegaCache menu and select OBJ Ref, this will create a new gameobject in the scene with all the components required attached, you just then need to click the source value in the inspector and select the MegaCache Object in the scene that has the cache data you wish to use, once selected you will see the mesh in the scene, you can then use the frame slider or animation controls to play back that sequence independently of the original MegaCache object.

## Object Ref Params

### Source

Select the object that already has the cache data you wish to use with this object.

### Frame

The current frame number of the sequence, if the Animate option is off then you can scrub this value to playback the sequence.

### Animate

If you want your sequence to animate in play mode check this box, the system will then update the frame number for you based on the time and fps settings and will loop the animation using the value set in the loop mode below.

### Time

The current time of the animation playback, if Animation is on you can scrub this value to playback the animation.

### Fps

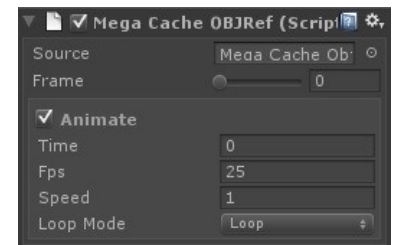
The fps for the sequence, you should set this to the same value as the 3d editor you exported your sequence from to get the correct length of the animation playback

### Speed

You can alter the speed of the animation playback by changing this value.

### Loop Mode

You can select the loop mode for the animation from loop, ping pong or clamp.



## MegaCache Particle

The MegaCache Particle system allows you export particle systems from 3d packages and import them into Unity, you can control the scale, speed and emit rate of the system as it plays back either system wide or per particle. The system works alongside the Shuriken values so you can make use of the features in that to further enhance the imported simulation. The imported data is optimized and compressed to make playback of the simulation very fast, and you can also select a source object to be able to share the cached data so keeping memory use as low as possible. There is also a version of this system for the Legacy particle system in Unity.

### How to Use

To use the particle system you would first have created your simulation in 3ds max or Maya (support for other systems coming soon) and exported it to a cache file using one of our exporters. Next in Unity goto the GameObject/Create Other/MegaCache menu and select the Particle Pro option, this will create a new game object in the scene with the component attached, or you can add the component directly to any object or to an object with a Shuriken Particle system on it.

If you have not attached the component to a particle system then select the one using the Particle System option in the inspector, you should then set the Max Particles value to match that of the particle system you just selected. Depending on the settings you have in the 3d package that created the particle system you may need

to set the Import Scale option, if when you import the particle simulation it is too big or small then adjust the Import Scale value and reimport the file. Click the Import Particle Cache button and select the cache file generated by the exporter, the system will then load all the data. You can change the Import settings if you know you dont need parts of the saved data, so if you just need the positions then uncheck the other options, this will keep the memory use as low as possible.

You are now ready to have the system play back, when you press play the particles will start to play back the saved simulation, if you turn of the emit option on the Shuriken settings and check the Use Emit Rate then you can control the emission of particles from the this component. You can ask the system to pre warm the simulation, and you can adjust the overall scale and play back speed of the simulation as well as the speed and scale for newly emitted particles.

When you are happy with how the particle system works you can click the Save Particle Cache File to save an optimized version of the simulation data for quicker loading or you can load a pre saved file.

It could be that you want to have more than one copy of the particle simulation in your scene, creating a whole new system and importing the same data would waste a lot of memory, so you can just add the component and then select a MegaCache particle system that already exists in the scene, the component will then use the data on that object as well allowing you to have lots of particle system without the memory overhead.

## Params

Description of the params for the MegaCache Particle system.

### Particle System

Use this to select the particle system that will be displaying the simulation, if you have attached the component to an object that has a Shuriken particle system attached then the system will find that if you leave this as None.

### Max Particles

Set this value to the same as the Max Particles setting in the Shuriken inspector.

### Import Scale

This will scale the data as it is imported, if you find the simulation to be too big or small then change this value and reimport the data.

### Data Import Options

Allows you to select which data is imported from the cache file. Position data is always imported.

### Velocity

Import the velocity data, currently velocity data is not used so is not required but future systems may make use of it.

### Rotation

Import the rotation data, Unity can only have a single rotation on its particles so you can pick the axis to use with the Rot Axis below. If not imported then the Shuriken values are used.

### Scale/

*Import the scale data, this is needed to make the particles change size during the simulation, if not imported then the Shuriken values are used.*

### Spin

Import the spin data, not currently used by may be in the future.

### Import Particle Cache

Click this button to open a file select dialog to choose the particle cache file to import.

### Save Data Options

Options for data to save to the cache file.

### Optimize

Check this to tell the system to compress and optimize the data for greatly reduced memory use.

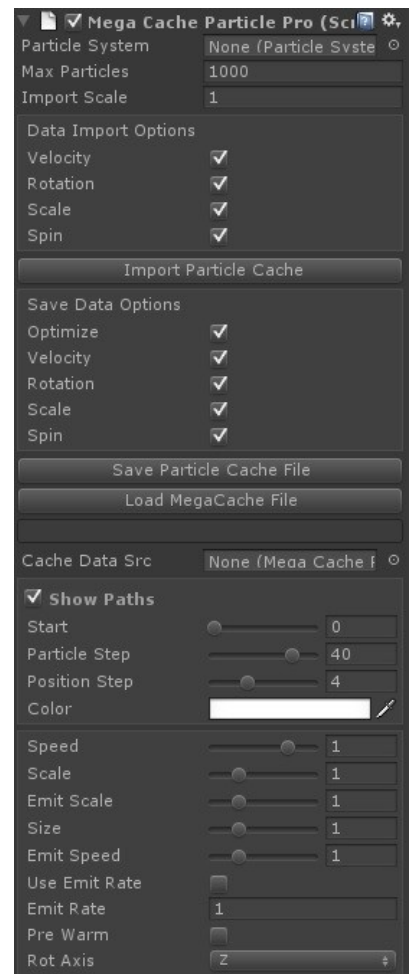
### Velocity

Save the velocity data to the cache file.

### Rotation

Save the rotation data to the cache file.

### Scale



## Scale

Save the Scale data to the cache file.

## Spin

Save the Spin data to the cache file.

## Save Particle Cache File

Click to save the data to a custom cache file, the data will be compressed and optimized if the Optimize setting is checked above.

## Load MegaCache File

Load a pre-saved cache file.

## Info

Some info such as memory use for the currently loaded particle simulation.

## Cache Data Src

You can tell the system to use the cache data that is already loaded to another particle system, select the object here.

## Show Paths

You can ask the system to show the paths of the particles in the system so you can get a feel for the size and look of the system in the editor.

## Particle Step

Showing all the particles could make the editor too slow, so you can ask the system to show every nth particle by changing this value.

## Position Step

When drawing the paths the system will draw every step, this could be too slow so you can adjust the steps between line draws by increasing this value.

## Color

The color of the lines to draw.

## Speed

The overall speed of the playback of the simulation, the higher the value the faster the playback, 0 will freeze the simulation and a negative value will play the simulation backwards.

## Scale

The overall scale of the particle system, you can at any time change the scale of the entire particle system by changing this value.

## Emit Scale

It maybe that you want the adjust the scale for only newly emitted particles, say for smoke it can start of small and then get larger over time, changing this value will effect the scale for any new particles emitted.

## Size

This will change the size of the particles only.

## Emit Speed

You can change the speed of the playback for any newly emitted particles by changing this value.

## Use Emit Rate

If you turn of the emission on the particle system you can control the emit rate by turning this on and changing the Emit Rate value below, with this off then the emission will be controlled from the Particle system.

## Emit Rate

How many particles a second to emit.

## Pre Warm

Tell the system to pre warm the particles so they will appear in the scene at start instead of starting from nothing.

## Rot Axis

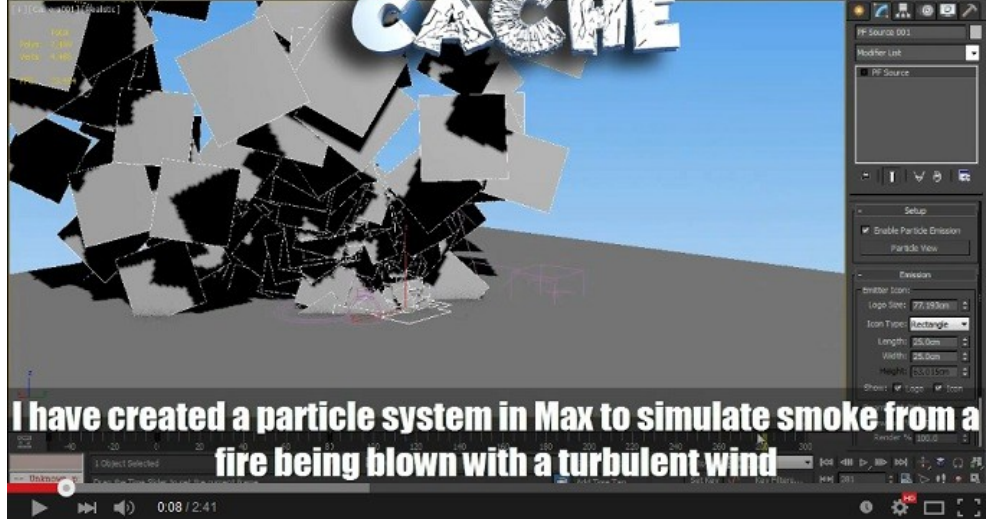
The axis to use in the rotation value.

## Video

This video shows the workflow of exporting a Particle Flow simulation from 3DS Max and then importing that data into MegaCache and playing it back.







## MegaCache Particle Legacy

The MegaCache Particle system allows you export particle systems from 3d packages and import them into Unity, you can control the scale, speed and emit rate of the system as it plays back either system wide or per particle. The system works alongside the Legacy particle values so you can make use of the features in that to further enhance the imported simulation. The imported data is optimized and compressed to make playback of the simulation very fast, and you can also select a source object to be able to share the cached data so keeping memory use as low as possible. There is also a version of this system for the Shuriken particle system in Unity.

## How to Use

To use the particle system you would first have created your simulation in 3ds max or Maya (support for other systems coming soon) and exported it to a cache file using one of our exporters. Next in Unity goto the GameObject/Create Other/MegaCache menu and select the Particle Pro Legacy option, this will create a new game object in the scene with the component attached, or you can add the component directly to any object or to an object with a Shuriken Particle system on it.

If you have not attached the component to a particle system then select the one using the Particle System option in the inspector, you should then set the Max Particles value to match that of the particle system you just selected. Depending on the settings you have in the 3d package that created the particle system you may need to set the Import Scale option, if when you import the particle simulation it is too big or small then adjust the Import Scale value and reimport the file. Click the Import Particle Cache button and select the cache file generated by the exporter, the system will then load all the data. You can change the Import settings if you know you dont need parts of the saved data, so if you just need the positions then uncheck the other options, this will keep the memory use as low as possible.

You are now ready to have the system play back, when you press play the particles will start to play back the saved simulation, if you turn of the emit option on the Shuriken settings and check the Use Emit Rate then you can control the emission of particles from the this component. You can ask the system to pre warm the simulation, and you can adjust the overall scale and play back speed of the simulation as well as the speed and scale for newly emitted particles.

When you are happy with how the particle system works you can click the Save Particle Cache File to save an optimized version of the simulation data for quicker loading or you can load a pre saved file.

It could be that you want to have more than one copy of the particle simulation in your scene, creating a whole new system and importing the same data would waste a lot of memory, so you can just add the component and then select a MegaCache particle system that already exists in the scene, the component will then use the data on that object as well allowing you to have lots of particle system without the memory overhead.

## Params

Description of the params for the MegaCache Particle system.

### Particle System

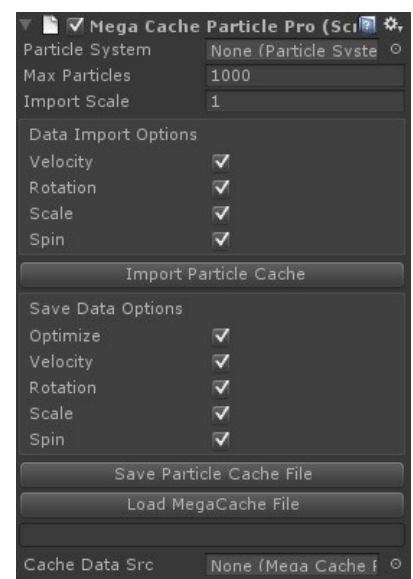
Use this to select the particle system that will be displaying the simulation, if you have attached the component to an object that has a Legacy particle system attached then the system will find that if you leave this as None.

### Max Particles

Set this value to the same as the Max Particles setting in the Legacy inspector.

### Import Scale

This will scale the data as it is imported, if you find the simulation to be too big or small then change this value and reimport the data.



### Data Import Options

Allows you to select which data is imported from the cache file. Position data is always imported.

### Velocity

Import the velocity data, currently velocity data is not used so is not required but future systems may make use of it.

### Rotation

Import the rotation data, Unity can only have a single rotation on its particles so you can pick the axis to use with the Rot Axis below. If not imported then the Shuriken values are used.

### Scale/

*Import the scale data, this is needed to make the particles change size during the simulation, if not imported then the Shuriken values are used.*

### Spin

Import the spin data, not currently used by may be in the future.

### Import Particle Cache

Click this button to open a file select dialog to choose the particle cache file to import.

### Save Data Options

Options for data to save to the cache file.

### Optimize

Check this to tell the system to compress and optimize the data for greatly reduced memory use.

### Velocity

Save the velocity data to the cache file.

### Rotation

Save the rotation data to the cache file.

### Scale

Save the Scale data to the cache file.

### Spin

Save the Spin data to the cache file.

### Save Particle Cache File

Click to save the data to a custom cache file, the data will be compressed and optimized if the Optimize setting is checked above.

### Load MegaCache File

Load a pre-saved cache file.

### Info

Some info such as memory use for the currently loaded particle simulation.

### Cache Data Src

You can tell the system to use the cache data that is already loaded to another particle system, select the object here.

### Show Paths

You can ask the system to show the paths of the particles in the system so you can get a feel for the size and look of the system in the editor.

### Particle Step

Showing all the particles could make the editor too slow, so you can ask the system to show every nth particle by changing this value.

### Position Step

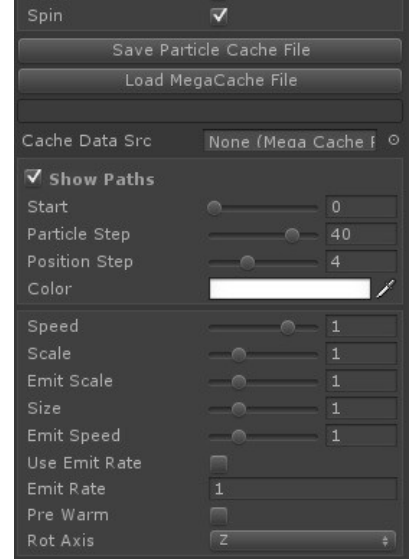
When drawing the paths the system will draw every step, this could be too slow so you can adjust the steps between line draws by increasing this value.

### Color

The color of the lines to draw.

### Speed

The overall speed of the playback of the simulation, the higher the value the faster the playback, 0 will freeze the simulation and a negative value will play the simulation backwards.



## Scale

The overall scale of the particle system, you can at any time change the scale of the entire particle system by changing this value.

## Emit Scale

It maybe that you want the adjust the scale for only newly emitted particles, say for smoke it can start of small and then get larger over time, changing this value will effect the scale for any new particles emitted.

## Size

This will change the size of the particles only.

## Emit Speed

You can change the speed of the playback for any newly emitted particles by changing this value.

## Use Emit Rate

If you turn of the emission on the particle system you can control the emit rate by turning this on and changing the Emit Rate value below, with this off then the emission will be controlled from the Particle system.

## Emit Rate

How many particles a second to emit.

## Pre Warm

Tell the system to pre warm the particles so they will appear in the scene at start instead of starting from nothing.

## Rot Axis

The axis to use in the rotation value.

## Video

This video shows the workflow of exporting a Particle Flow simulation from 3DS Max and then importing that data into MegaCache and playing it back.

