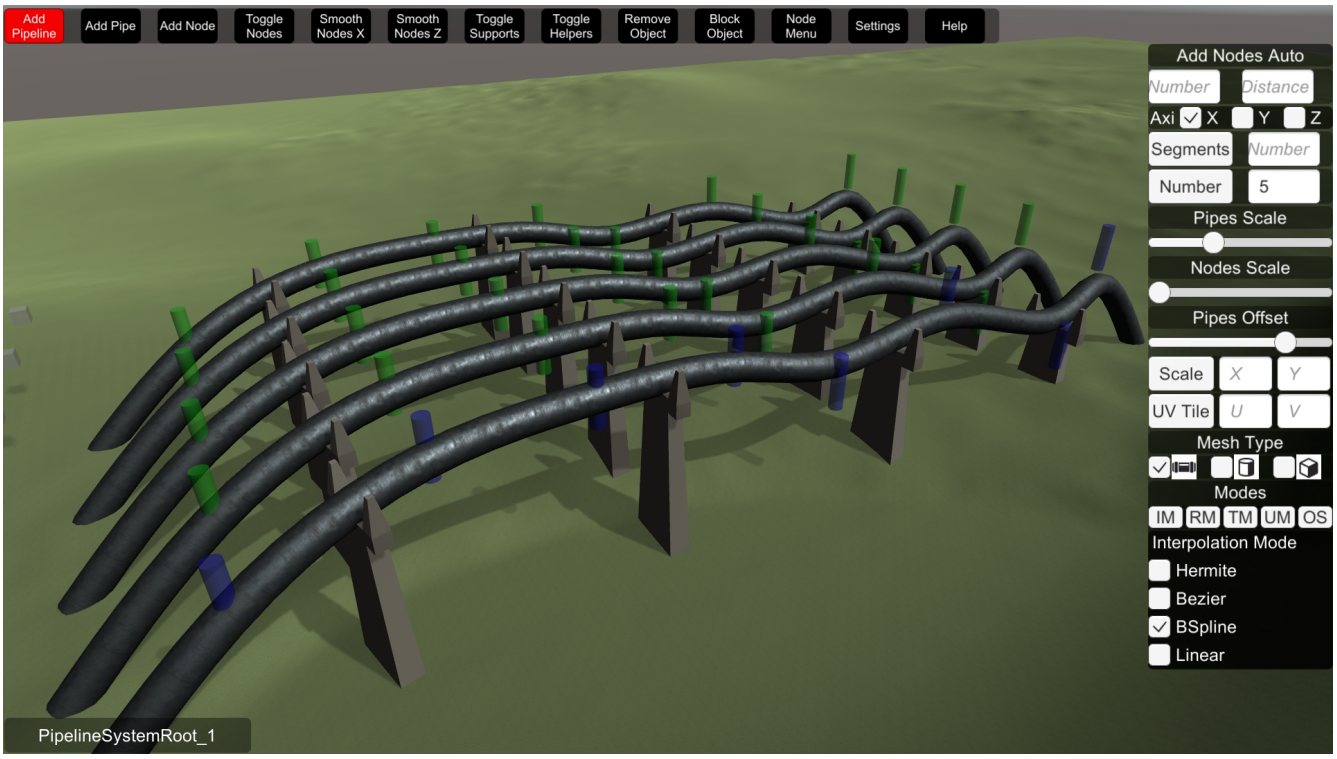
Procedural Pipeline Toolkit

by

Tomza



INTRO

Thank you for purchasing the toolkit. This product is designed for using with your games (also commercial) and isn't a complete project. The scripts and other assets in the package can be modified according to your needs.

PROCEDURAL PIPELINE TOOLKIT

With this toolkit, you can make a pipelines/pipes of a desired size/scale/diameter, etc. very easy at runtime. The default number of pipes in a pipeline is one, but you can increase the number according to your needs. No limitation for maximum, though. This toolkit was tested with 100 pipes in one pipeline. Editing procedural pipes can work both in the Editor and at runtime. The toolkit is made for any games that need to have industrial pipeline systems.

QUICK START

1/ Start the **PipelineTest** scene (**ProceduralPipelineToolkit** -> **Scene**)
2/ Spawn a pipeline helper (a cylinder)
3/ Make sure the helper is selected (black)
3/ Open the **Settings Panel** to manipulate properties
4/ Spawn pipes with the set up you want (the first spawned pipe with blue nodes is master, which other pipes are attached to)
5/ You can use a fly camera to fly around and see the pipeline.
6/ Manipulate the pipes using nodes to get a desired view
7/ You can block some nodes up (each node has its own node menu) on a pipe to make them move with a master node

INPUT

Q - quit game
Esc - hide/show cursor
holding RMB + X/Y axis - rotate camera
WSAD/Arrows - camera navigation

holding Shift - camer moves faster
holding Ctrl - camera moves slower
Delete - delete a selected object
RMB click on a node - toggle a node menu for blocking and unblocking nodes in one pipe
P - spawn a pipeline helper (a cylinder)
Space - spawn pipes when a pipeline helper (a cylinder) has been spawned

SCRIPT REFERENCE

The created mesh is based on a spline that is created of curves. Each spline is a separated pipe. Some scripts are for creating meshes of a pipe (they have a word “spline” in their names) and others are for control pipes and pipelines (they have a word “controller” or “root” in their names). And the rest are only helper scripts or just typical .NET classes, containers for data or static utility classes. The control and root scripts can be used for adding more features for pipes and pipelines or manipulating their values. It is recommended to use them instead of changing .NET classes. The important script is **PipelineGameManager** that is a singleton. It is an intermediary code between all control/root scripts and other scripts as **UIManager**. You must have one on your scene. The **UIManager** is a good example how to manipulate pipes and pipelines (changing their values) in a game.

CONTACT

Tomasz Zackiewicz
pogtoma@gmail.com

CHANGELOG

Version numbers:

v. 1.0:
- Base version
- All fundamental behaviours added