Geosci. Instrum. Method. Data Syst. Discuss.,
doi:10.5194/gi-2017-17-AC3, 2017

Geoscientific Instrumentation Methods and Data Systems Discussions

Interactive comment on "Automated observatory in Antarctica: real-time data transfer on constrained networks in practice" *by* Stephan Bracke et al.

Stephan Bracke et al.

stephan.bracke@meteo.be

Received and published: 28 April 2017

First of all thanks for this detailed review with lots of interesting remarks. It took me a couple of days to update the paper and include most of the remarks made in this review. Special attention was given to the language used and the restructuring of the document. It's available as attachment. I only include here some comments on things I didn't include in the new paper and the reasons why. a) Some more details on the self-written software would be appreciated,how MQTT is embedded in your system. For example, you should mention that you use MQTT.js (I assume), and very shortly mention how node.js works. MQTT is just a specification that describes a protocol, to use it you need to include a library in the programming language of your choice. This library

will always have basic mqtt methods: connect, deconnect, subscribe,unsubscribe, publish. On the site of mqtt https://github.com/mqtt/mqtt.github.io/wiki/libraries you find over 60 libraries for different programming languages. I use currently three programming languages C#(windows autodif program) javascript (because of nodejs), and python all with different libraries to publish/subscribe to mqtt topics, but with similar approaches to use MQTT. The choice of using nodejs has nothing to do with the use of MQTT. After 15 years of software engineering experience on building websites with most of the time java solutions (still one of my favourite programming languages), nodejs was the first time that I saw a lightweight asynchronuous event-driven approach to web development which scaled and preformed well. Node.js changed the way web sites are build today compared to the more classical approaches (Java EE,ASP.NET, Ruby ON rails) It was also perfect to install on beaglebones and build a web interface. The ideas of node.js are currently integrated in python 3.5 with asyncio libraries and webframeworks as Sanic or on the database site asyncpg. Although these are very interesting topics to discuss on, I think they are out of scope for this paper. b) If you used the mentioned proprietary file-based windows software of LEMI and GSM, how would you use MQTT in this case –> reading from the file ? This might be interesting for Windows users . Reading the file and using MQTT to send the last added second/minute would not add any value compared to the classical ways of transferring data (Rsync,FTP etc) but can be done. Software changes are needed to embed MQTT and send at the same moment as the writing to the file.All software I written works on windows as well.

Please also note the supplement to this comment:
http://www.geosci-instrum-method-data-syst-discuss.net/gi-2017-17/gi-2017-17-AC3-supplement.pdf

Interactive comment on Geosci. Instrum. Method. Data Syst. Discuss., doi:10.5194/gi-2017-17, 2017.