Documentation for Handshake (Dev. Level)

Introduction:

In order for the data warehouse to contain a client's data, the warehouse must know the format of the data being sent (its types, names, formats, and such). To achieve this, the client must participate in our Handshake Protocol. The following document explains what the Handshake protocol is, how it acts, and what it needs.
Disclaimer: This method of using the Handshake Protocol is for people who have a basic knowledge of the JSON standard.

Abstract:

The Handshake Protocol takes a request from the client in the form of a JSON file. The data warehouse takes that JSON file and creates a table specifically tailored to the client's data. After the table has been created, the warehouse sends you a JSON message that looks like the client's Handshake request, except this time has the table's unique identifiers.

Handshake Request:

The client's JSON file request is supposed to look like this

```
{
    "classification": "Water Tower",
    "group_name": "Lunatic Labs",
    "sources": [
        {
            "name": "Water Tower 1",
            "metrics": [
                {
                    "asc": false,
                    "data_type": "float",
                    "name": "temperature",
                    "units": "celcius"
                },                {
                    "asc": false,
                    "data_type": "float",
                    "name": "volume",
                    "units": "gallons"
                }
            ]
        },
        {
            "name":" Pipeline 1",
            "metrics": [
                {
                    "name": "flow rate",
                    "data_type":"float",
                    "units":"cubic liters per second",
                },
                {
                    "name":"open",
                    "data_type":"boolean"
                }
            ]
        }
    ]
}
```

The request has three required fields (group_name, sources, and metrics) that act as layers for the request. The rest of the fields are optional that the client can add of their own accord.

Required fields:

group_name: this is supposed to put a name on a group of sources related to each other in some way, whether that be they are in the same location or they are under a specific team

sources: this is supposed to name the types of instruments used to measure whatever the client needs to measure. For example: thermometers, water pressure indicators, GPS, wind speedometer, etc.

metrics: this is supposed to be the units that the source measures in, such as: grams, inches, x-coordinate, y-coordinate, liters, etc.

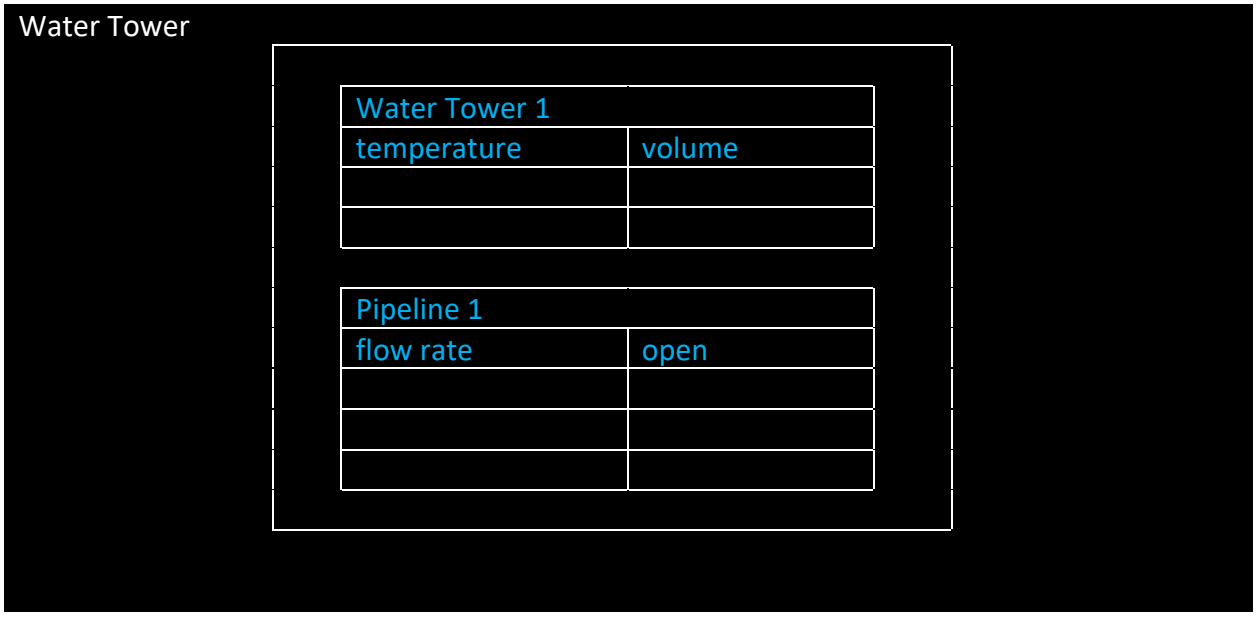
when creating the metrics portion of the JSON request it is required that the client specifies the metrics data type (integer, floating point, string, or Boolean) in "data_type:".

The relationship among each three required fields is such that a group_name can have many sources, but a source can only have one group_name, and a source can have many metrics, but a metric can only have one source.

Data Warehouse:

The client is to send a POST request to this URL
…/api/prepare

This creates a group with empty tables that are specific to the client's data in our database



*visual representation for conceptual understanding, not literal.

Finished Message:

After the empty table has been created, the data warehouse sends to the client a JSON message that looks like the client's Handshake request, except the group_name, sources, and metrics have unique identifiers so they can be found in our database.


```
{
    "classification": "Water Tower",
    "group_uid": "830486b6-0057-48ea-b820-cdaef9584289",
    "location": null,
    "name": "Lunatic Labs",
    "sources": [
        {
            "metrics": [
                {
                    "asc": false,
                    "data_type": "float",
                    "metric_uid": "0f88b6be-f8cd-4b9d-8dd1-62a1d0b0550b",
                    "name": "temperature",
                    "units": "celcius"
                },
                {
                    "asc": false,
                    "data_type": "float",
                    "metric_uid": "098e8aed-eefa-4da2-b95a-1feb9b19f82f",
                    "name": "volume",
                    "units": "gallons"
                }
            ],
            "name": "Water Tower 1",
            "source_uid": "06cc721e-abbd-4cd0-ae86-672c7690e94a"
        },
        {
            "metrics": [
                {
                    "data_type": "float",
                    "metric_uid": "25d17218-8e38-4424-8de4-c4a8a61c303c",
                    "name": "flow rate",
                    "units": "cubic liters per second"
                },
                {
                    "data_type": "bool",
                    "metric_uid": "5bae77fa-e45a-44af-aedc-e06c8b528019",
                    "name": "open"
                }
            ],
            "name": "Pipeline 1",
            "source_uid": "4637defd-d267-43ad-9dbf-4c66aa46c126"
        }
    ]
}
```