

Interactive comment on “A low-power data acquisition system for geomagnetic observatories and variometer stations” by Achim Morschhauser et al.

S. Bracke (Referee)

stephan.bracke@meteo.be

Received and published: 3 May 2017

The paper is good. The idea of having a low power general solution for timestamping/logging/transmitting magnetic data for different vector and scalar instruments is definitely something to encourage and can inspire others to commit to the project. Therefore I support the paper with some remarks .

On different places the stress is put on the fact that underlying os needs to be POSIX compatible (from the abstract to the conclusion). It's not clear to me why this is so important that it gets stressed more then once. In the section "Data Acquisition" it is not clear to me where data gets stored and when exactly rsync or ftp is used. Different suggestions or made "an external HDD/SDD may be attached to the PI" , "optionally

a laptop may be used for backups or second Raspberry PI may be installed" By using may we don't know exactly what is installed during your setup and how data is stored (Is data temporarily on the sd card of the PI , if so how long ?) and transferred (from Raspberry PI to PC? , from PC to data center via rsync,ftp ?) It is better to explain your current solution and suggest adaptations at the end of the section. The same remark is valid for the power supply needed when looking at the table 1 we see Netbook as biggest consumer but in the end you state that it is not needed in the setup. 4.2 Timestamping and Timekeeping In this chapter two different modes are mention to put the timestamp do you have an idea of the difference in accuracy ?

UML diagram Figure 5 : Public virtual method `filter_fir_sym` and `median_odd` are methods that use a vector of data objects to calculate median and filtered data and set the current object to this value I think this method can't be implemented at the level of `data_obs` because you don't have the possibility to know the real values (looking at the code the two methods on this level only work on the timestamp). They are overwritten in `data_obs_vector` (but missing on the UML diagram) where they have a reason to be. The class `data_obs_scalar` inherits the not useful methods on `data_obs`. It seems better to move them down to `data_obs_vector` class and even better to make a static factory method (that creates the necessary `data_obs_vector`) or a constructor that takes in the vector of `data_obs_vector` objects. While looking at the `data_obs_vector` class I have the following question: what about variometers that are not XYZ oriented where do they fit in ? To conclude on the UML I don't think they have an added value while explaining communication and datalogging by using the classes but not all are visible on the diagrams. If you want to use it I suggest to use one class diagram with all classes but hide methods and instance variables. 4.4 Integration in operating system. If I understand correctly cronjobs and batchjobs are used to make sure that everything started correct after reboot or powerloss. Is there a reason why you don't use services (systemctl) that always start or gracefully stop ?

Small type remarks : p2 1. 11 Niemgk p2 1. 15 in a productive system => in a

[Printer-friendly version](#)[Discussion paper](#)

production environment p2 1. 20 conclusions => conclusions p2 2. 25 specifies => specifies p3 3.1 25 Fig. XX p3 3.1 29 temporarily => temporarily p6 3.2.1 14 NGK

Best wishes Stephan

Interactive comment on Geosci. Instrum. Method. Data Syst. Discuss., doi:10.5194/gi-2017-23, 2017.

GID

[Interactive
comment](#)

[Printer-friendly version](#)

[Discussion paper](#)

