XINA Struct Trend

The XINA Struct Trend command line utility generates trend data products for a specified time range or Event Intervals. It is used by the XINA Trend Task. Generated data products include:

- Binned trend data
- Configurable trend plots
- Log file

Arguments

Name	Req	Description	Default
cache		Cache directory path. Used for any persistent caching across tasks for optimization purposes.	
conf	?	File path to the JSON config containing the trend parameters.	
env	?	File path to the JSON environment file which defines environment parameters such as project specific paths.	
help		Print available options.	
host		XINA Tunnel host.	`"localhost"`
port		XINA Tunnel port.	`41746`
import	?	Import directory path. Files that need to be imported will be placed here. `struct_trend` is responsbile for generating the corresponding import action file but `xina-run` will perform the actual importing.	
log	?	File path that log events will be written to.	
out	?	Ouptut directory path. Files placed here will be uploaded and accessible directly from the Task interface (e.g. notifications panel).	
post	?	Post directory path (for post-import outputs) e.g. uploading log files to the Task record.	
task	?	File path to the JSON file containing info about the Task.	
temp	?	Temp directory path for storing temporary files during execution.	

conf

The below table describes the available fields in the `conf` JSON file and is used to control how data should be trended.

Name	Type	Req	Description	Default
model	`utf8text`	?	Path of model to trend data from e.g. `moma.fm`	
copy	`boolean`		If true, the zip file will be uploaded and available from the Task interface.	false
trend	`utf8text`		The name of the Trend Definition to generate the trend for. If provided, then the `profiles`, `trend_conf`, and `plot_conf` will be loaded from it.	
label	`utf8text`		The text that will be used to name the files and final zip file. The format will be like `2024_06_12_00_00_00_2024_06_12_00_20_00_profile_label`	
profiles	`set(utf8string)`		Name of profile(s) to include in trend	
trend_conf	`struct_trend_conf`		See struct_trend_conf	
plot_conf	`jsonobject`		See trend plot conf	

Example conf:

This example performs a trend export with fields provided directly in the conf object.

```
{
  "label": "Test_Trend_From_Conf",
  "desc": "This is a test trend. The Trend parameters are provided in the conf.",
  "model": "model.path",
  "copy": true,
  "profiles": [
     "TEST_PROFILE"
  ],
  "trend_conf": {
     "t": [
        {
           "start": "2024-11-01T00:00:00Z",
           "end": "2024-11-01T00:10:00Z"
```

```
        }
      ],
      "bin_minutes": 1,
      "disable_filter": false,
      "intervals": null
    },
    "plot_conf": {}
}
```

In this example, the trend will be generated from an existing Trend Definition named `TEST_TREND`.

```
{
  "trend": "TEST_TREND",
  "model": "model.path",
  "copy": true
}
```

Revision #6
Created 5 February 2025 19:13:30 by Bradley Tse
Updated 5 February 2025 20:52:26 by Bradley Tse