MOMA Meta Markers

This page defines the specification that software should adhere to in order to support **MOMA Meta Markers**. Meta Markers were created to improve our post-processing capabilities by allowing us to add markers *after* the telemetry has already been generated. This functionality may be used for any number of purposes:

- An existing marker is not properly positioned (with respect to the time domain) and does not correctly capture the desired data
- An analysis of the data reveals that more markers would improve data trending
- Markers to indicate "astronaut" events (see also Annotations)
- Markers to indicate anomalous events (see also Annotations)

A Meta Marker consists of

- marker ID (an integer in the range [50,000, 100,000))
- marker text
- start time
- end time

Unlike normal telemetry markers, which are packets inside of a tm.mom file, Meta Markers are generated by post-processing software based on predefined rules. Despite their simple format, human beings do not define Meta Markers in terms of these four fields. Instead, humans define rules which software uses to generate Meta Markers. The format and other details of these rules are described below.

Note that Meta Markers are different from Annotations. Meta Markers are generated by software to augment XINA Online's trending capability, whereas Annotations are generated by human beings using MOMA Data View in order to describe events within a single TID.

Storing Markers

Markers should be stored in a .json file in the momagse/MetaMarkerRules directory.

Example: 50050.json

Meta Marker 50050 is intended to adjust marker 50 by 10 seconds. These fields are defined in a following section.

```
{
  "meta_markers": [
    {
      "tids": "0-39999",
      "meta_marker_id": 50050,
      "meta_marker_text": "Adjusted Background Collection",
```

```
    "start_conditions": [
      {
        "type": "marker",
        "marker": 50,
        "offset_in_seconds": 10
      }
    ],
    "end_conditions": [
      {
        "type": "next_marker"
       }
    ]
   }
 ]
}
```

In this example, any time marker 50 is encountered, a corresponding marker 50050 will also be created.

Meta Marker Rule Format

Note that all .json files in the momagse/MetaMarkerRules directory will be used. So, Meta Marker Rule filenames can be named anything, as long as they have a .json extension.

Root Key

- `meta_markers` - The root node that contains an array of Meta Marker objects.

Meta Marker Object Fields

Each Meta Marker object has the following fields:

- `tids` - **String - Number Range** - Defines inclusive selection using ranges such as "0-29999" or using commas to identify
- `meta_marker_id` - **Integer**. Example: 50050
- `meta_marker_text` - **String**. Example: "Adjusted Background Collection"
- `start_conditions` - **Object** - When any of these conditions are satisfied, the marker starts (i.e., the start timestamp field is set).
- `end_conditions` - **Object** - When any of these conditions are satisfied, and when a Meta Marker is active (meaning that the Meta Marker has started but not yet ended) the marker ends (i.e., the end timestamp field is set).

The `start_conditions` array is a set of trigger objects.

The `end_conditions` array is a set of trigger objects.

Supported Trigger Types

The trigger `type` dictates how the marker's timestamps should be calculated. The different trigger types and their fields are described below.

Marker

The meta marker is defined relative to another marker ID.

- `type` : "marker"
- `marker` - **Number** - The marker ID that the marker is defined relative to.
- `offset_in_seconds` - **Number** - The offset in seconds from the `id`. Negative values offset to the left, and positive values offset to the right.
- `seb_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "SEB Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `high_voltage_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "High Voltage Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `laser_ebt_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "Laser EBT" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `gc_ebt_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "GC EBT" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `heater_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "Heater Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.

Message

The meta marker starts when a line in the message log matches a given regular expression.

- `type`: "message"
- `regex`: - **String** - The regular expression to look for
- `case_sensitive` - **boolean** - true if the regular expression mapping should be case sensitive, and false otherwise
- `offset_in_seconds`: - **number** - The timestamp of the message log line is used as the meta marker start time (if this trigger is used as a start condition) or the meta marker end time. This value will be added to the message log line timestamp before it is assigned as the meta marker timestamp. Negative values are allowed.

Duration

This trigger only applies to end_conditions.

- `type` - "duration"
- `number_of_seconds` - **Number** - the amount of time between the start of the meta marker and the end of the meta marker. If this value is positive, then generated meta markers will end number_of_seconds after the time when their start_conditions are satisfied. If this value is negative, then generated meta markers will *end* when their start_conditions are satisfied, and will start number_of_seconds before that end time.

Next Marker

This trigger only applies to end_conditions. If this trigger is set, then the Meta Marker end timestamp will be equal to the timestamp of the next telemetry marker.

- `type` - "next_marker"

Future Trigger Types

These trigger types are not currently supported, but might be in the future if the need arises.

Time

The marker is defined with a timestamp.

- `type` : "time"
- `mode` - **String** - "tid_relative", "absolute"
- `timestamp` - **Number** - The timestamp of the marker

Housekeeping Value

The marker is triggered by a HK value.

- `type` : "hk"
- `id` - **Number** - The HKID
- `comparison` - The comparison type: "<"
- `value_mode` - **String** - The value's mode: "raw", "eng", or "sci"
- `value` - **Number** - The housekeeping value
- `offset_in_seconds` - **Number** - The offset in seconds from the when the HK value comparison is satisfied. Negative values offset to the left, and positive values offset to the right.

Conventions

If a Meta Marker is relative to a normal telemetry marker (i.e., a marker packet in a tm.mom file), define the Meta Marker ID as `original_mkid + 50000`. This is convention only. If more than one Meta Marker is relative to the same marker, a different offset will be necessary.

Meta Markers cannot be relative to other Meta Markers.

A separate file should be used for each Meta Marker Rule.

Dependencies

This section details the dependencies that the Meta Markers have on the telemetry data. Ideally, any of the telemetry data that Meta Markers depend on should not change, but if it must, then whoever created the Meta Marker should be notified so that the Meta Marker can be updated.

Messages

- "engaging open loop mode"
- "engaging closed loop mode"
- "wrp startup complete"

Revision #1
Created 22 March 2023 20:35:41 by Nick Dobson
Updated 24 March 2023 13:47:06 by Nick Dobson