

# NURBS - Documentation

## [CREDITS]

Developed by Unity Asset Store Publisher: Dozed  
<https://assetstore.unity.com/publishers/28998>

Contacts:

Website: <https://github.com/Dozed12>

Email: [frantomor@gmail.com](mailto:frantomor@gmail.com)

## [SUMMARY]

Generate NURBS (Non-Uniform Rational B-Splines) with ease for complex procedural mesh generation. Ideal for runtime mesh generation, terrain generation, precision modelling, etc

## [FEATURES]

- Curve Degree: Number of control points influencing each position. (1 - linear; 2 - quadratic; 3 - cubic, etc)
- Surface Resolution: Use more or fewer polygons to describe the curve with more or less detail.
- Knot Vector: Customizable knot vector for advanced uses.
- Surface Sampling: Sample surface coordinates with mathematical precision.
- Editor and Runtime: Generate NURBS at runtime and in-editor.
- Runtime Editing: Optimized for runtime generation.

## [USAGE]

1. Setup control points as a 2d array of `NURBS.ControlPoint` each with coordinates and weight
2. [Optional Advanced Use] Setup custom knot vectors for the curvatures
  - a. Knot vectors are arrays of floats with length = number of Control Points + degree + 1
  - b. `GenerateKnots(...)` can be used to generate basic progressive knot vectors with optional start and end clamps
3. Build the NURBS surface with `NURBS.Surface(...)` specifying the control points, degree in both axis and optional knot vectors. If knot vectors are not provided they will be automatically generated using `GenerateKnots(...)` with start and end clamps.
4. [Optional] Change the control points of the surface by modifying its `controlPoints` 2d array. Change the degrees of the surface using `DegreeU(...)` and `DegreeV(...)` with optional custom knot vectors.
5. Build the surface mesh using `BuildMesh(...)` providing the surface resolution in both axis and the number of polygons used.

- a. A mesh can be passed as a parameter in which case it will be used to generate the mesh. If a mesh is not provided a new one will be created and returned. Reusing the mesh in multiple BuildMesh(...) saves memory allocation from mesh creation.

The coordinates of any position in the surface can be sampled using GetPoint(...) which receives the two UV coordinates [0, 1] and returns a Vector3 with the coordinates relative to the control points origin.

The sample scene includes examples of:

- Surfaces with the same control points but different degree
- Surface control point editing at runtime as a sine wave
- Cylinder Surface with exact representation using custom knot vectors
- Surface editing in editor

For help with the various concepts involving NURBS check the wikipedia page at [https://en.wikipedia.org/wiki/Non-uniform\\_rational\\_B-spline](https://en.wikipedia.org/wiki/Non-uniform_rational_B-spline) and this useful browser NURBS calculator <http://nurbscalculator.in/>

### **[PERFORMANCE]**

NURBS performance depends primarily on the number of surface samples performed on Mesh generation. This is dependent on the degree of the curve - causing more or fewer control points to influence the calculation - and on the mesh resolution - directly controlling the number of samples performed.

### **[CLOSING]**

Thank you for purchasing this asset.

If you need any help, support or even have suggestions or requests feel free to contact me by email at [frantomor@gmail.com](mailto:frantomor@gmail.com)