



3Data File

XML Specifications

Table of Contents

Background Information	3
Managing 3Data Files	3
Creating 3Data Files	3
3Data File Structure	3
3Data Plot Container	3
Geospatial	6
Basic Reference Map	6
NASA Map	6
ArcGIS Map	7
Dataset Connectors	8
REST API Dataset Connector	8
Assets	8
3D Models	8
Images: PNG and JPG	8
PDFs:	8
Videos: Mp4	9
Lines	9
Animations	9
Position	9
Opacity	9
Scale	9

3Data XML File Specification

Background Information

The 3Data XML file format is a proprietary file format which defines all of the details for a 3Data presentation. The file contains the slide types, the data set mappings, and the look and feel of the slides. You can copy the presentation from one room into another room using this file.

Important: When editing the file, make sure that the data sets which are referred to by the file are also present in the room where the presentation is located. The slide will not load if the data file is not loaded in that room.

Managing 3Data Files

Because 3Data files are simply XML, you can check them into your source code repository and it will track your changes over time. This practice is recommended if you want to be able to refer back to previous version of the same presentation.

Creating 3Data Files

There are two primary ways of creating these files:

- You can use the 3Data GUI to create a new presentation, and then edit the file after downloading it from <https://portal.3data.io/>.
- You can also create the 3data file using any text editor from scratch. Starting with an existing 3data file is recommended, however.

3Data File Structure

The 3data files always start with the presentation version and a first slide tag:

```
<presentation version="1.3">  
<slide>
```

Every slide is going to begin with a <slide> tag and end with a </slide> tag.

3Data Plot Container

The next section defines the mappings of the plot container to the underlying data set. In this exam-

ple, all of the aesthetic-mappings are mapped to various columns of the sample data set. To point these to a different column, just change the text below to the desired columns:

```
<plot-container aesthetic-mappings="x:X Axis;y:Y Axis;z:Z Axis;shape:Shape Axis;-size:Size Axis;color:Color Axis;spin:Rotation Axis"
```

The Theme is defined next. Color refers to the color of the shapes in the legend on the left side of the plot. Changing fontColor and titleColor will change the colors of the axis labels on the plot. The default colors are created in 3-digit hex color notation, but six digit colors are also supported:

This is also where you can edit the default point size. To do so, add `;size:0.01` inside the quotes. The default size is 0.01, a value between 0.005 and 0.02 is recommended.

```
theme="color:#AAA;fontColor:#FFF;titleColor:#FFF;font:./assets/Roboto-Black.json;fontShader:msdf;fontScale:1;titleScale:1.2;size:0.01"
```

The position of the graph within the room is defined next, along with the transparency and opacity of the graph itself:

```
position="0 1.6 0" rotation="0 45 0" scale="1 1 1" coordinates-cartesian="padding:0.05" background-color="#777" transparent="true" opacity="0.5" presentation-object-id=""
```

The dataset which is tied to the graph along with the scale of the axes is defined next. If you wanted one of your axes to be logarithmic instead of linear, simply replace `scale:linear` with `scale:log`:

```
dataset="Sample Scatterplot" scale-position-x="scale:linear" scale-position-y="scale:linear" scale-position-z="scale:linear" scale-shape="platonic" scale-size="1, 3" scale-color="Blues" scale-spin="0, 6.28">
```

This section will highlight the data point that you point to with the cross-lines so you can see what you are selecting more clearly:

```
<annotate-crosslines-interactive mod-animate-transition="duration:1000;ease:default" aesthetic-mappings="" line-color="#FFFFFF" interaction-highlight="enabled:true" presentation-object-id=""></annotate-crosslines-interactive>
```

This section lets you see the interactive panel when you click on a data point:

```
<annotate-panel-interactive mod-animate-transition="duration:1000;ease:default"
aesthetic-mappings="" text-color="#FFFFFF" characters-per-line="25" dimen-
sions="0.6 0.4" panel-alignment="top-left" panel-margin="0.025" interaction-high-
light="enabled:true" presentation-object-id=""></annotate-panel-interactive>
```

We are then going to define the tick marks on the graph itself. Skipping this section will leave your graph with no tick marks:

```
<guide-axis theme="" axis="x" title="" tick-limit="15" presentation-ob-
ject-id=""></guide-axis>
<guide-axis theme="" axis="y" title="" tick-limit="15" presentation-ob-
ject-id=""></guide-axis>
<guide-axis theme="" axis="z" title="" tick-limit="15" presentation-ob-
ject-id=""></guide-axis>
```

This section defines the legends which are displayed on the left side of the graph. Skipping this section will leave your graph with no legend for shape, color, size, or rotation:

```
<guide-legend theme="" aesthetic="shape" title="" tick-limit="10" presenta-
tion-object-id=""></guide-legend>
<guide-legend theme="" aesthetic="color" title="" tick-limit="10" presenta-
tion-object-id=""></guide-legend>
<guide-legend theme="" aesthetic="size" title="" tick-limit="10" presentation-ob-
ject-id=""></guide-legend>
<guide-legend theme="" aesthetic="spin" title="" tick-limit="10" presentation-ob-
ject-id=""></guide-legend>
```

This next section defines which graph type is going to be displayed. `Layer-point` refers to a 3d scatter-plot, while `layer-surface` would refer to a surface plot:

```
<layer-point aesthetic-mappings="" theme="" dataset="Sample Scatterplot" presen-
tation-object-id="" mod-animate-transition="duration:1000;ease:default" interac-
tion-select="index:-1"></layer-point>
```

Close the plot container to and close the slide. Repeat this process as for as many slides as you would like in your presentation:

```
</plot-container>
</slide>
```

After all of your slides are created, end the whole presentation:

```
</presentation>
```

Geospatial

Basic Reference Map

reference-map will display an equirectangular map projection as a surface in the plot. It will transform the plot x, y, and z to latitude, height, and longitude.

```
<reference-map mod-animate-transition
map="https://upload.wikimedia.org/wikipedia/commons/e/ea/Equirectangular-projection.jpg"></reference-map>
```

Attribute	Description	Required
map	URL of equirectangular map projection raster image.	Yes
height	Height scale override.	No
longitude-extent	Overrides reference grid transformation for textureDomainZ. For a whole Earth equirectangular projection: [-180, 180]	No
latitude-extent	Overrides reference grid transformation for textureDomainX. For a whole Earth equirectangular projection: [-90, 90]	No

NASA Map

Requests map imagery for geospatial visualizations from NASA's Global Imagery Browse Services (GIBS). The GIBS imagery archive includes approximately 900 imagery products representing visualized science data from the NASA Earth Observing System Data and Information System (EOSDIS). More information about available map layer names can be found in NASA's GIBS documentation: <https://wiki.earthdata.nasa.gov/display/GIBS/GIBS+Available+Imagery+Products> Detailed developer documentation: [https://wiki.earthdata.nasa.gov/display/GIBS/GIBS+API+for+Developers#GIBSAPIfor-Developers-OGCWebMapService\(WMS\)](https://wiki.earthdata.nasa.gov/display/GIBS/GIBS+API+for+Developers#GIBSAPIfor-Developers-OGCWebMapService(WMS))

```
<nasa-map mod-animate-transition
map-name="BlueMarble_ShadedRelief_Bathymetry"
map-time="2018"-10-01
backup-texture="https://upload.wikimedia.org/wikipedia/commons/e/ea/Equirectangu-
lar-projection.jpg"
"></nasa-map>
```

Attribute	Description	Required
map-name	NASA GIBS map layer name or comma separated list of map layer names.Defaults to: BlueMarble_ShadedRelief_Bathymetry	No
height	Height scale override.	No
map-time	Requested time of NASA GIBS map imagery. If omitted, most recent image will be used. Uses ISO 8601 date format. For example: 2018-10-01	No
backup-texture	URL of equirectangular map projection raster image to use if NASA imagery cannot be loaded.	No

ArcGIS Map

Experimental. Requests map imagery for geospatial visualizations from an ESRI ArcGIS Server Feature Service.

```
<arcgis-map mod-animate-transition
map-server-url="https://services.arcgisonline.com/arcgis/rest/services/World_
Street_Map/MapServer"
backup-texture="https://upload.wikimedia.org/wikipedia/commons/e/ea/Equirectangu-
lar-projection.jpg"></arcgis-map>
```

Attribute	Description	Required
map-server-url	Specifies the public URL of the ArcGIS Feature Service. Defaults to: https://services.arcgisonline.com/arcgis/rest/services/World_Street_Map/MapServer	Yes
height	Height scale override.	No
backup-texture	URL of equirectangular map projection raster image to use if ArcGISimagery cannot be loaded.	No

Dataset Connectors

REST API Dataset Connector

Experimental. Converts existing JSON-based REST API into zues.gl dataset. Nested objects in response will be flattened (one level only).

```
<rest-dataset
title="SpaceX Rockets"
url="https://api.spacexdata.com/v3/rockets"
></rest-dataset>
<plot-container
dataset="SpaceX Rockets"
></plot-container>
```

Attribute	Description	Required
title	The zues.gl title for the dataset. Needs to exactly match the value set in <plot-container dataset="">	Yes
url	The URL of the JSON REST API endpoint.	Yes
refresh	Number of seconds to wait before re-requesting the JSON REST API. By default, data will not be refreshed.	No
headers	JSON string of headers to pass along with REST API request. For example: headers="{ 'x-my-api-key': 'abc123' }"	No
limit	Number of items to limit in returned response.	No

Assets

There are also a number of assets that you can bring into the presentation. These assets must be dragged into your room prior to updating the 3Data XML file.

3D Models

GLB is the supported 3D file type. You may also have animations baked into your 3D file, which will automatically play.

```
<gltf-model-3data model="model.glb" presentation-object-id="model"
position="0 0 0" rotation="0 0 0" scale="1 1 1" animation-mixer=""></gltf-mod-
el-3data>
```

Images: PNG and JPG

```
<image-plane src="image.png" presentation-object-id="image"
geometry="primitive: plane; width: 1; height: 1"
position="0 0 0" scale="1 1 1" rotation="0 0 0"></image-plane>
```

PDFs:

```
<pdf-plane document="doc.pdf" presentation-object-id="doc" page="1"
position="0 0 0" scale="1 1 1" rotation="0 0 0"
width="1" height="1"></pdf-plane>
```

Videos: Mp4

```
<network-video-player video="video.mp4" presentation-object-id="video"
position="0 0 0" scale="1 1 1" animation-mixer=""></network-video-player>
```

Lines

```
<a-entity line="start: 0 0 0; end: 1 1 1; color: white; opacity: 1"></a-entity>
```

Animations

Each asset and plot have the ability to add animations with different properties such as position, scale, opacity, and rotation. Animations for slide in and slide out are defined within the asset properties. Dur defines the animation duration. Delay defines the time before the animation starts.

Position

```
animation__in="property: position; from: 0 0 0; to: 0 1.3 0; dur: 500;delay:500"
```

Opacity

```
animation__in="property: model-opacity.alpha; from: 0; to: 1; dur: 500"
```

Scale

```
animation__in="property: scale; from: .1 .1 .1; to: 1 1 1; dur: 500"
```

For multiple animations, add a second line starting with:

```
animation__in2=
```

Out animations need to include the startEvent

```
animation__out="startEvents:ev-3data-presentation-transition-out; property: model-opacity.alpha; from: 1; to: 0; dur: 1500"
```