

JSON Formats

The JSON (Javascript Object Notation) format is an incredibly flexible file format that is useful for web-based graphing and presentations. JSON is a human-readable format and is typically used to pass data from a web server to a javascript program in a web browser. Although the JSON format is not specific to motion capture or biomechanics, Visual3D offers limited support for both reading and writing the format.

Reading JSON Files

The only JSON file format currently supported to be read in by Visual3D is the markerless motion capture format used in Maple Leafs Sports and Entertainment's [SPL Open Data repository](#). Visual3D will automatically build a model for this data as it does for other markerless motion capture files.

Important Notes

Some motion capture vendors, such as [Qualisys](#) and [Theia](#), allow users to export motion capture data to the JSON format. In the case of following on processing in Visual3D, however, we recommend that users export their motion capture data to the [C3D format](#) instead.

Writing JSON Files

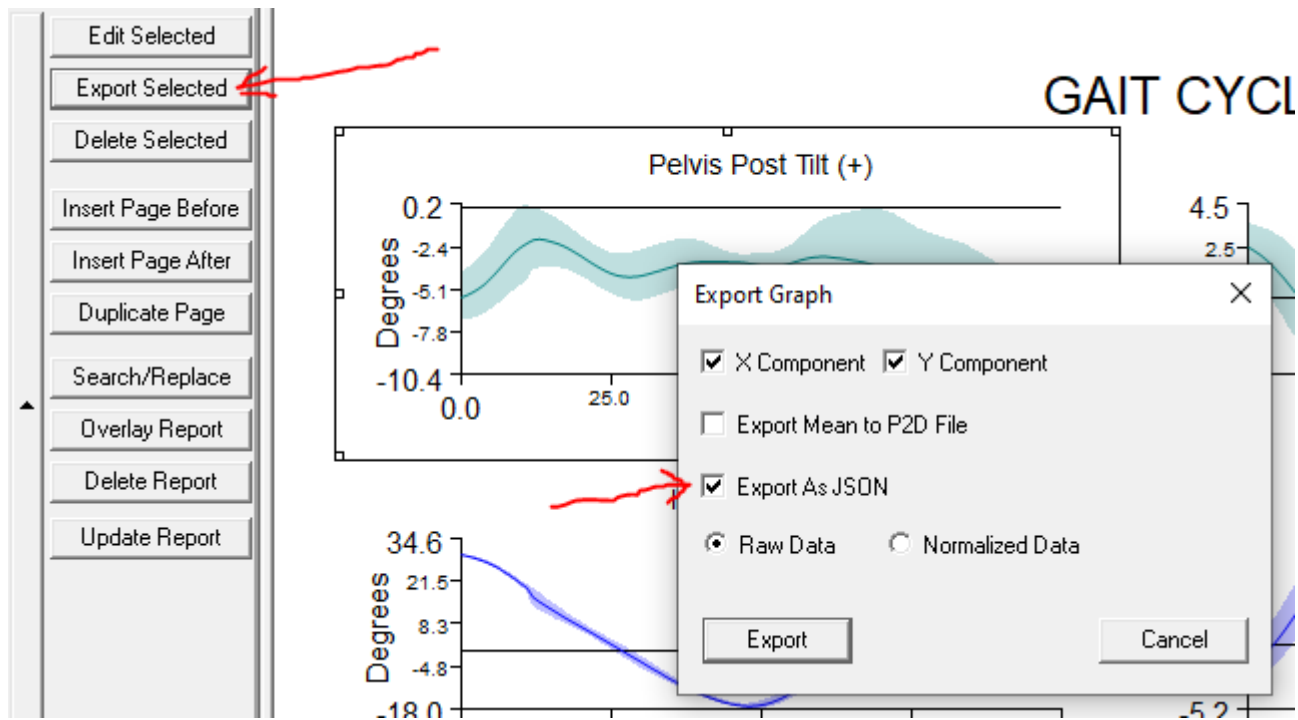
Visual3D can produce a JSON output file that can then be parsed and used by a javascript, ES6, or PHP program for use on the web. This option is only available in Visual3D version 2020.06.01 and later.

The format of the export is different depending on where in Visual3D you generate the export.

Example 1: Report Graph Export

This example demonstrates how to export data from a report graph to JSON format. This requires that you first create your report and graphs ([Documented here](#)).

1. Select the report you want to export and select the JSON output option from the pop-up dialog.



2. If the mean and standard deviation were displayed in the graph, they will be in the exported data too. The JSON format for mean and std dev are similar to the normal ASCII output in that the 2 X-components are listed first, and then the 2 Y components. The graphed data is listed in X and Y sections together.

The JSON output is a single array for the graph, with sub-arrays for signal information and for data.

```
{
  "title": "graph_title_here",
  "graphdata": [
    {
      "filename": "C:/Visual3D/Data/Full Body Gait/FullBodyWalking.c3d",
      "type": "FRAME_NUMBERS",
      "folder": "Original",
      "name": "TIME",
      "X_axis_label": "Time",
      "frames": 112,
      "data": [0.933333, 0.941667, ...1.125000]
    },
    {
      "filename": "C:/Visual3D/Data/Full Body Gait/FullBodyWalking.c3d",
      "type": "LINK_MODEL_BASED",
      "folder": "Original",
      "name": "RPV_ANGLE",
      "Y_axis_label": "Degrees",
      "frames": 112,
      "data": ["no_data", "no_data", -1.811149, -1.873471, ...]
    }
  ]
}
```

Example 2: Export Data As ASCII Command

The more general option is to use the [Export Data To ASCII File](#) pipeline command.

1. Simply check the JSON format checkbox in the dialog or un-comment and change the parameter USE JSON FORMAT = TRUE from the textual interface.

The resulting JSON output format for this export is:

```
{ "Visual3D" : [
  { "filename" : "Mean"
    "type" : "LINK_MODEL_BASED",
    "folder": "ORIGINAL",
    "name" : "LFT_ANGLE",
    "frames" : "101",
    "signal" : [
      { "component" : "X", "data" : [0, ...,null ]},
      { "component" : "Y", "data" : [-7.00954, ...,null ]},
      { "component" : "Z", "data" : [16.03672, 13.68475, ... ]}
    ]
  }
]}
```

From:

<https://www.wiki.has-motion.com/> - Wiki Documentation Site

Permanent link:

https://www.wiki.has-motion.com/doku.php?id=visual3d:documentation:visual_3d_formatting:json_formats

Last update: 2026/03/26 15:20

