

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

**Achim Morschhauser et al.**

mors@gfz-potsdam.de

Received and published: 15 June 2017

Dear Stephan,

thanks for your helpful and detailed review. Below you will find my responses to the issues raised in your review.

Best wishes, Achim

=====

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

C1

important that it gets stressed more then once.

- I have removed the repeated occurrence of POSIX-compatibility. Now, it's only mentioned in the section "Software Layout" and the "Conclusions".

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.

- Now, the current solution is first explained before options are mentioned at the end of the section. For this purpose, the "Data Acquisition" section has been restructured and partially been rewritten.

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.

- Now, I left the laptop in table 1 in order to show the power consumption of the standard setup. At the end of the section "Power Supply Chain", I shortly discuss the power consumption of the different options. A note has been added to table 1 that the laptop and router may be removed.

### 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 ?

- Not exactly, but we plan to characterize the exact time delay in the future. According to the OBSDaQ manual, the triggered mode results in a timing accuracy of better than

C2

0.1 ms. Similar to basic mode (renamed freerun mode), the MinGEO PalmAcq module uses the GPS time of the incoming signal to produce timestamps. In this case, the manufacturer claims a timing accuracy of 1.5 ms, which may be similar to what can be achieved in basic mode. However, an additional time delay is present due to the response function of the used magnetometers. A note on time delays has been added to section 4.4.

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.

- As you pointed out, it is right that these methods can only work on the timestamp at the level of `data_obs`. The reason why I did not completely move these methods to `data_obs_vector` is that other subclasses (except `data_obs_vector`) may want to make use of the methods in `data_obs` (sure, at the price all subclasses inherit them). For example, a `data_obs_scalar` may want to overwrite this method in a similar way as `data_obs_vector`. The advantage is that the calculation of the timestamp's median is then implemented at only one point in the code instead of in every subclass. I am not sure if I correctly understand your suggestion regarding static factory methods and constructors, but as I understand, one resulting disadvantage would be that every time the filter is called, a new instance of `data_obs_vector` must be created. As the filter is called continuously in e.g. `Filter_Obs::fir_window_vec`, this might lead to more overhead in the code. Anyway, I am not 100% happy with the current implementation of

C3

median and filter calculation, mainly as I don't like that everything needs to be implemented separately in `Filter_Obs` for different subclasses of `data_obs`. If you have any more suggestions or if I didn't completely get your point, I will be very happy about contributions and/or discussions.

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 ?

- In principle, any other orientation data can be stored in the X,Y,Z objects of the `data_obs_vector` class. However, if any calculations should be done with the data where the orientation is important (which is not intended at the moment), the class may be extended with a flag that stores the orientation in order to properly perform any such calculations.

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.

- I agree it's better to show all instance variables and methods or none. As things get too complex when showing everything, I follow your suggestions to only show the classes and their relationships in one UML diagram.

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 ?

- That's correct. Although systemd is better embedded in the operating system, it is more complex to administrate. Also, service managers differ for some distributions, and have been replaced in the past (Init, SysV, rc), whereas cron and bash are available on all distributions. A note has been added to section 4.4.

C4

Small type remarks :

p2 1. 11 Niemgk p2 1. 15 in a productive system => in a environment - Changed to in an operative environment. p2 1. 20 conslusions => conclusions p2 2. 25 secifies => specifies p3 3.1 25 Fig. XX p3 3.1 29 temporarilly => temporarily p6 3.2.1 14 NGK - changed to Niemegk Observatory

- These typos have all been corrected.

---

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