Discussion Paper

ANDS Project EIF 023

The application of quality control (QC) processes and QC flags to ship-based observations and measurements.

Authors:

Kim Finney (AAD)

Miles Jordan (AAD)

Paul Tildesley (CMAR)

Siddeswara Guru (CMAR)

Version 1.0
December 2010

CSIRO

ands
AUSTRALIAN NATIONAL DATA SERVICE

Australian Government
Department of Sustainability, Environment, Water, Population and Communities
Australian Antarctic Division

Table Of Contents

Appendix

1.0 Background

The Australian National Data Service (ANDS) has sponsored a project (EIF 023), aimed at publishing near real-time data from two of Australia's research vessels *RV* Southern Surveyor (RV SS) and *RV* Aurora Australis (RV AA). The managers of these vessels are CSIRO Marine and Atmospheric Research (CMAR) and the Australian Antarctic Division (AAD), respectively. Both CMAR and AAD are responsible for implementing EIF023, with assistance from the Australian Ocean Data Network Development Office (AODN DO).

RV SS and *RV* AA are sophisticated data collection platforms capable of acquiring data from the sea surface, the water column and the sea-bed, as well as from the immediate atmosphere. Typically, a suite of data known colloquially as "underway data", are sampled from a wide range of instruments whilst the vessels are on the move. The type of sampled parameters making up this "underway data" suite, the sampling regimes, sampling rates and formats of the captured data currently varies between vessels as do the data publishing policies.

The EIF 023 project was designed to bring the data publication processes of the two vessels into alignment so that both vessels routinely publish their "underway data" suites and accompanying metadata, in near-real time (automatically) to the AODN, and via this network into the Australian Research Data Commons (ARDC) service. This has the benefit of providing Australian researchers and environmental managers with almost immediate access to these data as they are collected, along with appropriate (shallow and deep) metadata. Importantly, data emanating from each vessel can then be integrated because the vessel managers are working together to ensure common data publication protocols and standards.

In addition to publishing near real-time underway data, the project partners have also published a number of historical underway datasets to the AODN. Unlike the near real-time data, these historical data are often subjected to a wide range of quality control (QC) processes, often conducted post voyage and implemented by on-shore data processing centres. As a result of these QC processes QC flags are usually inserted into these delayed mode datasets to indicate the results of processing. However, there is currently no "common" quality control framework adhered to by Australian vessels collecting underway data so the format of these "processed" data are different and the tests and flags that are used within these published datasets are not uniform. Agreeing to adopt a common data quality framework is important because near real-time voyage data will always be progressively replaced in the AODN by delayed mode (potentially processed) data. The quality control undertaken for near real-time underway data, if performed at all, is minimal. The data may be checked for values within range and a data-filler (some pre-defined value) is used to fill missing values.

This paper is intended as a discussion starter on what might constitute a data quality control framework for "underway data". Ideally this same framework, if sufficiently flexible, could also be applied to other observation and measurements collected onboard Australian research vessels.

2.0 What Is A Quality Control Framework

Most data that are captured by automated instrumentation and techniques can be subject to a wide range of problems and distortions caused by equipment malfunction, equipment operator error, electronic interference and/or poor instrument calibration. Depending on the uses to which data are put, it is usually wise to "inspect" captured datasets prior to their application so that there is some level of confidence in the "quality" of the data. Knowing something about data quality permits a user to assess whether a dataset is fit for purpose. Most scientific Data Centres generally perform some level of "vetting" on data acquired from ship-based instrumentation prior to publishing it, or disseminating it widely to user groups.

With the advent of service-oriented computing environments it is important that the methods used to "flag" any data that have been subjected to quality control processes are eventually machine, as well as human interpretable. This is a complex area of data management because to attain true machine interpretation would require a formal (logic-based) description of all of the terms and symbols used in the QC process and rule-based descriptions of all possible tests performed.

Machine "interpretable" in the sense that this paper coins the term is therefore restricted to mean a set of agreed conventions for "marking up" published data containing quality control flags that have been applied after data has been subjected to one or more tests. "Interpretable" in this case means that the conventions provide for an explicit and transparent encoding of the flags and a description of the tests, such that datasets that have undergone QC can be readily compared (from the perspective of the quality control processes performed on them).

For the purposes of this discussion paper a Data Quality Control Framework addresses several aspects of performing quality control on acquired data:

(a) The processes and tests that can be applied to "qualify" datasets,
(b) The flagging regime used to represent performed tests and the flags that are used to "mark-up" data that has been the subject of processing,
(c) Guidance on the manner in which QC flags are encoded in published data,
(d) The versioning process and notification methods used to indicate which "version" of tests have been applied to datasets.

It is assumed that information (metadata) pertaining to the type of equipment used and any associated equipment-based sensors, their identifiers and any relevant calibration coefficients, are all also routinely recorded and retrievable as part of the published data package. In circumstances where laboratory analysis has yielded measurement values it would also be wise to include such information as sample preservation/handling techniques, analytical methods, detection limits, sample preparation procedures and any associated known measures of uncertainty. These items of information are also intrinsic to assessing the "quality" of data and hence its fitness for purpose.

3.0 Existing QC Frameworks

A significant amount of work has been done recently on quality control frameworks for scientific oceanographic observations and measurements. It is not the purpose of this paper to do an inter-comparison of all of these frameworks. Instead, we have drawn on this existing body of work and extracted from it the main issues that we believe: (a) are commonly addressed across the frameworks and (b) are significant points for discussion and debate amongst Australian data managers (and scientific data users). Rather than landing on any specific set of recommendations, we have chosen to point out the positive and negative aspects of any approaches that we canvass.

It is envisaged that the AODN DO, ANDS or the Australian Ocean Data Centre Joint Facility (AODC JF) might establish a dedicated working group devoted to formulating, or adapting a framework that is suitable and flexible enough to be adopted nationally (for specific data types).

3.1 Common Framework Issues

(a) QC Flags

At the heart of the Data QC Framework are the QC flags. QC flags are usually one or two digit integers that are attached directly to measurement or observation values in the dataset. In rare cases these flags are characters, as in the system used by the British Oceanographic Data Centre (BODC) and the World Ocean Circulation Experiment (WOCE). Each flag can be considered a code, which has an assigned meaning (see Table 1 for an example).

Table 1 – GTSPP QC Flags (GTSPP, 2010)

Code	Code Description
0	No quality control has been assigned
1	QC was performed; appears to be correct
2	QC was performed; probably good
3	QC was performed; appears doubtful
4	QC was performed; appears erroneous
5	The value was changed as a result of QC
9	The value is missing

Most systems that were reviewed for this paper were established on the principle that data which undergoes quality control is flagged, not replaced. So whilst it is possible to flag a value as erroneous and then insert an additional new value (flagged as having been inserted), the original (erroneous) value would not be removed from the published dataset.

In the main, existing QC flagging systems either provide an assessment of data quality (e.g. good, bad, unknown, questionable), give information on what tests the value may have failed, or use a combination of both of these.

When there is a quality assessment flag only attached, there is more of an inference of subjectivity about the assessment, than if there is a quality assessment provided that indicates that a data point has failed a specific QC test. For example, a flag of “6” in a system might mean “bad data; failed range test”. The reason for the quality assessment then becomes more transparent for the user.

Recent thinking (see IOC Workshop Report No. 28) is that ideally a flexible QC system would incorporate both a quality assessment element and a quantifiable component describing the rationale for the assessment. The argument used to support this case is that given the propensity for a large number of QC schemes, particularly those developed for discipline-specific purposes, having a universally applicable quality assessment component would enable a high level mapping between disparate schemes. It would also permit searching and/or extraction of data from data delivery systems based on data quality. Suggestions arising from the IODE workshop on chemical data QC (IOC Report No. 28) culminated in the set of quality assessment flags as shown in Table 2.

Table 2 – IOC QC Flags (IOC, 2010)

Code	Code Description
0	Good (= passed all of the applied QC tests)
1	Quality Not Evaluated
4	Questionable/Suspect (meaning the evaluation is inconclusive because the value failed non-critical objective/quantifiable or subjective QC tests)
8	Bad (=data value failed critical objective/quantifiable QC test)
9	Missing data (although not a data quality indicator, it serves as an important marker).

CMAR (Pender, 2000) has been using a relatively sophisticated quality flagging system for over 10 years, which it applies to a wide variety of datasets (not just underway data). This system is based on the notion that there are three types of information recorded in the quality control flag:

- Data State: synonymous with the quality assessment paradigm outlined in Table 2 above,
- Operation Type: where the type of manipulation performed on the data is described through a code (e.g. the data value is the result of filtering, interpolation, or manual adjustment), and
- Error Type: the type of test the data value failed to pass giving rise to the assigned Data State.

The CMAR system sets the flags using a “byte” (i.e. a unit of information described using 8 bits, where a byte can represent a numeric value between 0 to 255). Error Types, Data States and Operations are each assigned different numeric values and are then encoded into different bits of the byte. Essentially the final flag numeric value is the sum of the numeric values for each of the three types of constituent components. A flag of “9” for example means a Data State of “Good”, an Operation of “None” and an Error Type of “Data out of range”. Using bytes and bits is a rather complicated way of encoding the flag value (for humans), but the conceptual, three-tiered design of the system which ascribes a Data State, articulates an Operation performed on the data and lists the detected Error, accords well with more recent musings on the preference for constructing multi-dimensional flagging systems.

It is obvious that any single system of devised flags will fail to cover all of the niche-specific tests that can and will be performed on multi-thematic, multi-disciplinary scientific data. The reality is that there will be multiple systems. Being able to map between flagging systems would be facilitated by garnering some agreement on what the common information facets are that any flagging system should represent. The CMAR case has identified three types of information, but perhaps there are more. It may well be that it’s relatively easy to standardise on the content for some of these facets, whilst accepting that others, such as the detected errors/tests components will always be data-type or use-type specific.

It is also worth considering how we should encode flags. Most of the systems reviewed used either an integer value or a character, presumably for the sake of brevity and simplicity. The question is are there better ways to construct codes which represent multi-dimensional information such that they are easily disambiguated and interpreted (by humans and machines) ?

In most circumstances flags are applied to individual data values but there are cases where it is desirable and appropriate to flag aggregates of values. For example, in time series data such as CTD profiles measured along a ship's track, it may be relevant to flag an entire profile, as well as the individual profile observations.

(b) QC Tests

As discussed above, performing quality control processes is primarily concerned with ultimately annotating data values with an assessment of quality. Generally, Data Centres use more than intuition and subjective judgement about the validity of data values and instead base their quality assessments on whether a particular data value passes, or fails, a number of "tests". The quality assessment of the data is therefore valid only in the context of the "tests" performed. It is up to a user to determine whether a specific dataset, that has passed several described tests, is therefore fit for purpose. It is highly possible that "good" data for one use-case is deemed "bad data" for another use-case. An example of this might be where marine data that passes a set of simple, but rudimentary tests designed to establish whether observations points fall on the land or in the sea, and which are classed as "good" data, are subsequently judged as "bad" data by someone who then applies a further "data value in range test" and finds the instrument was periodically malfunctioning. So, in summary, quality is context specific and it's important to be able to describe the context.

Regardless of the exact tests performed there are some standard pieces of information that should be recorded and available with respect to each QC test applied to a data set. The IOC Global Temperature and Salinity Pilot Project (GTSPP) Manual 22, recommends that a full description of a quality control test should include:

- **Test Name**: A short name and a number
- **Pre-requisites**: A description of what tests are assumed to have been applied before the application of this test and what additional files might need to be used as part of the test.
- **Description**: An outline of how the test is executed and what actions are taken as a result.
- **History**: This information element records any changes that have taken place in the test procedure and the date on which these changes were made. This item records the lineage and evolution of the test.
- **Rules**: This information identifies the rules that are applied to affect the various tests. Each rule should be numbered. The GTSPP rules are often sets of individual IF-THEN-ELSE statements and are described using flow chart notation.

As can be seen from this GTSPP example it is often important to apply tests in some type of order. This view is confirmed by the QARTOD organisation, a NOAA backed group that is experimenting with encoding QC tests using SensorML. QARTOD describes tests as components of SensorML processes that have "inputs", "associated processing components" and "outputs" including the QC flags generated by the test. Processes can be grouped into ProcessChains (which have a clear processing order). Tests, Processes, and Flags are given their own URIs by QARTOD so that they can be easily accessed and referenced. There is, however, no standard method as yet for describing quality control tests and processes.

Individual tests that are often applied to underway datasets include checks for values out of range; statistical outliers as compared to a representative climatology and unrealistic ship movements or positions. This last category of test is applied to in-file metadata for recording ship or platform position, observation date and time. The former categories are considered to be tests of “data value” veracity.

The WOCE QC Handbook (1996) describes in detail the types of tests applied to marine meteorology data, which are often an important constituent of the “underway data suite”. WOCE tests include both data value and inline metadata centric assessments.

In the WOCE framework, checks encompass a first-pass suite of tests to establish whether units of measure have been included with the data, if there are duplicate records or datasets and whether there are any perceived typographic errors. Typographic errors are common in datasets where there is some human transcription involved in data recording (for example a timestamp of 2008 in a dataset where every other data point is stamped 2010 would tend to indicate an obvious transcription error). There are also other types of checks designed to assess whether data points have been recorded in an appropriate time sequence, presumably again to detect transcription errors.

It can be common practise for a Data Centre to convert file formats, or transform particular data values, based on a conversion between original units of measure and those desired by the Data Centre (often for the sake of creating uniformity between datasets). Changes can also be made when there is a need for a dataset to conform to a particular convention (e.g. the WMO convention on recording wind direction differs by 180 degrees to the conventions used in oceanography, so in WOCE, wind direction values were often transformed and brought into alignment with WMO standard practises).

It is important to mention that within WOCE, datasets are encoded using a file format called netCDF (network Common Data Form - http://www.unidata.ucar.edu/software/netcdf/). netCDF is a self-describing file format which readily accommodates the application of QC flags. In netCDF files there are: global attributes (which apply to the whole file e.g. file title, ID); well-described variables (e.g. pressure, temperature, time latitude, longitude) that have their own attributes (e.g. units, instrument height, instrument type) and the actual data values. There is ample scope within this rich encoding for attaching flags to any of the contained file items. Generally only a single alphanumeric character is assigned in the WOCE system, which means that the WOCE framework has had to adopt an order of preference for flags (i.e. if a data value fails two tests, you can only use one flag, so the flag chosen is assigned according to a flagging hierarchy). This practise seems a little restrictive (and potentially confusing) given the scope for applying multivariate flags in netCDF files.

While many existing QC frameworks use a wide variety of automated tests, most systems still rely on human moderation. Automated tests are used in the main to highlight potential problems so that a domain expert can then use subjective judgement to either confirm or reject the software’s classification of the data. Most frameworks still extol the virtues of human moderation, particularly for the more “scientific” tests (e.g. tests that evaluate whether suspect data are realistic given the local sampling context and environmental conditions). These manual checks are also needed to detect problems in the data that are not detected by automated procedures.

In general, existing systems also encourage the creation of QC Processing Reports that provide a commentary on various issues that arise during processing (e.g. interesting but suspect features, decisions made that might be borderline etc). These reports are usually linked in some direct manner with the processed dataset.

(c) Versioning Issues

One problem encountered in processing datasets is how to record (a) the processing history of datasets when a dataset has been subjected to multiple processing systems (often undertaken by different organisations), (b) the evolution of the tests and testing regimes applied to datasets by communities of practise using “named” testing suites [e.g. the GTSPP Real Time Quality Control Checks, or the SEADATANET Quality Control Standards] and (c) the evolution of flagging systems.

All of the above are “versioning” issues of one form or another. With respect to (a), (b) and (c), explicit encoding of the processes used to apply tests, recording of the descriptions of the terminologies that are involved and the flags associated with each of these processes, can all easily be performed using some XML-based language. These facets of information can either then become part of the delivered dataset or be referenced from within the datasets. Having all of this information available for each qualified dataset is a foolproof mechanism for versioning. But the overheads of establishing the necessary web-based resources are high and there is also the problem that dataset “bloat”, where the in-file dataset values themselves become obscured in a sea of qualifying (metadata) information.

Experimentation being undertaken by the QARTOD Q20 Group is specifically using Sensor Observation Services (SOS) to deliver data and QC information to users. The content-based resources established to support the SOS can theoretically be consumed by any web client. So an investment in developing a flexible, well described and sufficiently re-usable framework will probably pay dividends many times over, despite the up-front effort involved in its initial establishment.

Other groups such as the GTSPP community have chosen to maintain a community document which is versioned and this version number is used in datasets that have had the GTSPP QC test suite applied. All that is required in this approach is for a community to update and maintain a manual (which can be web-based), version the document and then list the document version number in the processed dataset. This is a less elegant versioning system than the QARTOD approach but highly achievable and could form the basis of a more sophisticated (QARTOD-like) approach later.

The problem of passing datasets through multiple (often different) QC systems and then tracking QC history is a more difficult issue to solve. The GTSPP community permits the use of the document version number (e.g. version 1.02), where the first 2 digits represent the GTSPP QC suite and the 3rd digit is agency-specific and indicates that additional tests have also been applied to the data. Presumably this digit means something to the agency that has applied additional tests and hopefully the number links to an explicit public description of what these additional tests are.

3.2 Potentially Useful References

Fredericks, J., M. Botts, L. Bermudez, J. Bosch, P. Bogden, E. Bridger, T. Cook, J. Graybeal, S. Haines, C. Rueda and W. Christoph (2009). Integrating QA/QC into Open Geospatial Consortium Sensor Web Enablement. OceanObs'09. Venice, Italy.

Smith S, Harvey C, Legler D.M. (1996). WOCE QC Handbook of Quality Control Procedures and Methods for Surface Meteorology Data. Accessible from: http://www.coaps.fsu.edu/woce/docs/qchbook/qchbook.htm

UNESCO,.2010. GTSPP Real_time Quality Control manual, Revised Edition, IOC manuals and Guides 22. Accessible from: http://www.iode.org/index.php?option=com_oe&task=viewDocumentRecord&docID=6437

UNESCO. 2010. First IODE Workshop on Quality Control of Chemical Oceanographic Data Collections. Workshop Report No. 228. IOC Project Office for IODE, Oostende, Belgium 8-11 February. Accessible from: http://www.iode.org/index.php?option=com_oe&task=viewDocumentRecord&docID=5109

QARTOD Q20 web site at http://q2o.whoi.edu/

Marine Metadata Interoperability web site at http://marinemetadata.org/

SEADATANET QC Standards Presentation accessible at: http://www.seadatanet.org/imdis2010/content/download/24093/342048/file/IMDIS2010_oral1_05_Rickards.pdf

Uncertainty Mark Up Language web site describing UncertML a language for encoding probabilistic uncertainties at: http://www.uncertml.org/

Appendix 1

DATA QUALITY CONTROL FLAGS

LINDSAY PENDER

29 June 2000

Specification

All quality control flags are to be unsigned integer numbers in the range 0 to 255. Where appropriate, the flags are to be stored as unsigned byte length values. Each byte length QC flag is subdivided into 3 fields. These fields are defined as follows:

Data State (bits 6 & 7)
The data state describes the overall status of the data without concern about the type of error, and the type of correction process performed on the data, if any. If the QC is unknown, the person loading data must determine the data state, i.e. unknown QC does not necessarily imply no QC.

Data State	Numeric value	Description
0	0	Data is good
1	64	Data is suspect
2	128	Data is bad
3	192	No QC

Operation type (bits 4 & 5)
The operation type describes the type of operation performed on the data to enable it to be classified with the given data state.

Operation	Numeric value	Description
0	0	No operation – data used as is.
1	16	Data has been interpolated to replace bad values.
2	32	Data has been averaged or otherwise filtered.
3	48	Data has been manually adjusted.

Error type (bits 0 & 3)
The error type describes the type of data error detected which resulted in the given data state and subsequent operation on the data.

Error type	Numeric value	Description
0	0	No error – data is good, or if no QC, error is unknown.
1	1	Hardware error.
2	2	Software error.
3	3	Operator error.
4	4	Error flagged by hardware.
5	5	Error flagged by processor.
6	6	Analytical error.
7	7	Recording anomaly, e.g. transcription error.
8	8	Data stream corrupted, e.g. communications fault.
9	9	Data out of range.
10	10	Anomalous spike, e.g. data spikes.
11	11	Preliminary processing (calibration) only.
12	12	Unprocessed (uncalibrated) or processing error.
13	13	No data – data missing for unknown reason.
14	14	Timing error.
15	15	User defined – user must provide adequate description.

Numeric interpretation

The complete flag for a given data element is the sum of the numeric values of the 3 fields. To unpack a flag, the user can either use a lookup table, or perform the following manipulations:

Arithmetic method	Bit manipulation method
To unpack a flag: state = int(flag / 64) op = int((flag – state * 64) / 16) error = flag – state * 64 – op * 16	*To unpack a flag:* state = flag >> 6 op = (flag & 0x30) >> 4 error = flag & 0x0f
To pack a flag: flag = state * 64 + op * 16 + error	*To pack a flag:* flag = (state << 6) & (op << 4) & error

On some systems and file formats, eg. netCDF, it is not possible to store unsigned byte values. In this case, flags greater than 127 are stored as negative numbers. To convert them to unsigned integers, add 256.

If a user is only interested in the state flag, the following can be used to interpret flags:

State	Unsigned Byte	Signed Byte
Good	$0 \leq flag \leq 63$	$0 \leq flag \leq 63$
Suspect	$64 \leq flag \leq 127$	$64 \leq flag \leq 127$
Bad	$128 \leq flag \leq 191$	$-128 \leq flag \leq -65$
No QC	$192 \leq flag \leq 255$	$-64 \leq flag \leq -1$