Aquaplan Data Access Protocol (ADAP) Protocol specification

Markus v. Brevern, aqua_plan GmbH

Aachen, Dezember 2006

Inhaltsverzeichnis

1 Pre-considerations

1.1 Motive

Stations set up at distant places measure data. These measurements are saved. The memory can be read electronically. For the remote readout, the data transfer occurs via modem or ISDN.

The retrieving station saves the data in a central time series data base system. The retrievals occur periodically or on demand (for instance, in the case of a flooding).

The stations are equipped with current hardware. Modems are not slower than 56kb which means effectively not slower than 33kb. Instead of a strong compression of the data thus the easy error detection and the easy usability come to the fore.

A good data protocol is the foundation stone of a smooth dataflow between measuring station and data base system. It inures to the benefit of the user (thus the client) and strengthens the market position of the involved companies.

1.2 Volume consideration

Usually, the retrieval will occur once a day. A lot of measuring stations measure just one or two parameters (sensors), some have many more. In an example, a flood detention basin measures 10 parameters. We therefore consider 10 parameters to be a typical setting.

We take 5 min as a dense grid. Most water levels at surface waters are only stored in a 15 min grid. Rainfalls are even only stored event-driven. Therefore, they may have many in an event case, but, at an average, much less values than parameters stored with a fix timestep.

The dial-up access takes about 20 seconds via a modem. 'Long" in this sense would therefore be a transmission time of more than 10 seconds. A processing time of 30 seconds per sensor-rich measuring station is very good according to our understanding.

When thus per sensor 1s is provided per day and 288 5 minute values are transmitted with 3kbyte/s, an amount of good 10 characters per value will occur. If one considers the header overhead, an amount of 9 characters will occur.

1.3 Transmission error

Since years already, modems of all manufacturers have been able to transmit data with automatic error correction. Therefore, a checksum must not explicitly be provided in a protocol. Should the connection be very defective, some characters or whole sections

could be missing. Therefore, the protocol should contain redundancy and synchronisation positions.

1.4 Connection to the aqua_plan time series data base

Besides the value pair lists, aqua_plan time series also contain attributes for the identification and description of data (e.g. Location, Sub-location, Parameter, Unit). When these attributes are provided by the measuring station, the data will reach the proper time series without any detour (and without any misinterpretations involved).

The resolution of the time axis is not more precise than 5s. Points in time can thus only be placed on 5s. This is sufficient for all hydrological parameters, but must certainly be considered in case the measuring stations measure more accurate.

One of the biggest annoyments for the automatic further processing of data is the incorrect saving of gaps (i.e deficiencies). Therefore, one must regard the exact specification of gap values and gap dimensions.

2 Protocol definition

2.1 Hardware conditions

Generally the data protocol should be suitable for each transmission path and should not be subject to any conditions. An analog modem as least common denominator will be subject to review. The modem should nominally provide 56kbaud and V.42. The transmission mode should be 8,N,1. Of course, also inferior modems may be used due to technical boundary conditions. This, however, limits the safeness and the velocity. Faulty transmissions will lead to a redial and, with that, to extra setting up time.

2.2 Data type

All data are transmitted in the text mode. Numbers are expressed byte-by-byte as binary hex value. This counts for values as well as for points in time (see below).

Header information, except values and points in time, is output in readable plaintext.

2.3 Connection establishment

We have not planned any login control for the time being. However, this can easily be backfitted.

2.4 Query of the sensor information

The recalling institution (caller) sends orders to which the station reacts.

The connection persists until the caller explicitly abandons it or a timeout has been reached (e.g. 2 minutes).

The caller sends

`SENSORS?`

with a subsequent line feed (LF), in order to receive a list with information per sensor. The station answers with

`SENSOR` LF
Sensor information 1 LF
Sensor information 2 LF
...
`END` LF

The *sensor information* consists of fields that are separated by semicolons: sensor number; parameter; location; sub-location; unit; starting point in time; ending point in time; comment; current value; current time.

Feld Sensor number: Integer number in decimal notation

Feld Parameter: e.g. `water level`

Feld Location: e.g. `Buchenscheid HRB`

Feld Sub-location: e.g. `Inlet` or blank

Feld Unit: e.g. `cm`. Exception for raw data `RAW`

Feld Starting point in time: Beginning of the data to be queried

Feld Ending point in time: Ending of the data to be queried

Feld Comment: arbitrary comment

Feld Current value: last measured value

Feld current point in time: Time of the last mesurement

Each field (except the sensor number) can be left out, only the semicolon must be indicated. For a smooth process, the indication of at least the parameter, location and the unit is advisable.

See chapter 2.7 for the format of the points in time and the values.

The exception RAW in Unit incurs that the data in the measuring station have not necessarily been converted to the real units, but are existent as raw voltages, ticks or the like. For the conversion to the real unit, the data must be converted in the data base system. The infomation needed for this purpose is not held in the measuring station. For this case the data types Long and Short are also provided (see below).

The current value with point in time is meant for the case of emergency. This way, for example, current water levels of all involved gauges can be queried very fast by the simple query SENSORS?.

The format of this header line is made so that it is normally smaller than 100 characters and can thus be generated with SPS facilities.

2.5 Sensor Data Query

The caller sends DATA *Sensor number from to* LF, in order to query the data of a sensor on the time interval [*from*,*to*]. Example:

DATA 100 06461DE0 06469DE0 (12.4.2006 7:30 to 13.4.2006 7:30)

The station answers with a list of data blocks. It will normally only contain one data block, on deficiencies however, it can also deliver the existing data blockwise. The station will first send BLOCKS *Sensor_number Number_of_blocks*.

Subsequently, the blocks will follow, each starting with a header line with the following format BLOCK Number-of-Values Type-of-values Number type from to.

Field Number of values: Integral number in decimal notation

Field Type of values: I,*width*= equidistant interval values, *width in minutes* or K=nodes with timestamp

Field Number type: Display format of the numbers. The byte order is always little endian byte order. F=Float (4 byte floating point according to IEEE 754) L=unsigned long (32bit) VL=signed long (32bit) S=unsigned short (16bit) VS=signed short (16bit)

Field from: Starting point in time of the data in this block

Field to: Ending point in time of the data in this block

Example:

```
BLOCKS 100 1
BLOCK 96 I,15 F 06461DE0 06469DE0
```

Subsequently, the values, one value (or a value pair) will follow per line. Then follows a blank line.

2.6 Equidistant interval values

One value will be output per line which counts as **Mean value** for the interval. Each line ends with LF. Timestamps are not output. Example:

407072F6
4070FCF6

These lines describe the numbers 3,757017 and 3,765440 in IEEE-floating point notation with little endian byte order.

2.7 Nodes with timestamp

Each line consists of two words in hexadecimal notation, point in time and value which are seperated by a blank. The Y-value is formed as for equidistant interval values. The point in time is an unsigned long with the following bit assignment:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Year								Month				Day					Hour					Minute						Second			
0=2000								1-12				1-31					0-23					0-59						0,5,...,55			

Example:

026BBCA4 407072F6

Stands for 23.6.2002 15:10:20 3,757017.

The timestamps have an accuracy of 5 seconds since the aqua_plan-time series management has this accuracy. Split second values cannot be processed.

The format of the timestamps allows points in time to be compared with one another as integer values and so simplifies time domain analyses.

2.8 Gap display

For the display of gap values, instead of a hex value, an X is used as first character for the Y-value followed by, optionally, the type of gap or further X. Example:

XXXXXXXX

For equidistant interval values, the gap is valid on exactly that interval which is covered by the value. So, for 15 minute values, it is 15 minutes. For nodes with timestamp, the gap reaches from the last value pair that is not a gap to the next value pair that is not a gap. Example:

026BBCA4 407072F6 (23.6.2002 15:10:20 3,757017)
026BCC43 XXXXXXXX (23.6.2002 16:10:15 Gap)
026BCC82 407072F6 (23.6.2002 18:10:10 3,757017)

In the upper example, the gap reaches from 15:10:20 to 18:10:10.

3 Actuator commands

3.1 Time synchronisation

Using the SETTIME command one can transmit a timestamp to the measuring system of the station. The system time of the station is thus synchronised to the system time of the caller. This occurs up to the split second. Therefore the format of the timestamp is different to the one of the data recall. The time is transmitted in the format JJJJMMTTHHMMSS with a subsequent line feed.

Example:

`SETTIME 20061204141023 LF`

The remote station must acknowledge the SETTIME command within 10s with the answer ACK and LineFeed.

`06 LF`

If there is no acknowledgement, the current timestamp should be sent up to three times.

3.2 Further control commands

ADAP is expandable in order to use it also for controlling devices.