Feature Specification for

SiESoCom

Time Series Aggregator 'TSA'

1.0

Revision History

Date	Who	Version	Info
17 May 2021	EP	0.1	Creation
02 Jul 2021	EP	0.2	Modeling Rework, slight logic update
02 Jul 2021	EP	1.0	Baselined

Contents

1 Context & System Topology

This document outlines the architecture and system requirements for the SiESoCom Time-Series Aggregator (TSA) application .

1.1 System topology views

The SiESoCom application is a multi-tier application which topology is depicted below. This specification deals with the Time-Series Aggregator (TSA) application tier(5)

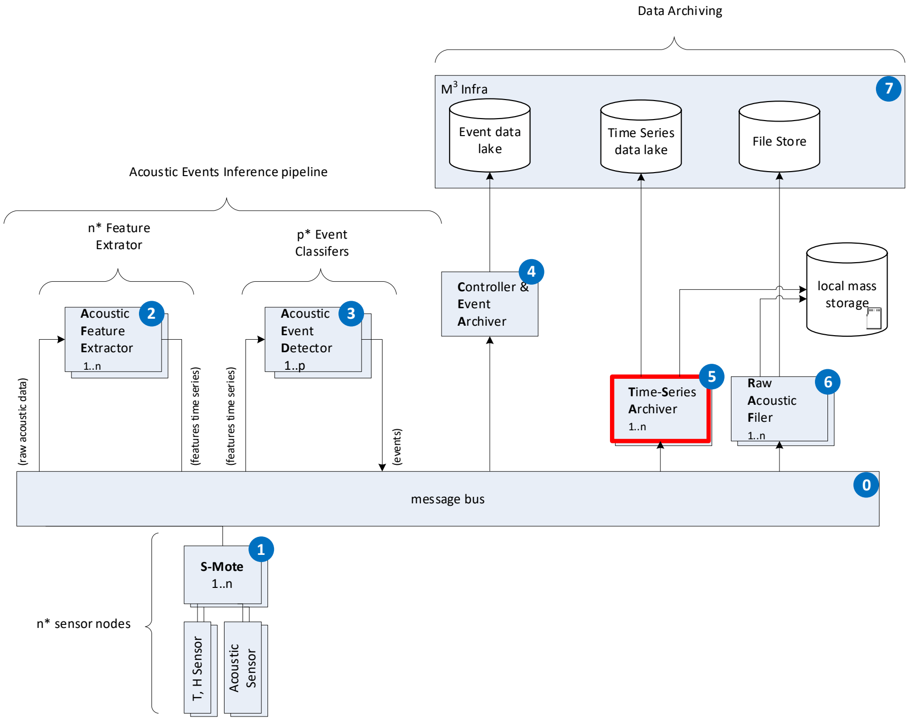

Fig: SiESoCom application topology – all tiers –

The SiESoCom TSA implements the aggregration logic of the M^3 generic controller meants to deal with multiple SiESoCom motes (S-Motes). Its main purpose is to upload, in batch, time series of scalar values into M3 time-series data lake and/or a local mass storage device.

Notes:

- We eliminate the G-Controller min/max/average static compression logic and upload the time-series without any alteration.
- Future versions of this components shall be able to deal with time series of multi-dimensional vectors.

2 Requirements

2.1 General requirements

In this version, only time series of scalar values can be archived. These time series can be uploaded, after some accumulation into M^3 generic time-series big table , or into csv formatted files on the local mass storage device available on the setup, or both.
These time-series are assumed to comply to the 'ScalarTimeSeries' dataschema

Time-series are serialized by the producer and shall therefore be deserialized before beign archived.

2.2 End Point selection

The number of time-series sources in a SiESocom setup can be large, so we assume that TSA can exist in more than one instance on the setup.

So, it is necessary for a given TSA instance to select a subset of the eligible data sources in the setup: A '**TSA input EPs**' configuration variable shall be implemented to allow a multiple selection of the scalar time-series endpoints of the application.
This config parameter is managed using the new portal "Dynamic Custom Config Parameter List" functionality. The dynamic query attached to the "TSA input EP" is specified in the "SiESocom Inference Pipeline" specification document in the section "List of TSA Scalar Input EPs"

Once the list of data source is set, the TSA application gets all the necessary information to subscribe to the relevant topics in order to ingest the time series.

2.3 Acquisition in Local data storage

We shall implement this feature using the following set of config parameters:

- Local Acquisition control
    - Acq State (per FSM defined below)
    - Acq State Additional Info
    - Mass Storage Device State (subcribed)
    - Mass Storage Avail Space (subscribed)
    - Readings per File
    - Command (start/stop)
    - start_date
    - duration
    - tag
    - run_id

When the local storage is running the time series shall be output into a dedicated path of the mounted mass storage device (based on setup, tag, run_id) , using a the following name format:

Directory: /<Setup_name>/<Tag>/<Run_Id>/*.raw
File name: CurrentYearMonthDayHourMinuteSecond-DeviceID-SensorID-Tag.csv

Also, a .txt companion file is created for each .csv file for the meta-data:

- file size
- setup name
- Run Id,
- Tag ,
- Tenant Id/name,
- Account Id/Name,
- Start date,
- Duration,
- Identification of the data source:
  - Device S/N,
  - Sensor S/N,
  - Sensor Configuration = {sampling_frequency}

This companion file shall be similar in structure than the RAF companion file.

The TSA application shall report it's current state and possibly additional state information at any time in the 'State', 'Additional State Info' config params of the 'Local Acquisition Control' group. (See FSM details below)

The acquisition state transitions shall be governed by the finite state machine define hereafter.

Note: The TSA local acquisition logic shall monitor the mass storage device mounting state and remaining avalaible space. In particular:

It shall not be possible to start a local acquisition if the USB mass storage device is not mounted or out of space.

Acquisition shall gracefully stop is USB mass storage device is full or unmounted.

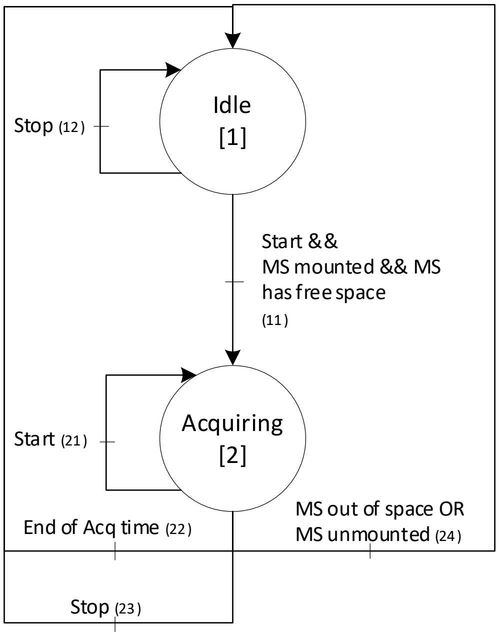

Fig: TSA Finite States Machine

Transition details & state information reporting:

11	Transition from Idle [1] to Acquiring [2] only on start signal, if MS is mounted with "enough" space. 'Enough' can be arbitrarily set to 100 MB. Once in state [2], report 'Acq State' as "Aquiring" and clear 'Acq State Additional Info'
12	Stay in Idle [1] on stop signal while in state [1] No update of 'Acq State' or'Acq State Additional Info'
21	Stay in Acquiring [2] on start signal while in state [2] No update of 'Acq State' or'Acq State Additional Info'
22	Transition from Acquiring [2] to Idle [1] on end of acquisition time timer. Once in state [1], report 'Acq State' as "Idle" and set 'Acq State Additional Info' to "run <run id> ran to completion"
23	Transition from Acquiring [2] to Idle [1] on user Stop signal. Once in state [1], report 'Acq State' as "Idle" and set 'Acq State Additional Info' to "run <run id> stopped by user"
24	Transition from Acquiring [2] to Idle [1] on device unmounted signal or device out of space. Once in state [1], report 'Acq State' as "Idle" and set 'Acq State Additional Info' to "run <run id> stopped due to MS device unmounted or out of space"

2.4 M^3 data lake archiving

This functionality requires a few control config parameters

Data Base storage control
    Active y/n
    Uploading period (sec)

The database archiving shall be activated by default but can be deactivated on demand using the 'Active' config parameter.

The upload period value controls the amount of buffering the TSA shall implement for DB storage. Typical upload period: 5 sec

This functionality is inherited from the G-controller aggregation: the TAS application shall call M^3 service to upload time series in batch into M3 generic time series table.

Note: Remember that min/max/avrg compression is NOT required for TAS.

Note: The Data Base storage activation/deactivation is independent from the local storage control. In particular, it shall be possible to excute a local archiving while the database upload is active.

3 TSA modeling recap

Based on the requirement of this document, the TSA (tentative) modeling is:

"State": (i) Not Stated, (ii) Initializing and (iii) Ready.
"Additional state information" can be provided optionaly e.g. 'Loading Configuration' during the Initialization state.
"Broker host": A group of subscribed config parameters defining the message bus details: IP address, port, security mode.

"Application credentials": A group of config parameters for (Broker username,password) and TLS cred (CAcert, ClientCert, Key and expiry date)
States of the other application tiers: Subscribed configs to controller state and message broker state at least, maybe more.

"TSA input EPs"

Data Base storage control
    Active y/n
    Uploading period (sec)

Local Acquisition control

```
Acq State (per FSM defined below)
Acq State Additional Info
Mass Storage Device State (subcribed)
Mass Storage Avail Space (subscribed)
Readings per File
Command (start/stop)
start_date
duration
tag
run_id
```