

Language: [English](#) • [français](#) • [italiano](#) • [português](#) • [español](#) ****

==== Contents ====

- * [1 Downloads and Relevant Links](#)
- * [2 Build CMZs for Hitting Data](#)
- * [2.1 Refining Hitting Data](#)
- * [3 Build Your Own Metadata File](#)
- * [3.1 Using Metadata to Refine Queries](#)
- * [3.2 Visualizing Refined Queries](#)
- * [4 Further Reading](#)

Through the OpenBiomechanics Project, Driveline Baseball has made elite-level motion capture data publicly available. The data can be downloaded from Driveline's GitHub repository.

Having access to large datasets is a great asset, but processing these large datasets can take time (and patience). Inspect3D is designed to streamline this process. The steps in this tutorial describe how to create CMZ files from the pitching and batting data sets, then inspect and group the data in Inspect3D.

Downloads and Relevant Links

To follow along with this tutorial, ensure that your version of Inspect3D is at least [v2023.08.08](#). Earlier versions will not have the Build CMZ feature. After verifying that you have the correct version of Inspect3D, download the following two zip files.

1. [OpenBiomechanics Project: Driveline Baseball data](#)
2. [C-Motion Specific Files](#)

Please note, these downloads contain a large amount of data. Building CMZs is a slow process, so if you want to speed it up to follow along with building CMZ portion of this tutorial cut the number of datasets under openbiomechanics-main down to 3-5 subjects.

If you want to skip the Build CMZ step of this tutorial and focus on the metadata applications, you can:

- download [this](#) pre-built CMZ workspace and skip to [Refining Hitting Data](#)
- skip to the [Build Your Own Metadata File](#) example.

For more technical information about loading metadata, see: [Metadata Documentation](#)

Build CMZs for Hitting Data



In the Build CMZ dialog:

1. **Set the path to:** C:\...\openbiomechanics-main\baseball_hitting\data\c3d

2. **Set the Metadata File to :** C:\...\openbiomechanics-main\baseball_hitting\data\metadata.csv
3. **Set the MDH File to:** C:\...\openbiomechanics-main\baseball_hitting\code\v3d\model\hitting_v1_model_hybrid.mdh
4. Add the following scripts and ensure they are in the correct order:
 1. C:\...\CMotion Files\Filter.v3s
 2. C:\...\CMotion Files\Events.v3s
 3. C:\...\openbiomechanics-main\baseball_hitting\code\v3d\CMO.v3s
5. Click **Create CMZs**
6. A metadata dialog box will appear after clicking Create CMZs. Make the following changes in this box:
 1. Height Units: in
 2. Weight Units: Lb
 3. session_swing: Dynamic Trial Identifier
 4. user: Static Trial Identifier
 5. Check "Subject Specific" next to highest_playing_level, hitter_side, bat weight/length, and Click "Apply".
7. CMZ files may take a while to build. Check the status bar in the bottom left hand corner of the interface for the build status. After the files are built, check that the Library Path dialog is set correctly and hit "Load" in the Load Library dialog.

Refining Hitting Data

From here on it is assumed that you either have: a) built CMZs from scratch using the hitting data set, or b) downloaded the pre-made CMZs linked above.

The zip file openbiomechanics-main has a metadata file for the hitting data set which you can view at your convenience. This metadata file has a number of categories including player age, bat weight, hitting side etc.

Let's say we want to query this data set, but we **only** want data from subjects that are under age 20.

We start by loading library with the hitting data set, navigating to the Query Builder dialog, and opening the Auto-Populate dialog. Ensure that "Baseball" is checked. Find the "Refine using signal" checkbox and click "Add". Then you can enter your query refinements.



When refining queries it can be helpful to have the metadata file open, so that you can see the values it contains. Note that in this case, accessing athlete_age has been set as a global metric, and you will not be able to find the meta folder unless Global Value is checked.

The result of this procedure is that data is queried only if it meets this refinement category, and the refinement name will be applied as a suffix to the end of the signal name.

Build Your Own Metadata File

Let's say you have three subjects worth of data where each subject walks at four different speeds and

has a different weight.

In this example let us say we want to include metadata on the subject's walking speed and weight. To create a .csv file for this metadata, start by creating a row of headers. The minimum requirements for a metadata file include the static file name and dynamic file name. After that, we can add the other parameters we are interested in.

In the example below, we can see that there are four headers (static_model, dynamic_file, walking_speed and subject_mass).



Here you can edit the description fields and subject specific fields. After you are satisfied with the fields, click "Apply" to build CMZs and load the library.

Using Metadata to Refine Queries

Once your data is loaded it can be queried. This example shows how you can use metadata to refine queries.

In this instance, we have data where three subjects walk at four different speeds. The subjects also have their weight recorded.

If you navigate from the Query Builder dialog to the Auto-Populate menu, there is an option called "Refine Using Signal". Click to enable this checkbox, and then press "Add". In the image below, I have loaded a pre-built auto-settings file, but also have the signal refinement dialog open.



Since I want to refine my data by walking speed in this example, I name the refinement "3.0", choose the data under METRIC>meta>walking_speed>X and indicate that the value must be "Equal to" 3.0. I can then save my refinements as an .xml file which can be loaded at any time if I want to re-query my library.

Press save and then create and close the auto-populate window. Back in the query builder you should see queries that reflect the refinements you just made.



Visualizing Refined Queries

In this example I have imported the X, Y and Z components of the ASIS marker trajectories for the subjects walking at four different speeds.

We can now use the refined queries to visualize different relationships between speed and subject kinematics, as seen in the example below:



In this example we can see how the trajectory for the ASIS markers changes as walking speed

changes for Subject 1.

Further Reading

In this tutorial, we reviewed how to refine data queries using metadata from OpenBiomechanics open source dataset. We also reviewed how to generate a metadata file from scratch and use it to refine data queries.

If you want to review another example of building CMZs with OpenBiomechanics data, visit the [Building CMZ Tutorial](#).

Retrieved from ""

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

Permanent link:
<https://has-motion.com/wiki/doku.php?id=inspect3d:tutorials:metadata&rev=1718643895>

Last update: **2024/06/17 17:04**

