Draft Final Report **R&D Project 239**

Codes of Practice for Data Handling
Version 1

WRc plc

September 1992

R&D 239/6/ST

CODES OF PRACTICE FOR DATA HANDLING

J C Ellis, P A H van Dijk and R D Kinley

Research Contractor:
WRc plc
Henley Road Medmenham
PO Box 16 Marlow
SL7 2HD

National Rivers Authority
Rivers House Waterside Drive
Almondsbury Bristol BS12 4UD

NRA Draft Final Report 239/6/ST

National Rivers Authority
Rivers House
Waterside Drive
Almondsbury
BRISTOL
BS12 4UD

Tel: 0454 624400
Fax: 0454 624409

Dissemination Status
Internal: Limited
External: Restricted

Research Contractor
This document was produced under R&D Contract 239 by:

WRc plc
Henley Road Medmenham
PO Box 16 Marlow
Buckinghamshire
SL7 2HD

Tel: 0491 571531
Fax: 0491 579094

WRc Reference: NR3203/4224

NRA Project Leader
Dave Brewin (Severn Trent Region)

Additional Copies
Additional copies of this document may be obtained from the Project Leader. The codes of practice will be updated periodically; these updates will be sent to registered document holders.

NRA Draft Final Report 239/6/ST

CONTENTS

EXECUTIVE SUMMARY

The NRA's routine quality monitoring activities generate a huge amount of data. To provide useful information, this data needs to be analysed appropriately; and the overall aim of the project has been to improve the general effectiveness with which the NRA carries out such data analyses. In pursuing this aim, the project has focused on two key requirements:

1. the need for statistical methods that will perform as reliably as possible in circumstances when specialist advice is not readily available; and

2. the importance of consistency between NRA regions.

The project has produced a series of seven Code of Practice guidance notes with the following titles:

- Dealing with less-than and greater-than values;
- Handling outliers;
- Estimating percentiles;
- Presenting summary statistics;
- Estimating mean load;
- Using the Test Data Facility; and
- Developing software for the Test Data Facility.

Two of the Codes of Practice refer to the 'Test Data Facility' (TDF). This is a software system enabling the practical value of any proposed method of data assessment to be judged directly by applying the method to a wide selection of user-defined data sets. To demonstrate the ease of use of the system, four TDF procedures were developed during the project: these deal with river quality classification, outlier testing, summary statistics, and step-trend detection.

A supplement to the 'Handling outliers' Code of Practice has also been produced, entitled 'Guidelines and methods for Data Quality Control'.

KEYWORDS

Summary statistics, parametric methods, non-parametric methods, confidence limits, percentiles, loads, outliers, less-than values.

INTRODUCTION TO THE SERIES

Background

These Codes of Practice provide a series of standard methods for the handling and interpretation of data within the NRA.

Before privatisation, each of the ten Regional Water Authorities had its own responsibilities for collecting and processing water quality data. As a consequence, data handling developed more or less independently within each Authority, leading to a variety of methods being used for the same task.

The situation changed with the formation of the NRA, when for the first time there was the opportunity to adopt a common set of procedures for data handling. To provide this uniformity across the ten NRA regions, the Steering Group on Data Handling was set up in Jan 1991 to establish protocols and to issue these in the form of a series of Codes of Practice.

Format

Codes of Practice are generally divided into three parts. Part A gives a concise declaration of agreed rules and methods. Next, Part B provides some background discussion enabling the interested reader to understand the reasons behind the recommended rules. Part B also includes worked examples where appropriate. Finally, Part C provides technical details such as statistical algorithms, reference tables and program listings.

Scope

The immediate benefit to the NRA of adopting any particular Code of Practice is in the greater consistency it brings between regions. It should be stressed, however, that the Codes of Practice do not provide 'rubber stamp' procedures to be used uncritically in all circumstances. As the supporting discussions emphasize, the statistical and other assumptions underpinning the recommended approach need to be checked wherever possible. Whenever the user is in any doubt as to the applicability of a Code of Practice, therefore, it is important that he or she seeks the advice of a statistician.

Developments and amendments

When the project ended in March 1992, eight Codes of Practice had been prepared. Colleagues are invited to bring to the notice of the Steering Group any requests for further Codes of Practice to cover new areas of application. Notification of errors in existing Codes of Practice, or any improvements that could be made to them, will also be welcome.

METHODS FOR HANDLING LESS-THAN VALUES AND GREATER-THAN VALUES

PART A - RULES

This Code of Practice gives a set of rules designed to ensure that less-than and greater-than values are handled consistently throughout the NRA. The rules themselves are given here in Part A without further clarification. More detailed explanation is given in Part B.

We first define two terms for use in this Code of Practice:

a. The **true result** is the result that would have been obtained if the less-thans or greater-thans could have been measured;

b. The **face value** of a less-than or greater-than is the value after the '<' or '>' sign.

RULE 1: Substitutions for less-than values

Either Rule 1a or Rule 1b should be applied to less-thans.

If it is logically impossible for the determinand to take negative values then zero acts as a lower bound for the less-thans and Rule 1a should be applied. (It is possible, but rare in practice, for some value other than zero to act as a lower bound.)

If there is no clear lower bound, then Rule 1b should be applied.

RULE 1a: Double substitution for less-than values with a lower bound

Substitute zero (the lower bound) for each less-than value and perform the desired calculation. Then substitute the face value for each less-than, and perform the calculation again. Both results should be quoted. The true result will lie somewhere between them. Only if their difference is of no practical importance may a single value (halfway between the two calculated results) be used in subsequent work.

RULE 1b: Single substitution for less-than values with no lower bound

Find a one-sided limit to the true result by replacing each less-than by its face value and performing the desired calculation on the modified data set. Where an arithmetic mean is calculated by this method, the output should state that this is an over-estimate.

RULE 2: Substitutions for greater-than values

Either Rule 2a or Rule 2b should be applied to greater-thans. Rule 2b should be used in those rare cases where there is an upper bound for the greater-thans. This upper bound is an absolute maximum which it is logically impossible for values of the determinand to exceed. In general, there will be no clear upper bound and Rule 2a should be applied.

RULE 2a: Single substitution for greater-than values with no upper bound

Find a one-sided limit to the true result by replacing each greater-than value by its face value and performing the desired calculations on the modified data set. Where an arithmetic mean is calculated by this method, the output should state that this is an under-estimate.

RULE 2b: Double substitution for greater-than values with an upper bound

Substitute the face value for each greater-than value and perform the desired calculation. Then substitute the upper bound for each greater-than value, and perform the calculation again. Both results should be quoted. The true result will lie somewhere between them.

RULE 3: Other methods

Always seek advice from a qualified statistician before using any other method for handling less-thans and greater-thans.

RULE 4: Graph plotting

Plot less-than and greater-than values at their face value, using special symbols or colours or both. The recommended symbols are downward-pointing triangles or downward-pointing arrows for less-thans and upward-pointing ones for greater-thans. A key explaining the symbols should be included on the graph.

RULE 5: Notification

In data summaries, always indicate clearly the existence of less-than or greater-than values in the data, and the method and value of any substitution used.

METHODS FOR HANDLING LESS-THAN VALUES AND GREATER-THAN VALUES

PART B - BACKGROUND

Water quality archives often include less-than values (e.g. <0.1) or greater-than values (e.g. >43.0). Less-thans commonly arise through the existence of a Limit of Detection for the analytical method, whereas greater-thans are generally the consequence of an insufficiently wide range of dilutions in the chemical analysis. Unless some substitution is made for the less-than or greater-than values, this 'censoring' of the data causes problems when performing numerical calculations such as deriving the annual mean.

Part A gives a set of rules designed to ensure that less-thans and greater-thans are handled consistently throughout the NRA. Part B now describes these rules in more detail.

Throughout this Code of Practice, we use two terms defined below.

a. The true result is the result that would have been obtained if the less-thans or greater-thans could have been measured.

b. The face value of a less-than or greater-than is the value after the '<' or '>' sign. Note that the face value need not be the same throughout a set of data.

Less-than values in water quality data generally arise where, because of experimental uncertainty, it is impossible to distinguish between very low concentrations and complete absence. To handle this uncertainty, Analytical Quality Control (AQC) methods are required. Two important concepts are involved: the criterion of detection and the limit of detection.

The criterion of detection (C) is the lowest observed value that is statistically significantly greater than zero. Thus, if an observation is less than C, it is impossible to claim with adequate confidence that the substance in question has been detected. If a value greater than C is observed, then the analyst is confident that the substance is present.

The limit of detection (L) defines the lowest true concentration that will be detected with a given degree of confidence. In other words, if the true concentration is equal to L, there is an adequately high probability that the observed concentration will be greater than C. The magnitudes of C and L are determined by AQC methods, taking into account the repeatability of analytical results and the required levels of confidence. A full description is given in WRc's AQC Manual (NS31).

Under the AQC procedure, any observation greater than the criterion of detection will be recorded as it stands. On the other hand, any observation smaller than the criterion of detection will be recorded as being less than the limit of detection, i.e. any observation between 0 and C is quoted as <L. For example, suppose that C = 10 and L = 20 µg/l. Then an observed value of 12 µg/l will be recorded as 12 µg/l, but an observation of 8 µg/l will be recorded as <20 µg/l. Therefore, any values recorded as <20 are, in truth, bounded below by zero and above by 10 µg/l.

In practice, the distinction between C and L is often blurred, and it is not always clear whether the quoted face value for less-thans is the criterion or the limit of detection. This Code of Practice, therefore, takes a pragmatic view rather than a strictly theoretical one, and, to err on the safe side, requires that the face value itself be used in all substitutions. Thus, to continue with the above example, the bounds on values recorded as <20 will be taken to be 0 and 20.

RULE 1: Substitution for less-than values

Either Rule 1a or Rule 1b should be applied to less-thans. If it is logically impossible for the determinand to take negative values then zero acts as a lower bound for the less-thans and Rule 1a should be applied. (It is possible, but rare in practice, for some value other than zero to act as a lower bound.) If there is no clear lower bound, then Rule 1b should be applied.

RULE 1a: Double substitution for less-than values with a lower bound

Substitute zero (the lower bound) for each less-than value and perform the desired calculation. Then substitute the face value for each less-than, and perform the calculation again. Both results should be quoted. The true result will lie somewhere between them. Only if their difference is of no practical importance may a single value (halfway between the two calculated results) be used in subsequent work.

The reason for this two-pronged approach is easy to see. The two substitutions produce boundary values that enclose the result that would have been obtained had the actual analytical observations been known. If the difference between the two boundary values is large, then the outcome of the calculation is clearly sensitive to less-than values. This indicates that there may be something seriously wrong with either the choice of the analytical method or the monitoring objective.

Table 1 gives an example showing nine recorded values which include three less-thans, and illustrates the effects of making the double substitution for less-than values when calculating a variety of statistics, namely the mean, standard deviation, standard error of the mean, confidence intervals for the mean and the median. The data has been sorted into ascending order to make it easier to calculate the median. The limit of detection is 0.20 mg/l.

Table 1: Example illustrating the double substitution rule for less-than values

	Values recorded	Values after substitution for <0.20	
Substitution values		0.00	0.20
	<0.20	0.00	0.20
	<0.20	0.00	0.20
	<0.20	0.00	0.20
	0.22	0.22	0.22
	0.25	0.25	0.25
	0.29	0.29	0.29
	0.31	0.31	0.31
	0.42	0.42	0.42
	0.54	0.54	0.54
Total		2.03	2.63
Mean		0.226	0.292
Standard deviation		0.194	0.118
Standard error of mean		0.065	0.039
90% confidence limits on the mean (lower)		0.106	0.219
(upper)		0.346	0.365
Median		0.25	0.25

Note:
The confidence intervals are calculated assuming a Normal distribution. From tables of the t-distribution, the 95% point for 8 degrees of freedom is 1.860

Table 1 shows that the lower estimate for the mean is associated with the higher estimate for standard deviation and vice versa. This will always be the case with any data set involving less-than values.

The table also illustrates the general rule that, if a minority of observations are less-thans, both substitutions give identical results for the median. This useful principle extends to the non-parametric estimation of other percentiles (see CoP/PLE on 'Methods for estimating percentiles'); thus the 95-percentile, for example, will be unaffected unless the vast majority of observations are less-thans.

Where the desired calculation involves taking logarithms of the data, the double substitution method will not work if zero is the lower bound for less-thans. This is because the logarithm of zero is minus infinity, making further calculations impossible. A possible way round this problem is to use a value very close to zero, rather than zero itself, as the lower bound. However, defining the size of the lower bound for substitution is beyond the scope of this Code of Practice, since it depends upon the conditions of each individual case. Indeed, careful consideration should be given as to whether log-transformation is appropriate at all in these circumstances. However, note that the double substitution rule may not be necessary if log-Probit methods can be used (see Rule 3).

RULE 1b: Single substitution for less-than values with no lower bound

Find a one-sided limit to the true result by replacing each less-than value by its face value and performing the desired calculations on the modified data set. Where an arithmetic mean is calculated by this method, the output should state that this is an over-estimate.

Rule 1a cannot be applied to cases where the determinand does not possess a clear lower bound or where the absolute minimum for the determinand could not be considered a realistic lower bound for less-thans. It would be wrong, for example, to use zero as the lower bound for either conductivity or pH in river water.

It is then impossible to find a pair of limits that bracket the true result. Only a single limit can be calculated. Thus, the single substitution rule, Rule 1b, acts as a fall-back position.

Note that this method will give an over-estimate for the mean but an under-estimate for the standard deviation. However, if a minority of observations are less-thans, the true result will generally be obtained for the median. These conclusions are illustrated in the rightmost column of Table 1 where substitution by the face value is shown.

If the face value is the same throughout the data, it should be quoted.

RULE 2: Substitutions for greater-than values

Either Rule 2a or Rule 2b should be applied to greater-thans. Rule 2b should be used in those rare cases where there is an upper bound for the greater-thans. This upper bound is an absolute maximum which it is logically impossible for values of the determinand to exceed. In general, there will be no clear upper bound and Rule 2a should be applied.

There are two cases to consider. In general, Rule 2a is the one to use. However, circumstances do arise, though rare in practice, in which there may be a maximum value which cannot logically be exceeded by the determinand. Rule 2b should then be applied.

RULE 2a: Single substitution for greater-than values with no upper bound

Find a one-sided limit to the true result by replacing each greater-than value by its face value and performing the desired calculations on the modified data set. Where an arithmetic mean is calculated by this method, the output should state that this is an under-estimate.

It is generally impractical to define an upper bound for greater-than values. If so, it is impossible to find a pair of limits that bracket the true result. At best, only a single limit can be calculated. Note that this method will give a under-estimate for the mean and also an under-estimate for the standard deviation.

The face value should be quoted if it is constant throughout the data.

BOD is a common example of a determinand where greater-than values can occur and where the face value may vary from sample to sample.

RULE 2b: Double substitution for greater-than values with an upper bound

Substitute the face value for each greater-than value and perform the desired calculation. Then substitute the upper bound for each greater-than value, and perform the calculation again. Both results should be quoted. The true result will lie somewhere between them.

This case is introduced mainly for completeness, as in practice it is rarely needed. Just occasionally, there may be some logical upper limit to the values of a determinand. For example, the values of some determinands measured in percentages cannot exceed 100%. This upper limit can then be used as the upper bound for greater-thans in the double substitution rule.

RULE 3: Other methods

Always seek advice from a qualified statistician before using any other method for handling less-thans and greater-thans.

For example, the so-called log-Probit method is sometimes used to estimate population parameters for environmental quality data. However, there must be a statistical assessment as to whether there is sufficiently strong evidence that the underlying distribution is, in fact, log-Normal before the method may be used.

RULE 4: Graph plotting

Plot less-than and greater-than values at their face value, using special symbols or colours or both. The recommended symbols are downward-pointing triangles or downward-pointing arrows for less-thans and upward-pointing ones for greater-thans. A key explaining the symbols should be included on the graph.

Rule 4 applies only when plotting the observed data values in, say, a time series or a scatter graph. No general rules can be given for graphs involving derived statistics (such as a series of annual means), but the method of substitution used in calculating the statistics should be indicated in a footnote, if possible.

RULE 5: Notification

In data summaries, always indicate clearly the existence of less-than or greater-than values in the data, and the method and value of any substitution used.

Samples are collected and analysed and summary statistics are calculated in order to provide information that is useful in some way, otherwise the effort is wasted. Potential users of the information may reach invalid conclusions if they are unaware that the data contained less-thans or greater-thans or if they are uncertain how the summary statistics were derived. By implementing Rule 5, the risk of misinterpretation is greatly reduced.

METHODS FOR HANDLING OUTLIERS

This Code of Practice is in four parts. First, Part A lists the rules for dealing with the problem of outliers in data. These are then discussed in detail in the main narrative presented in Part B. Two technical sections then follow. Part C covers methods for determining whether or not a suspected extreme value really is an outlier (so-called 'discordancy' tests). Part D then describes statistical methods for 'accommodating' outliers - that is, for reducing the distorting influence that outliers (whether suspected or confirmed) can have on various types of summary statistics.

PART A - RULES

Definitions

Outlier... a data value which has arisen from some statistical population that is more extreme than the population from which the bulk of the values have arisen.

Suspected outlier... a data value which is so far above or below the bulk of the data values that it causes surprise to the user of the data.

RULE 1: Taking suspected outliers seriously

A suspected outlier may be a pointer to something important, so always try to explain it rather than just discard it as a nuisance.

RULE 2: Checking distributional assumptions

Verify that any assumptions about the probability distribution (e.g. Normal, log-Normal) are justified.

RULE 3: Detecting outliers

Suspected outliers should be investigated using Rules 3a, 3b and 3c.

RULE 3a: Clarity of purpose

Be clear about the purpose of the data analysis, and remove any data values which are not appropriate for that purpose.

RULE 3b: Statistical influence

Assess the 'influence' of the suspected outlier by performing the data analysis twice - once with it included, and again with it excluded. If this makes little difference, then nothing further need be done.

RULE 3c: 'Discordant' data

If the suspected outlier does have a severe effect on the result of an analysis, first see what is revealed by plotting the data. Then, if necessary, use one of the Normality-based or non-parametric 'discordancy' tests described in Part C in order to decide whether or not the point is a genuine outlier.

RULE 4: Excluding outliers

When performing a detailed analysis of a data set containing values that are known to be outliers or have failed a discordancy test, proceed as follows:

(i) exclude the outliers from the analysis and proceed with the remainder of the data as if they hadn't occurred; but also

(ii) add to the report a separate section which:
- lists the outlier dates, times and values;
- explains why they were excluded; and
- indicates any operational changes required to prevent them occurring again.

RULE 5: 'Accommodating' outliers

When routinely obtaining summary statistics for a data source that is known to be subject to an intermittent cause of outliers, 'accommodate' the outliers - either by redefining the model of the process, or by using the so-called 'robust' methods of Part D.

METHODS FOR HANDLING OUTLIERS

PART B - BACKGROUND AND EXAMPLES

B.1 DEFINITION - WHAT IS AN OUTLIER?

Definitions

Outlier... a data value which has arisen from some statistical population that is more extreme than the population from which the bulk of the values have arisen.

There are several ways in which the outlier population might be more extreme than the 'usual' population. For example, it might have a much higher or lower mean. Thus, dissolved oxygen in a river might usually follow a well-behaved distribution, representing stable, routine conditions, but be contaminated by values from a second distribution of much lower mean that applies when storm sewage overflows have been active. Another possibility is where the outlier distribution is characterised by a much higher standard deviation than usual - perhaps due to a less precise analytical method, or the taking of a sample outside the customary restricted time-window.

Suspected outlier... a data value which is so far above or below the bulk of the data values that it causes surprise to the user of the data.

Although the definition of suspected outlier appears rather subjective, it carries with it the implication that the observer must have had some 'correct' probability distribution model in mind (however vague), and believes that the suspected outliers are not consistent with that model. In other words, he or she suspects that the sample has been contaminated by observations from some statistical distribution other than the one expected.

Outliers are also known as 'flyers', 'sports', 'mavericks', 'wild values', 'discordant values', 'anomalous values', and 'rogue points'. In the following discussion we will generally for convenience refer to outliers as being exceptionally high values, but the various methods described apply equally, and can easily be adapted, to the case where the outlier is an exceptionally low value.

B.2 EFFECTS - WHY OUTLIERS MATTER

RULE 1: Taking suspected outliers seriously

A suspected outlier may be a pointer to something important, so always try to explain it rather than just discard it as a nuisance.

At the very least, the presence of an outlier may be indicating vital information about problems with data gathering procedures. At best it may be the one piece of information about some new discovery, as was the case with the single petri dish that led to the discovery of penicillin. This is why an attempt should always be made to explain outliers.

Outliers have a wide variety of unwelcome effects. For example, high outliers can cause:

+ mean, standard deviation and parametric percentile estimates to be inflated;
+ non-parametric estimates of high percentiles and their upper confidence limits to be inflated;
+ parametric confidence intervals to be shifted and inflated;
+ hypothesis tests to give the wrong conclusion;
+ erroneous failures of standards;
+ misleading indications of trend.

For all these reasons, therefore, it is important to have effective methods for first detecting and then coping with outliers.

B.3 CAUSES - HOW AN OUTLIER ARISES

In the Water Industry it is commonplace for data to be stored in a database or 'Archive', and retrieved for analysis at some later date. Inconsistent data values - and hence suspected outliers - can be caused by the database being structured inappropriately or being used incorrectly. Genuine outliers, on the other hand, arise as a result of clerical error, software problems, or poor practice in sample collection, transport, treatment and analysis.

The best way to minimise the chance of outliers reaching the archive is to ensure that Data Quality Control (DQC) procedures are written and implemented. These should lay down standard operational requirements for the whole chain of events from sample collection right through to archiving. Such procedures must clearly state exactly who must do exactly what, where and when. They must be easily accessible, widely communicated, regularly updated and faithfully followed. In addition, DQC should provide for the screening of data values as they are submitted to the archive, in order to maximise the chance of detecting errors (including outliers) which slip through the procedural net.

The issue of DQC procedures in general and data screening in particular are covered fully in the supporting note on Data Quality Control.

RULE 2: Checking distributional assumptions

Verify that any assumptions about the probability distribution (e.g. Normal, log-Normal) are justified.

As stated in the definition, a data value is a suspected outlier when there is a clash between (i) the value observed, and (ii) the data user's model of the process which gives rise to the data.

The user's first thought is that the data value must be at fault. However, it may be that the data is perfectly correct but that his model of the process is wrong, or only approximate. For example, he may have been mistakenly assumed that the determinand in question follows a Normal distribution, whereas the suspected outlier is actually the first evidence that a more appropriate model for the data would be the log-Normal distribution.

It is therefore of great importance to check periodically that any assumption about the distribution of a determinand is justified.

B.4 DIAGNOSIS - DETECTING AND ASSESSING OUTLIERS

RULE 3: Detecting outliers

Suspected outliers should be investigated using Rules 3a, 3b and 3c.

RULE 4a: Clarity of purpose

Be clear about the purpose of the data analysis, and remove any data values which are not appropriate for that purpose.

If insufficient thought has gone into the exact purpose of the data analysis and, as a result, some of the data obtained is not appropriate to that purpose, it is quite possible that some values will appear to be outliers. For example, a data set intended to characterise routine conditions should not include pollution incident data. Always ensure, therefore, that full use is made of the right retrieval mechanisms, such as purpose codes, method codes and lab codes, to fine down the selection.

RULE 4b: Statistical influence

Assess the 'influence' of the suspected outlier by performing the data analysis twice - once with it included, and again with it excluded. If this makes little difference, then nothing further need be done.

Even if the value or values which are arousing suspicion really are outliers, they may make no difference to the conclusion of the analysis. In statistical jargon, their 'influence' may be small. The recommended procedure is therefore to perform the analysis twice - once with the suspected outliers included, and once with them excluded. If the two analyses lead to the same conclusion, or cause the analyst to take the same course of action, then there is little point in considering the matter any further. If, however, the suspected outliers do affect the outcome, Rule 3c should be applied.

RULE 3c: 'Discordant' data

If the suspected outlier does have a severe effect on the result of an analysis, first see what is revealed by plotting the data. Then, if necessary, use one of the Normality-based or non-parametric 'discordancy' tests described in Part C in order to decide whether or not the point is a genuine outlier.

The simple graphical exploration of data - by (for example) time series, scatter plots, histograms, and Normal probability plots - is always a very worthwhile first step. This in itself may be sufficient to demonstrate the genuineness of the outlier, and perhaps even to suggest how it arose (drought year; unusual sampling time; obvious mis-punching). Where more formal evidence is needed, Part C.1 gives details of the recommended discordancy test for the situation where the data (or some transformation, such as the logarithm of the data) is known or can be assumed to be Normally distributed. Where the form of distribution cannot reasonably be assumed, Part C.2 gives a selection of non-parametric and graphical methods which may be tried.

Note that the methods of Part C may be used either in producing an exception report before archiving the data, or as a precursor to the analysis of data after retrieval from the archive. It is also worth remembering, before delving into sophisticated techniques, that simple checks based on the physics and/or chemistry of the determinand in question can often show up errors. For example, pH values greater than 11.0 or temperatures below zero are suspect.

B.5 TREATMENT - WHAT TO DO ON DETECTING AN OUTLIER

RULE 4: Excluding outliers

When performing a detailed analysis of a data set containing values that are known to be outliers or have failed a discordancy test, proceed as follows:

(i) **exclude the outliers from the analysis and proceed with the remainder of the data as if they hadn't occurred; but also**

(ii) **add to the report a separate section which:**
- **lists the outlier dates, times and values;**
- **explains why they were excluded; and**
- **indicates any operational changes required to prevent them occurring again.**

We stated in Rule 1 that outliers should not just be discarded without investigation, because they may be conveying valuable information. For this reason it is important, if suspected outliers have been excluded from a particular analysis, that the results of that analysis are accompanied by a statement of what values were excluded and why.

Valid reasons for exclusion (and separate reporting) of suspected outliers would be that they arose through an intermittent instrument fault, or that they have failed a discordancy test.

RULE 5: 'Accommodating' outliers

When routinely obtaining summary statistics for a data source that is known to be subject to an intermittent cause of outliers, 'accommodate' the outliers - either by redefining the model of the process, or by using the so-called 'robust' methods of Part D.

It sometimes happens that summary statistics are still needed even when the source of the data is well known to be prone to outliers. If investigation has shown that the outlier has arisen because the current model is only approximate, and needs to be made more sophisticated, then it may be worthwhile to update the model, and to revise the calculation methods and reports accordingly. An example of this would be when a Normal distribution model was updated to a log-Normal. If the regular report included an estimate of, for example, the 95%ile, then the procedure for calculating it would also need to be updated accordingly.

If frequent contamination by outliers is anticipated but revision of the distribution model is impractical, use the robust methods of Part D in order to accommodate them. Robust methods are procedures which can be easily applied to every data set obtained from the source, and which reduce the harmful effects of outliers when they are present without causing undue bias in the summary statistics when they are not present.

METHODS FOR HANDLING OUTLIERS

PART C - 'DISCORDANCY' TESTS

C.1 TESTING FOR DISCORDANCY WHEN THE STATISTICAL DISTRIBUTION MODEL CAN REASONABLY BE ASSUMED

The general testing procedure

When the probability distribution of the data can be assumed, the general procedure for testing whether a particular data value is an outlier is as follows:

Suppose that n random samples have been taken from the assumed distribution, and the maximum of those n data values is suspiciously large•. That value should be declared to be an outlier only if its probability of occurrence is small when referred to the distribution of the quantity:

'The maximum out of n values drawn at random from the assumed distribution'.

The recommended test assuming Normality

The following test may be used either:

(i) when the assumed underlying distribution is Normal; or
(ii) when the data can be transformed (e.g. by taking logarithms) so as to make it Normal.

The test statistic is:

$$t_{max} = (x(n)-\bar{x})/s,$$

where $x(n)$ is the maximum of the n data values,

$\bar{x}$ is the mean of the data values, and

s is the standard deviation of the data values.

Critical values of the test statistic are given in Table C.1.

• As mentioned earlier, we will assume for the sake of illustration that the outlier is an unusually high data value. Exactly the same approach can be used when testing for low outliers, with only minor changes of detail needed to the method.

Table C.1 - Critical values of the statistic t_{max}

No. of values	Critical values 5%	1%
4	1.46	1.49
5	1.67	1.75
6	1.82	1.94
7	1.94	2.10
8	2.03	2.22
9	2.11	2.32
10	2.18	2.41
12	2.29	2.55
14	2.37	2.66
15	2.41	2.71
16	2.44	2.75
18	2.50	2.82
20	2.56	2.88
30	2.74	3.10
40	2.87	3.24
50	2.96	3.34
60	3.03	3.41
80	3.13	3.53
100	3.21	3.60
120	3.27	3.66

Further details on this and other tests of this type may be found in the key work 'Outliers in Statistical Data', by Barnett and Lewis.

Worked Example

The following example illustrates the use of the procedure.

Suppose we wish to estimate the mean BOD concentration for a particular effluent over the last three years, and have retrieved the relevant data from the archive. The resulting 60 values, ranked into increasing order, are listed below:

1.3	1.4	1.5	1.6	1.6	1.7	1.8	1.8	1.8	1.9
2.1	2.1	2.3	2.4	2.5	2.5	2.7	2.9	3.3	3.3
3.5	3.5	3.6	3.6	3.7	3.7	3.8	3.8	3.9	3.9
3.9	4.0	4.0	4.1	4.4	4.4	4.8	4.9	5.1	5.3
5.4	5.4	5.6	5.7	5.8	5.8	5.8	6.2	6.9	7.4
7.4	7.5	7.5	7.9	8.0	8.3	8.4	9.1	21.6	28.2

Suppose we are suspicious of the maximum of these 60 values, namely 28.2 mg/l. As effluent BOD is commonly found to be approximately log-Normally distributed, it is appropriate to carry out the test on the logarithms of the data. So the first step is to log each of the 60 values. Using logs to base 10,

this gives:

0.114 0.146 0.176 etc ...
... 0.959 1.334 <u>1.450</u>

These 60 logged values have a mean of 0.603 and a standard deviation 0.271, whilst the suspected outlier is 1.45. The test statistic is therefore:

$$t_{max} = (1.450 - 0.603)/0.271 = 3.13.$$

Consulting Table C.1 we find that the 5% and 1% critical values for 60 values are 3.03 and 3.41. The observed value, 3.13, is therefore significant at the 5% level but not the 1%. In other words, the 28.2 mg/l BOD value does appear to be unusually high, but not overwhelmingly so.

Applying the test to multiple suspected outliers

If multiple outliers are present, the effectiveness of a single-outlier test will be weakened: there is a 'masking' effect when two large outliers are close together which hides one of them. If, on the other hand, a 'block' test is used to test specifically for, say, two outliers, this is liable to reach a significant conclusion even when only a single large outlier is present - a phenomenon called 'swamping'.

The best approach, therefore, when the data contains an unknown number of suspected outliers, is the 'outward consecutive' method. With this, an upper limit k on the number of possible outliers is specified, and the suspects are then tested one at a time, working from the least to the most extreme. The details of the procedure are as follows:

> Starting with the (n-k) 'reliable' data values, augment these by just the least extreme of the k suspected outliers. Calculate the mean and standard deviation of those (n-k+1) values, and perform the single-outlier test as usual. If the suspect fails to be confirmed as an outlier, pool it with the (n-k) reliable values, recalculate the mean and standard deviation, and then test the next least extreme suspect. Continue in this way until a suspected outlier is declared to be a genuine outlier. All the remaining (and hence more extreme) values are then declared to be outliers also.

It is recommended that an upper ceiling of n/10 be placed on the maximum number of possible outliers, k.

It might be thought that, as the multiple-outlier test provides several opportunities for false positives, the actual significance levels would be somewhat higher than the nominal values quoted in Table C.1 for the single-outlier case (viz 5% and 1%). That is not the case, however. When the data values really do come from a Normal population, the multiple-outlier test very rarely produces false positives unless the single-outlier test does so also - a characteristic that we have confirmed by computer simulation.

The procedure described above has been produced as a computer routine, MOT, and this is used as the worked example in Code of Practice CoP/TDFd on 'Developing software for the Test Data Facility'.

C.2 TESTING FOR DISCORDANCY WHEN THE STATISTICAL DISTRIBUTION MODEL CANNOT REASONABLY BE ASSUMED

General principles

If our model of the process is too vague, we cannot use the methods of Part C.1. Indeed, some writers insist that if there is no model, there can be no such thing as an outlier.

What we can do, however, is to make use of more empirical methods such as range checks, and graphical methods which check in some way that current data is at least reasonably consistent with past data.

Range checks

A range check is established by choosing upper and lower values, and arranging for a warning or 'exception report' to be issued whenever a new data value falls above the upper or below the lower limit. As specified in the Data Quality Control procedures, the exception report should then be passed to the appropriate officer. It is his or her responsibility to find any error and correct it, and then sign off the result if it is valid, or reject it if it is invalid.

There are a variety of ways of choosing the parameter values for the range checks. Four possible approaches are described below.

Range checks set by experience

The upper and lower values may be initially set on the basis of experience, or on the basis of what appears to make physico/chemical sense. When the range check is set in this way, values are likely to arise, sooner or later, that are outside the range but are nevertheless subsequently verified as being valid. When that happens, it is vital that the upper/lower range check parameters are updated accordingly. It is the failure to do this in the past which has so often led to range checks becoming distrusted and eventually ignored or overridden.

Conversely, if it becomes apparent that the upper and/or lower limits are insufficiently stringent, they should likewise be revised accordingly.

Range checks using minimum and maximum of past data

If we happen to have a past data set that we believe to be representative of the population of interest and to contain no outliers, we can use this to derive the range check. One approach is simply to take the minimum and maximum values from this data set, and use these as our range check parameters. The disadvantage of this method (and the preceding method), however, is the lack of information about the likelihood of false alarms.

Range checks based on estimated percentiles

A better way of deriving an upper range check, which does give a known probability of a false alarm, is as follows.

+ Select a suitably high percentage point, P (e.g. 99.5)
+ Calculate n such that the maximum of n random samples is a non-parametric estimate of the P-percentile.
+ Then use the maximum of n values randomly selected from valid past data to define the upper warning limit.

For example, using the Weibull convention (see the Code of Practice CoP/PLE on 'Methods for estimating percentiles') the largest of 199 randomly selected values is an estimate of the 199/(199+1) = 99.5%ile of the distribution. Thus, from 199 randomly selected values of valid past data, the largest value could be used as the upper warning limit. This would ensure that, in the long run over many data sets, there was a probability of only 0.5% of issuing an exception report when a data value was actually valid.

Lower range checks may be derived in a similar way.

Range checks based on confidence limits for percentiles

This is a slightly more complex variant of the previous method which builds in a greater safety margin. The scheme is to arrange that the maximum of n values, rather than estimating the P-percentile, is some suitable upper confidence limit on the P-percentile. For example, the largest of 460 randomly selected values from a distribution is a non-parametric upper 90% confidence limit for the 99.5%ile of that distribution. By this method, therefore, 460 observations are selected at random from validated past data and the largest value is used as the upper warning limit.

Now our guarantee is this. Even if our initial choice of 460 values were to contain unluckily few high values, we would still be 90% confident that the percentile estimated by the maximum was at least as extreme as the 99.5%ile. Thus we would be 90% confident that the chance of wrongly issuing an exception report when a high data value really was valid was no more than 0.5%.

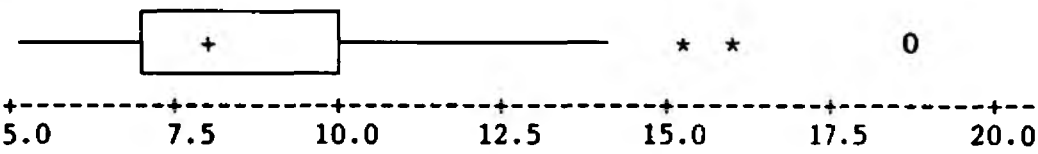
Again, lower range checks may be derived in a similar way.

Graphical methods

A variety of graphical summary methods can be used to pick out those values which in some respect stand apart from the bulk of the data.

The Box and Whisker plot

The 'boxplot' is a specialised diagram for displaying the main features of a set of data. It is capable of being produced by most statistical software packages. The technique is presented here for its particular use in highlighting suspected outliers.



The 'box' represents the middle 50% of the data - that is, the interval from the 25%ile to the 75%ile; within this, the median is marked with the symbol '+'. The 'whiskers' then indicate the spread of the data beyond either edge of the box. With a well-behaved data set, the left-hand whisker would run from the left edge of the box to the minimum data value, whilst the right-hand whisker would run from the right edge of the box to the maximum value.

If, however, the data should contain suspiciously extreme values - as defined by the various criteria built into the boxplot calculation procedure - the whiskers stop short of the minimum and/or maximum. The suspect values are then plotted separately beyond the whiskers using special symbols to indicate the location of '<u>possible</u>' outliers ('*') or '<u>probable</u>' outliers' ('O').

It should be noted that the boxplot definitions of possible and probable outliers are based on empirical rules of thumb rather than specific distributional assumptions. The flagging of such data values in a boxplot should therefore be regarded as supporting rather than definitive evidence.

Other plots

The supporting note on Data Quality Control gives a further selection of graphical methods which may be used to detect outliers (as well as other types of error).

METHODS FOR HANDLING OUTLIERS

PART D - 'ACCOMMODATION'

D.1 ACCOMMODATION WHEN ESTIMATING THE MEAN AND STANDARD DEVIATION

Estimating the mean

With the discordancy tests described in Part C, an observation which the test decrees to be an outlier is excluded from the routine analysis and is reported separately.

The situation in which accommodation is called for is different, and is something of a last resort. Here, we are in the position of knowing that data sets are quite likely as a general rule to contain outliers, but we do not wish - or perhaps are not able - to inspect each data set in detail. What we wish to do is to estimate the mean of the underlying distribution in a way that will minimise the ill effects of any outliers that may have arisen from some contaminating distribution.

The recommended procedure in this case is the technique known as 'trimming'. In its simplest form this consists simply of omitting the minimum and maximum values from the data set, and then using the arithmetic mean of the remaining values to provide a 'trimmed' estimate of the mean.

Depending on the expected degree of contamination, it may be advisable to trim to a greater extent - that is, to remove not just one but several values from either end. In this case, however, the advice of a statistician should be sought.

A frequently used alternative to trimming is 'Winsorising', whereby extreme values are replaced by their nearest neighbours (rather than simply being omitted). We mention the technique because some readers might encounter it in statistical packages. Again, this is a somewhat specialised procedure that should not be used without statistical guidance.

Worked example

Suppose we have the following 20 sample values:

23.2, 23.4, 23.4, 23.5, 24.1, 25.5, 25.5, 27.0, 27.5, 28.1,
28.8, 29.1, 29.9, 31.5, 33.1, 33.4, 33.8, 44.2, 44.2, 75.5

Using trimming, the data set would be modified to become:

- 23.4, 23.4, 23.5, 24.1, 25.5, 25.5, 27.0, 27.5, 28.1,
28.8, 29.1, 29.9, 31.5, 33.1, 33.4, 33.8, 44.2, 44.2 -

Thus the trimmed estimate of the mean is:
$(23.4 + 23.4 + \ldots + 44.2)/18 = 29.8$.

If the value 75.5 was not an outlier, then by discarding values at both ends of the sample the net effect on the mean should be more or less neutral. If, however, 75.5 was an outlier, then we have successfully eliminated it, at the minor cost of also losing the smallest data value. In the long run, the effect of this will be to introduce a slight upwards bias in the estimate of the mean.

Estimating the standard deviation

We can similarly use the trimmed data set to estimate the standard deviation. However, this will tend to under-estimate the true standard deviation whenever the data in fact contains no outliers. The bias is more pronounced than in the case of the mean, and so it is advisable to apply a correction factor. As Table D.1 shows, the factor depends on the number of values in the (untrimmed) data set: with only 10 values, the trimmed standard deviation estimate must be increased by the substantial figure of 1.37; whilst with 50 samples, the estimate needs to be inflated by less than 10%.

Table D.1 - Bias-correction factors for trimmed standard deviation estimates

No of values	Factor	No of values	Factor	No of values	Factor
10	1.37	18	1.21	36	1.12
11	1.34	20	1.20	40	1.11
12	1.31	22	1.18	45	1.10
13	1.29	24	1.17	50	1.09
14	1.27	26	1.16	60	1.08
15	1.25	28	1.15	70	1.07
16	1.24	30	1.14	80	1.06
17	1.23	33	1.13	100	1.05

Worked example

For the trimmed data used in previous example, the standard deviation is 6.26. As the number of values is 20, a factor of 1.20 should be used. Thus the corrected estimate of the standard deviation is 7.51.

D.2 ACCOMMODATION WHEN ESTIMATING A PERCENTILE

Parametric estimates

When trimmed estimates of the mean and standard deviation of the underlying distribution have been obtained, these may be inserted into the relevant expression (see CoP/PLE on 'Methods for estimating percentiles') to calculate the estimate of any required percentile.

Non-parametric estimates

Because non-parametric methods are based on ranking the data, they are relatively immune to outliers in all but the smallest samples (or the most extreme percentiles). This is one of the main reasons to prefer non-parametric estimates of percentiles.
For example, in a sample of 19 values containing a single outlier, the estimate of any percentile equal to or lower than the 90th will be unaffected. This is because, using the Weibull convention, the estimate of the 90%ile is given by the 0.90*(19+1)th value - that is, the 18th.

Non-parametric methods are sometimes criticised for the fact that when estimating a high percentile with a small sample size, a single outlier can itself become the estimate. For example, in estimating the 95 percentile from just 19 sample values, the estimate is the 0.95*(19+1)th = the 19th largest value. If the largest of the 19 values is an outlier, therefore, the non-parametric estimate of the 95%ile will be that outlier.

In answer to this criticism, the following points are relevant:

(i) A parametric estimate would also be wrong.

(ii) If genuine outliers are so common that they have a significant impact on, for example, estimated river class across a whole region, then the most important message is not that the method of estimating percentiles should be changed, but that the implementation of DQC procedures should be a high priority.

(iii) Reiterating an earlier point, it may actually be that the data values which the analyst thinks may be outliers are in fact valid, and that his perception of the process giving rise to the data is at fault.

METHODS FOR ESTIMATING PERCENTILES

This Code of Practice gives a set of rules for estimating percentiles. The rules are stated and illustrated here in Part A without elaboration. Part B then discusses the rules in more detail. Finally, Part C provides the statistical background needed in applying certain of the rules.

PART A - RULES

Definition

The P-percentile... the value below which P% of all possible values from a particular statistical population fall. A convenient and common shorthand for the 'P-percentile' is the 'P%ile'.

For example, the 95%ile ammonia concentration at a particular river sampling point for 1990 is the value that was met or bettered for 95% of the time by ammonia concentrations at that point of the river during 1990. Equivalently, it is the concentration value that was exceeded for 5% of the time.

Note that the true 95-percentile can never be known except by continuous, error-free monitoring; it can only be estimated from a set of sample values.

RULE 1: Need for consistency

Whenever percentile estimates are likely to be used to provide a national assessment, or for inter-regional comparisons, consistency between NRA regions in the derivation and interpretation of those percentiles is of paramount importance.

RULE 2: Check that a percentile is actually needed

If the reason for calculating a percentile is solely to test for compliance with a percentile standard, it is usually preferable to carry out such assessments in terms of the number (or percentage) of excedences. Thus in some situations it may not actually be necessary to calculate a percentile at all.

RULE 3: Estimation method used

The statistical method used for estimating percentiles should always be stated explicitly. Provided sufficiently many samples are available (see Rule 6), a non-parametric method is preferable - especially for general-purpose reporting (see Rule 4a). In more specialised applications, or as a fall-back position in cases where the data is limited, a parametric method may be used (see Rule 4b).

RULE 4a: Non-parametric methods

The recommended non-parametric method for estimating percentiles is the 'Weibull' method. Using this, the P-percentile is estimated by the 'r-th' value out of n random sample values sorted into increasing order, where r = (P/100)(n+1).

If (as will generally be the case) r is not an exact whole number, the estimate should be obtained by linear interpolation between the relevant pair of neighbouring values.

RULE 4b: Parametric methods

Where a parametric method is used, any evidence for the assumed distributional model should be indicated in a footnote. In the absence of specific evidence, the log-Normal assumption will usually provide an adequate approximation for most common determinands. For DO, pH and temperature, however, the assumption of Normality is more appropriate.

The most appropriate formulas to use in various circumstances are are presented and illustrated in Part B.

RULE 5: Confidence limits

The NRA's monitoring objectives often call for the estimation of extreme percentiles such as the 95%ile and beyond from limited numbers of samples. Percentile estimates in such circumstances can be subject to considerable statistical uncertainty, and so it is especially important to accompany them with confidence limits. For routine applications, a confidence level of 90% is recommended (see Code of Practice CoP/SSS on 'Presenting summary statistics').

Where only approximate confidence limits can be calculated, these are wholly acceptable as a substitute provided that the captions are amended accordingly.

RULE 6: Amount of data required for the Weibull method

The more extreme is the required percentile in relation to the median (50-percentile), the greater is the amount of data needed before the Weibull method can be used. If Q is the numerical difference (ignoring sign) between P and 50, then the minimum number of samples needed to estimate the P-percentile is given by (50+Q)/(50-Q). For the 95%ile (or the 5%ile), for example, Q is 45 and so the minimum number of samples is 95/5 = 19.

A Weibull percentile estimate will be more reliable - particularly where the data may contain occasional suspected outliers - if its calculation makes no direct use of the maximum (or minimum) data value. To achieve this, the number of samples must be at least (150+Q)/(50-Q). For the 95%ile, this gives a target sample number of 195/5 = 39.

PART B - BACKGROUND

RULE 1: Need for consistency

Whenever percentile estimates are likely to be used to provide a national assessment, or for inter-regional comparisons, consistency between NRA regions in the derivation and interpretation of those percentiles is of paramount importance.

Whilst this rule applies to any type of summary statistic, it is particularly important where percentiles are concerned - for two main reasons:

(i) several different percentile estimation methods are in common use, each relying on a particular set of statistical assumptions, and the answer will depend on which is chosen;

(ii) whatever method is used, the uncertainty in the estimate will in many cases be considerable, given the amount of data typically available.

RULE 2: Check that a percentile is actually needed

If the reason for calculating a percentile is solely to test for compliance with a percentile standard, it is usually preferable to carry out such assessments in terms of the number (or percentage) of excedences. Thus in some situations it may not actually be necessary to calculate a percentile at all.

Suppose, for example, that a 90%ile limit has a numerical value of 6 mg/l. One way of judging compliance with this limit, given a set of sampling data, would be to estimate the 90%ile and see whether or not the estimate was greater than 6 mg/l. As we note above, however, percentile estimation is complicated by the variety of methods available and the differing assumptions they make. Further technical issues are raised if, additionally, the question of whether the estimate is significantly greater than 6 mg/l is to be addressed. Such complications are unwelcome in a legal setting.

If, in contrast, the assessment is based on the number of excedences of the standard in relation to the total number of samples taken, matters are much simpler. Suppose, for example, that the five largest values in 24 samples are 5.4, 5.5, 6.8, 7.2 and 47.1. We see from these that there are three excedences of the 6.0 mg/l limit out of the 24 samples; and those two numbers - 3 and 24 - are all that is needed for the assessment. This is true whether compliance is to be judged:

(a) by a 'P% of samples' rule - as it is with the majority of EC Directives; or

(b) with an appropriate allowance made for sampling variability - as in the case of the DoE effluent compliance Look-up Table, and, more recently, the Urban Waste Water Treatment Directive.

In the latter case, the statistical details are unambiguous and universally applicable; there is also the advantage that the approach can be applied however few samples are available.

RULE 3: Estimation method used

The statistical method used for estimating percentiles should always explicitly be stated. Provided sufficiently many samples are available (see Rule 6), a non-parametric method is preferable - especially for general-purpose reporting (see Rule 4a). In more specialised applications, or as a fall-back position in cases where the data is limited, a parametric method may be used (see Rule 4b).

The various recommended estimation methods are illustrated below with the help of a worked example based on data presented in Part C. The river quality data set summarised contains 67 values of DO(%) and BOD covering a three-year period.

RULE 4a: Non-parametric methods

The recommended non-parametric method for estimating percentiles is the 'Weibull' method. Using this, the P-percentile is estimated by the 'r-th' value out of n random sample values sorted into increasing order, where r = (P/100)(n+1).

If (as will generally be the case) r is not an exact whole number, the estimate should be obtained by linear interpolation between the relevant pair of neighbouring values.

Suppose we wish to estimate the 95%ile from the 67 BOD values listed in Part C. The Weibull formula gives r = 0.95(68) = 64.6. This is interpreted as 0.6 of the way between the 64th and 65th ordered data values (6.1 and 6.4 respectively). Thus the estimated 95%ile is 6.1 + 0.6(6.4-6.1) = 6.28 mg/l.

It is worth mentioning in passing that the Weibull is not the only possible non-parametric approach. As we discuss briefly in Part C, there are many other options, each with their various statistical strengths and weaknesses. We recommend the Weibull, nevertheless, for a combination of reasons:

+ through wide use in the water industry, it has become well established over many years of water quality applications;
+ its main statistical strength - biasedness in probability terms (see Part C)- is particularly relevant to compliance applications; and
+ it is easy to understand and use.

RULE 4b: Parametric methods

Where a parametric method is used, any evidence for the assumed distributional model should be indicated in a footnote. In the absence of specific evidence, the log-Normal assumption will usually provide an adequate approximation for most common determinands. For DO, pH and temperature, however, the assumption of Normality is more appropriate.

The most appropriate formulas to use in various circumstances are presented and illustrated in Part B.

Normality

Where Normality can be assumed, the formula to use is:

P%ile = mean + u(st.dev.) .....(1)

where u is the standard Normal deviate cutting off a cumulative probability of P%. For the 95%ile, for example, u is 1.645; for the 5%ile, u is -1.645.

We can illustrate the approach using the 67 DO values listed in Part C. For this data the mean is 66.8 and the standard deviation is 19.0. The 5%ile estimate is therefore:

5%ile = 66.8 - 1.645(19.0) = 35.5 % sat.

log-Normality

Where log-Normality can be assumed, there are two options. These are known as '**Maximum Likelihood**' (MaxL) and '**Method of Moments**' (MofM). According to standard statistical theory (see for example Aitchison and Brown), the MaxL method is the better choice. This involves taking logs of each raw data value, then applying formula (1) above, and finally anti-logging the result to return to the unlogged world.

MaxL

We can illustrate the approach using the BOD data again. (In practice it is more common to use logs to base 10, but here we will use logs to base e so that this example can be compared directly with the MofM example that follows.)

Summary statistics for the logged data (using base e) are:

mean = 1.153
and st.dev. = 0.376

and so 95%ile = 1.153 + 1.645(0.376) = 1.772,
which, when antilogged, gives a 95%ile estimate of 5.88 mg/l.

One practical problem arises with the MaxL approach when there happen to be values very close to zero: these, when logged, can produce large negative values which inflate the estimate of the standard deviation, thereby overstating the percentile estimate. The MofM approach avoids this problem by starting with the summary statistics for the raw data, and estimating the mean and standard deviation of the logged data indirectly via certain mathematical relationships that apply for the log-Normal distribution.

MofM

To repeat the BOD example, therefore, we find that for the raw data:

mean = 3.40 mg/l
and st.dev. = 1.38 mg/l.

Next we obtain estimates of the logged-data summary statistics using the equations given in Part C. These give:

mean = 1.148
and st.dev. = 0.391.

Now we proceed as before with formula (1). Thus:

95%ile = 1.148 + 1.645(0.391) = 1.790,
which, when antilogged, gives a 95%ile estimate of 5.99 mg/l.

The problem caused by near-to-zero data values is particularly likely to arise with ammonia concentrations in good-quality rivers. It is for this reason that the NRA guidelines for the analysis of the 1990 River Quality Survey data stipulated that percentiles for chemical quality data were to be estimated using Method of Moments.

Readers with access to suitable distribution-fitting software may find it instructive to examine a few river quality data sets and see how much or little difference there is, in cases where the log-Normal model gives a reasonable representation of the data, between the MaxL and MofM estimation methods.

RULE 5: Confidence limits

The NRA's monitoring objectives often call for the estimation of extreme percentiles such as the 95%ile and beyond from limited numbers of samples. Percentile estimates in such circumstances can be subject to considerable statistical uncertainty, and so it is especially important to accompany them with confidence limits. For routine applications, a confidence level of 90% is recommended (see Code of Practice CoP/SSS on 'Presenting summary statistics').

Where only approximate confidence limits can be calculated, these are wholly acceptable as a substitute provided that the captions are amended accordingly.

Formulas for calculating confidence limits for percentiles are detailed in Part C. Those for non-parametric limits are exact; those given for parametric cases are approximate.

Whichever formula is used, the limits can be tedious to calculate manually. Appropriate software is needed, therefore (whether as stand-alone routines or via summary options in the quality archive), to enable NRA river quality assessment staff to develop a practical appreciation of the uncertainty associated with percentile estimation. One example of such a program is ARCTIC_SEAL; this is available as the Test Data Facility procedure ARC.

RULE 6: Amount of data required for the Weibull method

The more extreme is the required percentile in relation to the median (50-percentile), the greater is the amount of data needed before the Weibull method can be used. If Q is the numerical difference (ignoring sign) between P and 50, then the minimum number of samples needed to estimate the P-percentile is given by (50+Q)/(50-Q). For the 95%ile (or the 5%ile), for example, Q is 45 and so the minimum number of samples is 95/5 = 19.

A Weibull percentile estimate will be more reliable - particularly where the data may contain occasional suspected outliers - if its calculation makes no direct use of the maximum (or minimum) data value. To achieve this, the number of samples must be at least (150+Q)/(50-Q). For the 95%ile, this gives a target sample number of 195/5 = 39.

Although the Weibull method does technically work for the bare minimum number of samples, viz (50+Q)/(50-Q), it is strongly influenced with this few samples by any occasional 'flyer' that turns up in the data. Instances of this were seen, for example, in the data analysis for the 1990 River Quality Survey when the Weibull method was applied to data sets containing between 20 and 38 samples.

Of course, it can be argued that if uncomfortably extreme values become a regular occurrence, they are a valid part of the population being sampled and so it is right that they should influence the percentile estimate. (See the discussion in Code of Practice CoP/OLR on 'Methods for handling outliers'.) Nevertheless, it is sensible to try to lessen the influence that dubious extreme data values exert on the estimate. One practical way of achieving this is to ensure that the number of samples is sufficiently large for the Weibull estimate to involve no value greater than the second-biggest. This leads to the result given in Rule 6.

The ratio of 'robust minimum' to 'bare minimum' number of samples is (150+Q)/(50+Q). For estimating extreme percentiles such as the 90%ile and beyond, therefore, the target number of samples is roughly double the bare minimum number.

REFERENCES

AITCHISON J and BROWN J A C (1969) The logNormal distribution. Cambridge University Press.

PART C - TECHNICAL DETAILS

The following two sections give details of methods for:

(i) estimating the P-percentile (P%ile) from n random samples; and
(ii) calculating an approximate 90% confidence interval around that P%ile estimate.

First, Section C.1 describes the non-parametric Weibull method. Section C.2 then describes various parametric methods assuming Normality or log-Normality. Listings of the data used for the worked examples given here and in Part B are provided in Section C.3. Finally, Section C.4 provides a wider discussion of non-parametric methods in general, and the merit of the Weibull method in particular.

C.1 FORMULAS FOR ESTIMATION AND CONFIDENCE LIMITS - NON-PARAMETRIC APPROACH

First, the n sample values should be sorted into increasing order. Let the sorted values be denoted by x(1), x(2),... x(n).

Next, calculate:

$p = P/100,$

$q = p(n+1),$

$r =$ integer part of q, and

$d = q - r.$

The P%ile can then be estimated by:

Estimate P%ile = $[1-d].x(r) + d.x(r+1)$, or, equivalently,
$x(r) + d.[x(r+1) - x(r)]$.

(Note: for certain combinations of P and n, no solution is possible.)

Now to calculate non-parametric confidence limits, we first need the binomial probabilities for the binomial distribution B(n,p). Suppose these are denoted by Pr(0), Pr(1),..., Pr(n). Each term can be calculated from:

$$Pr(r) = \frac{n!}{(n-r)!r!}p^r(1-p)^{(n-r)}, \quad r = 0,...n.$$

Next, the two integers v_{max} and w_{min} are required:

(i) v_{max} is defined as the maximum value of v such that

$$\sum_{i=0}^{v} [\ Pr(i)\ ] \text{ is } \leq 0.05.$$

(For low percentiles, even a value of v as little as 0 may not satisfy this inequality.)

(ii) w_{min} is defined as the minimum value of w such that

$$\sum_{i=0}^{w} [\ Pr(i)\ ] \text{ is } \geq 0.95, \text{ and}$$

$$w_{min} < n.$$

(For high percentiles, it is not always possible to find a value of w less than n.)

The confidence limits can then be estimated (assuming both v_{max} and w_{min} can be obtained) by the following order statistics:

Lower 90% conf. limit = $x(v_{max}+1)$, and

Upper 90% conf. limit = $x(w_{min}+1)$.

Example

In the Part B example, the Weibull 95%ile estimate for BOD was based on 67 values. With n = 67, the above inequalities lead to v and w values of 60 and 66. The confidence limits for 95%ile BOD are therefore the 61st and 67th ordered values - that is, 5.0 and 8.5.

C.2 FORMULAS FOR ESTIMATION AND CONFIDENCE LIMITS - PARAMETRIC APPROACH

Assuming Normality

For a Normal distribution with mean μ and standard deviation σ, the P%ile is given by:

P%ile = $\mu + u\sigma$,

where u is the standard Normal deviate at cumulative probability p = P/100. Values of u corresponding to common P values are:

P:	5	50	90	95	99
u:	-1.645	0.000	1.282	1.645	2.326

Suppose the n sample values have mean m and standard deviation s. The P%ile is then estimated by:

P%ile = m + us .....(1)

The calculation of exact confidence limits goes beyond the scope of this Code of Practice. The approximate limits given below, however, will be adequate for most purposes.

First, calculate the approximate standard error of P%ile by:

E = s.h.V[1/n], where h is the appropriate factor from Table C.1.

An approximate confidence interval for P%ile is then given by:

P%ile ± tE,

where t is the value of Student's t corresponding to the desired confidence level and degrees of freedom. For example, with 21 data values, there are 20 degrees of freedom, and so the t values for 90%, 95% and 99% confidence (obtained from any book of statistical tables) are 1.72, 2.09 and 2.85.

Note that the question of Maximum Likelihood versus Method of Moments does not arise when Normality is assumed, as the two statistical approaches both arrive at the same values of m and s.

Table C.1 - Approximate factors for calculating the standard error of percentiles estimated assuming Normality

Percentile to be estimated :	50	40 60	30 70	20 80	10 90	5 95	1 99
Factor :	1.00	1.02	1.07	1.19	1.42	1.64	2.10

Note: The derivation of the factors in this table is discussed in WRc's Sampling Handbook.

Example

In the Part B example, the 5%ile estimate for DO was 35.5, based on 67 values with mean and standard deviation 66.8 and 19.0. The quantity E is therefore:

19.0(1.64)V[1/67] = 3.81.

Also, Student's t for 90% confidence and 66 degrees of freedom is 1.67. An approximate 90% confidence interval is therefore:

35.5 ± 6.36

viz 29.1 to 41.9.
---- ----

Assuming log-Normality

Suppose a determinand x follows a log-Normal distribution with parameters μ and σ. This is equivalent to saying that the quantity **ln(x)** follows a **Normal** distribution with mean μ and standard deviation σ. ('ln' denotes 'log to base e'.) This forms the basis of the **Maximum Likelihood** approach described below.

The population mean (M) and standard deviation (S) of x itself can be expressed in terms of μ and σ as follows:

$$M = EXP(\mu + 0.5\sigma^2) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

$$\text{and} \quad S/M = \sqrt{[EXP(\sigma^2) - 1]} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

These relationships form the basis of the Method of Moments approach also described below.

Maximum likelihood

First, transform the original data values by taking logs. (Either base 10 or base e can be used.) Then for the logged data calculate the mean m and standard deviation s.

Next, simply follow the Normal-based procedure described in the previous section and calculate the P%ile estimate and its associated confidence limits in the 'log world'.

Finally, antilog the P%ile estimate and its confidence limits to translate them back to the 'unlogged world' of the original data.

Example

Continuing with the Part B example, there are 67 BOD data values, and ln(BOD) has mean 1.153 and standard deviation 0.376. This gives a 95%ile estimate in the log-e world of 1.154 + 1.645(0.376) = 1.772. Moreover, the quantity E is:

$$0.376(1.64)\sqrt{[1/67]} = 0.0753.$$

Also, Student's t for 90% confidence and 66 degrees of freedom is 1.67. Thus an approximate 90% confidence interval in the log-e world is:

$$1.772 \pm 0.126$$

viz 1.646 to 1.898; and these, when antilogged, give:

5.19 to 6.67.

Method of Moments

The MofM approach avoids the need to log the raw data. Instead, estimates of μ and ó are obtained by inverting relationships (2) and (3) above as follows:

Suppose that $\bar{X}$ and C are the observed mean and coefficient of variation (= standard deviation/mean) of the raw data. Then, using logs to the base e, calculate:

$s = \sqrt{\ln[1 + C^2]}$, and

$m = \ln(\bar{X}) - 0.5s^2$.

(Note that here, in contrast to the MaxL case, logs to base e rather than 10 must be used for the above expressions to hold.)

Now follow the same procedure as in the MaxL case. That is, use the Normal-based method for calculating the P%ile estimate and its associated confidence limits in the 'log world'. Then antilog those quantities to translate them back to the 'unlogged world'.

Example

For the unlogged BOD data, the mean, X, is 3.40 and the coefficient of variation, C, is 1.38/3.40 = 0.406. Thus:

$s = \sqrt{[\ \ln(1 + 0.165)\ ]} = \sqrt{(0.152)} = 0.391$, and
$m = \ln(3.40) - 0.5(0.165) = 1.148$.

This gives a 95%ile estimate in the log-e world of 1.148 + 1.645(0.391) = 1.772. Moreover, the quantity E is:

$0.391(1.64)\sqrt{[1/67]} = 0.0783$.

As before, Student's t for 90% confidence and 66 degrees of freedom is 1.67. Thus an approximate 90% confidence interval in the log-e world is:

1.772 ± 0.131

viz 1.641 to 1.903; and these, when antilogged, give:

5.16 to 6.71.

C.3 DATA USED IN THE WORKED EXAMPLES

DO (% sat): ranked data...

17.2	28.0	28.4	32.1	38.0	48.0	48.6	49.0	49.0	49.7
50.0	51.0	53.3	53.5	54.0	54.0	54.0	56.4	57.0	57.0
58.0	60.0	60.0	61.0	61.0	62.0	62.0	62.0	64.0	64.0
66.0	66.0	67.0	67.0	68.0	68.0	68.0	70.0	70.0	70.0
71.0	71.0	72.0	73.0	73.0	73.0	74.0	74.0	76.7	77.0
77.0	78.0	78.0	80.0	80.0	80.2	82.0	82.0	84.0	86.0
87.0	89.0	92.0	96.0	104.0	120.0	126.0			

BOD (mg/l): ranked data...

1.5	1.5	1.7	1.7	1.7	1.8	1.9	2.0	2.0	2.0
2.1	2.1	2.3	2.4	2.4	2.5	2.5	2.5	2.6	2.7
2.7	2.7	2.8	2.8	2.8	2.8	2.9	2.9	2.9	3.0
3.0	3.2	3.2	3.2	3.2	3.3	3.3	3.3	3.4	3.4
3.4	3.4	3.5	3.5	3.6	3.7	3.7	3.7	3.8	3.8
3.9	3.9	4.3	4.4	4.4	4.6	4.7	4.7	4.8	4.9
5.0	5.2	5.5	6.1	6.4	7.8	8.5			

C.4 SOME CHARACTERISTICS OF NON-PARAMETRIC METHODS

Any percentile estimation method will be subject to two sorts of uncertainty:

(a) bias - that is, a persistent tendency for estimates to err on one side or the other of the true value; and
(b) random error - that is, the haphazard scatter of estimates above and below their long-run average.

In general, all non-parametric methods have a poorer precision than the most appropriate parametric method: that is the price paid for the freedom from statistical assumptions about the underlying distribution. Where they mainly score over parametric methods is that their bias, though not necessarily small, is at least understood. When, in contrast, a parametric method is applied to data not in fact arising from the assumed distribution type, the bias will be unknown.

In the following discussion, we will assume for the sake of simplicity that the aim is to estimate high percentiles.

The Weibull method

The Weibull method is not unbiased; in fact, it suffers from two compounding elements of bias. The first is that the Weibull method is likely to produce rather more over-estimates than under-estimates. As Table C.2 indicates, the effect depends on the number of samples and the extremeness of the percentile being estimated. The most punishing combination is when the 95%ile is estimated from just 19 samples: in those circumstances, there is a 62% chance that the estimate will over-estimate the true 95%ile.

The second type of bias cannot be quantified without making some assumption about the shape of the underlying distribution - for example, that it is log-Normal. Depending on the degree of skewness, however, what commonly happens is that over-estimates tend to overstate the true concentration by more than the under-estimates understate it. This has the effect of introducing a positive bias.

Table C.2 - Probability that the Weibull method gives an over-estimate

No. of samples	50%ile	80%ile	95%ile
9	0.50	0.56	-
19	0.50	0.55	0.62
39	0.50	0.53	0.59
59	0.50	0.53	0.57

Note: the values in the table have been derived by binomial distribution theory.

There is, however, one important sense in which the Weibull method is actually unbiased. Corresponding to any given estimate of the P%ile will be some percentage figure measuring how much of the true underlying probability distribution falls below that estimate. This percentage figure will never be exactly equal to P: on some occasions it will be below P, on others above P. But what the Weibull method ensures is that, in the long run, these percentages will average out at P.

We can summarise this by saying that Weibull estimates are always unbiased in probability terms. Thus, for example, the true % compliance figures associated with a set of Weibull-estimated 95%ile-limits will in the long run average out at 95%. With so much percentile estimation being prompted by compliance issues, this statistical property of the Weibull method provides a reassuring guarantee.

Other methods

Many other non-parametric methods can be used, each with its own particular statistical pros and cons. One alternative that has a certain general appeal is the 'median' method. This offers the guarantee that, over repeated uses on different data sets (from whatever distribution), the method will tend to under-estimate the true percentile value exactly as often as it over-estimates it. The median method is discussed further in Appendix 5D of WRc's Sampling Handbook.

PRESENTING SUMMARY STATISTICS

PART A - RULES

This Code of Practice gives a set of rules relating to the layout and content of routine statistical summaries of data. The overall aim is to help the NRA to ensure that its data summaries are adequately informative whilst being as easy as possible to assimilate.

The rules are stated and illustrated here in Part A without elaboration. Part B then discusses the rules in more detail. Finally, Part C provides the statistical background needed in applying certain of the rules.

RULE 1: Need for consistency

Whenever statistical summaries are likely to be used to provide a national assessment, or for inter-regional comparisons, consistency between NRA regions in the derivation and layout of those summaries is of paramount importance.

RULE 2: Clarity of captions and headings

Captions on summary tables should not be compressed to such an extent as to obscure their meaning. If the available space is too small, remember that clarity can be achieved with the help of footnotes.

RULE 3: Confidence limits

Estimates of means, standard deviations and percentiles should wherever possible be accompanied by 90% confidence limits. Where only approximate confidence limits can be calculated, these are wholly acceptable as a substitute provided that the captions are amended accordingly.

RULE 4: The 'Standard' summary

As a minimum requirement, all summaries should state:

* the title of the data set;
* the date range spanned by the data;
* the selection criteria used for assembling the data (see CoP/DQC);

and then, for each determinand covered by the summary:

* determinand title and measurement units;
* total no. of data values;
* nos of less-than and greater-than values, and how these are dealt with;
* mean, with 90% confidence limits;
* standard deviation, with 90% confidence limits;
* the 95%ile (or 5%ile in the case of DO) with 90% confidence limits; and
* the minimum and maximum.

Table A.1 illustrates a possible layout for the Standard summary: the river quality data set used for the example is listed in Table C.1.

Table A.1 - Example of the 'Standard' summary

Site name: Avon at Cawling Farm, Willowdale
Date span of summary: 1/1/1988 to 31/12/1989
Selection criteria : Site Code: R02BF
: Purpose Codes: R (Routine); S (Supplementary)
: NGR: SD 547309

Determinand	BOD(ATU)	Amm.Nit.	DO (%)	Temperature
Units	mg/l	mg/l	% satn	deg C
No.of values	37	37	37	33
No.of <s	0	0	0	0
No.of >s	0	0	0	0
Mean	3.02	4.57	48.9	16.3
90% CI	2.74 - 3.29	3.80 - 5.34	41.3 - 56.5	14.7 - 17.9
St.dev.	1.00	2.76	27.45	5.42
90% CI	.84 - 1.24	2.32 - 3.44	23.07 -34.15	4.51 - 6.85
Minimum	1.50	1.02	8.0	2.0
95%ile♦	4.82	9.39	8.0♦	23.0
90% CI	4.4 - ???	7.7 - ???	??? - 10.0	22.0 - ???
Maximum	5.00	14.50	120.00	23.0

Note: 1. 5%ile rather than 95%ile is given for DO(%).♦
2. Percentiles are estimated by the Weibull non-parametric method.
3. '90% CI' denotes '90% confidence interval'. The CIs for standard deviations assume Normality and so are only approximate.

RULE 5: The 'Full' summary

Where a more comprehensive summary is required, the following statistics can be provided for each determinand in addition to those given in the Standard summary:

* the coefficient of variation, with 90% confidence limits;
* the ratio of 'successive differences deviation' (SDD) to conventional standard deviation, with 90% confidence limits;
* the percentage points estimated by the sample minimum and maximum; and
* the percentiles for the 1, 5, 10, 20, 50, 80, 90, 95 and 99 percentage points (or other percentiles as required by the application), each with 90% confidence limits where possible.

Table A.2 illustrates a possible layout for the Full summary. The same data set is used as that summarised more briefly in Table A.1.

RULE 6: Orientation of multi-determinand tables

In the (common) case in which the statistical summary covers more than one determinand, the information for any one determinand should preferably be arranged in a column rather than a row.

This rule is illustrated in Table A.1 and also (slightly adapted) in Table A.2.

RULE 7: Estimating percentiles

The statistical method used for estimating percentiles should always explicitly be stated. A non-parametric method is to be preferred - especially for general-purpose summaries - provided sufficiently many samples are available. Where a parametric method is used, the summary should indicate in a footnote the evidence for the assumed distributional model. More detailed guidance is available in Code of Practice CoP/PLE on 'Methods for estimating percentiles'.

RULE 8: Handling less-than and greater-than values

If the data set contains any 'censored' values, the methods given in Code of Practice CoP/LTV on 'Methods for handling less-than values and greater-than values' should be applied as appropriate.

Table A.2 - Example of the 'Full' summary

Site name: Avon at Cawling Farm, Willowdale
Date span of summary: 1/1/1988 to 31/12/1989
Selection criteria : Site Code: R02BF
: Purpose Codes: R (Routine); S (Supplementary)
: NGR: SD 547309

Determinand: Amm.Nit. Units: mg/l
No. of values: 37
No. of <'s : 0
No. of >'s : 0

Parameter	Estimate	90% conf.int.
Mean	4.568	3.802 - 5.335
St.dev.	2.763	2.321 - 3.437
Coeff.of Var:	.60	.49 - .72
SDD/SD ratio:	.92	.76 - 1.06
Minimum	1.02	(2.6 %ile)
Maximum	14.50	(97.4 %ile)

%ile	Estimate:	90% conf.int.
1	??? :	??? - 1.17
5	1.09 :	??? - 1.43
50	4.05 :	3.52 - 4.76
80	7.26 :	4.83 - 7.82
90	7.83 :	7.04 - 14.50
95	9.39 :	7.73 - ???
99	??? :	7.86 - ???

Determinand: DO (% sat) Units: % satn
No. of values: 37
No. of <'s : 0
No. of >'s : 0

Parameter	Estimate	90% conf.int.
Mean	48.89	41.27 - 56.51
St.dev.	27.45	23.07 - 34.15
Coeff.of Var:	.56	.45 - .67
SDD/SD ratio:	.55	.17 - .76
Minimum	8.0	(2.6 %ile)
Maximum	120.0	(97.4 %ile)

%ile	Estimate:	90% conf.int.
1	??? :	??? - 8.0
5	8.0 :	??? - 10.0
10	8.8 :	8.0 - 17.2
20	16.7 :	9.0 - 38.0
50	53.5 :	48.0 - 60.0
95	98.4 :	76.7 - ???
99	??? :	84.0 - ???

Determinand: Temperature Units: deg C
No. of values: 33
No. of <'s : 0
No. of >'s : 0

Parameter	Estimate	90% conf.int.
Mean	16.26	14.66 - 17.86
St.dev.	5.42	4.51 - 6.85
Coeff.of Var:	.33	.26 - .40
SDD/SD ratio:	.54	.04 - .76
Minimum	2.0	(2.9 %ile)
Maximum	23.0	(97.1 %ile)

%ile	Estimate:	90% conf.int.
1	??? :	??? - 5.0
5	4.1 :	??? - 9.2
50	18.0 :	16.0 - 19.0
80	21.0 :	19.0 - 22.0
90	22.0 :	21.0 - 23.0
95	23.0 :	22.0 - ???
99	??? :	23.0 - ???

Note: 1. 'SDD/SD ratio' denotes the ratio of the 'Successive Difference Deviation' to the conventional standard deviation. Ratios sig. less than 1.0 are a pointer to systematic time trends.
2. Percentiles are estimated by the Weibull non-parametric method.
3. Sample min. and max. values estimate the %iles shown in brackets.
4. '90% CI' denotes '90% confidence interval'. The CIs for standard deviations, coeffs of variation and SDD/SD assume Normality and so are only approximate.

PRESENTING SUMMARY STATISTICS

PART B - BACKGROUND

This Code of Practice aims to provide general guidance on the content of routine statistical summaries of data. It does not set out to cover all types of application. In particular, it is not intended to apply to one-off cases in which the user of the data is addressing a specific objective, and so will probably require a tailored, customized analysis of the data.

RULE 1: Need for consistency

Whenever statistical summaries are likely to be used to provide a national assessment, or for inter-regional comparisons, consistency between NRA regions in the derivation and layout of those summaries is of paramount importance.

Though the need for consistency is most evident where summaries are used at a national level, a consistent approach is also very desirable for summaries receiving only a local circulation. In any routine form of presentation, familiarity is a valuable aid. Thus, if a monthly summary report (say) always follows the same well-planned layout, this frees a part of the brain from needing to interpret the summary 'from cold' and so allows its content to be assimilated that much more readily.

RULE 2: Clarity of captions and headings

Captions on summary tables should not be compressed to such an extent as to obscure their meaning. If the available space is too small, remember that clarity can be achieved with the help of footnotes.

It is always a false economy to skimp on the words used in captions or column heads. Countless tables have been rendered unintelligible by the use of terse phrases such as " 95PC (+/-90%)". Thus extra time spent at the design stage on debating the precise wording of captions - and trying them out on colleagues - is very well worth while.

RULE 3: Confidence limits

Estimates of means, standard deviations and percentiles should wherever possible be accompanied by 90% confidence limits. Where only approximate confidence limits can be calculated, these are wholly acceptable as a substitute provided that the captions are amended accordingly.

Almost invariably, data sumaries are used to make statements about the quality of the entire body of water or effluent from which the samples were taken. For this reason it is important for the user to realise that summary statistics are not themselves the true values, but merely estimates of the underlying truth. The aim of Rule 3, accordingly, is to encourage an appreciation of the uncertainty inherent (to a greater or lesser degree) in all summary statistics.

Confidence limits give a useful quantitative measure of how far from the observed sample statistic the true value might lie. Consider, for example, the 90% confidence interval for mean DO % satn given in Table A.1, namely (41.3 to 56.5). This tells us that, although the observed mean DO was 48.9, the true mean could quite conceivably have been as low as 41.3 or as high as 56.5. Furthermore, even that fairly wide interval is not guaranteed to bracket the truth: we can be only '90% confident' that it does. A good way to appreciate what this means is to imagine a whole collection of 90% confidence intervals for mean DO, one for each of 100 different river sampling points. The guarantee then is that about 90 of those 100 confidence intervals (i.e. 90% in the long run) will successfully bracket the correct mean value.

What if a higher level of confidence is desired? This can readily be arranged - but only at the cost of widening the interval. In the example discussed above, the 90% confidence interval for DO was 'mean ± 7.6'. For 99% confidence, the interval must widen to 'mean ± 12.3'. For 99.9% confidence, it must widen further to 'mean ± 16.2'. In the last case, there is only a very slender risk - one in 1000 - that the true mean DO is not contained within the stated interval; but the interval is almost certainly too wide to provide much practical assurance.

There is nothing in principle to prevent the choice of confidence level from being varied according to the application. In practice, however, it is sensible to settle on a single fixed level wherever possible. This helps users to develop their appreciation of confidence intervals, and also enables the most to be made from between-summary comparisons.

For routine applications, a confidence coefficient of 90% - as proposed by Rule 3 - generally strikes a reasonable compromise between the conflicting goals of high confidence and narrow interval-width. Incidentally, a confidence coefficient of 95% - though widely used in statistics textbooks and elsewhere - is deliberately and strongly discouraged because of the confusion it would inevitably invite with 95%iles and 95% compliance.

Part C outlines the technical details of how confidence intervals are calculated, and provides references to sources giving a fuller discussion.

RULE 4: The 'Standard' summary (see Table A.1)

As a minimum requirement, all summaries should state:

- **the title of the data set;**
- **the date range spanned by the data;**
- **the selection criteria used for assembling the data (see CoP/DQC);**

and then, for each determinand covered by the summary:

- **determinand title and measurement units;**
- **total no. of data values;**
- **nos of less-than and greater-than values;**
- **mean, with 90% confidence limits;**
- **standard deviation, with 90% confidence limits;**
- **the 95%ile (or 5%ile in the case of DO) with 90% confidence limits; and**
- **the minimum and maximum.**

Table A.1 illustrates a possible layout for the Standard summary: the river quality data set used for the example is listed in Table C.1.

It is important that all summaries include explicit details of the criteria used in selecting the data from the quality archive, so that the user can check that the summary is appropriate for the required purpose. For example, 'special investigation' data should be excluded when producing information used for investigating long-term trends.

It will often be appropriate to screen data for outliers prior to - or as an integral part of - producing the summary statistics. This is another substantial topic in its own right, and is the subject of a separate Code of Practice CoP/OLR on 'Methods for handling outliers'.

Because of the variety of legislative instruments requiring 95%ile values to be quoted, this is one of the statistics recommended for inclusion in all Standard summaries. On purely statistical grounds, however, the 95%ile is not an especially useful or reliable summary measure - particularly with sample numbers of 40 or fewer. This, therefore, makes the supporting evidence provided by 90% confidence intervals all the more important.

RULE 5: The 'Full' summary (see Table A.2)

Where a more comprehensive summary is required, the following statistics can be provided for each determinand in addition to those given in the Standard summary:

- **the coefficient of variation, with 90% confidence limits;**
- **the ratio of 'successive differences deviation' (SDD) to conventional standard deviation, with 90% confidence limits;**
- **the percentage points estimated by the sample minimum and maximum; and**
- **the percentiles for the 1, 5, 10, 20, 50, 80, 90, 95 and 99 percentage points (or other percentiles as required by the application), each with 90% confidence limits where possible.**

Table A.2 illustrates a possible layout for the Full summary. The same data set is used as that summarised more briefly in Table A.I.

Layout

Apart from the greater variety of statistics that it contains, Table A.2 differs from Table A.1 most obviously in its 'blocked' layout, whereby all the information is presented one determinand at a time in a series of compact blocks or panels. After a great deal of experimentation, the Steering Group concluded that this was, on balance, the most satisfactory type of layout. There are two main points in its favour:

(i) It retains the spirit of the 'vertical' orientation recommended in Rule 6, and in doing so enables between-statistics comparisons to be made particularly easily.

(ii) The approach entirely bypasses the complications that arise with a conventional tabular layout whenever the captions need to be modified according to the determinand. For example, it is easy to:
- indicate that, for DO, 5%ile rather than 95%ile is quoted (see Table A.1); and
- implement the advice of CoP/LTV in cases when less-thans are present in some determinands but not in others (see Table B.2).

Additional statistics

The coefficient of variation (CoV) is a useful summary measure because it reflects the 'multiplicative' nature of variability often found in river and effluent quality, whereby the standard deviation tends to increase in proportion to the mean. This allows convenient rules of thumb such as: 'For effluent BOD and SS, the CoV is commonly around 0.4 - 0.5'.

The successive differences deviation (SDD) is a measure of the short-term variability in the data. If SDD is roughly the same size as the conventional standard deviation (SD), then the data consists largely of random scatter; but a value of SDD that is small in relation to SD is a good indication that the data contains some longer-term component of variation. Thus the SDD/SD ratio is a useful device - as the Table A.2 footnote indicates - for flagging up the presence of trends in the data. For some types of summary, indeed, it would be useful to go a stage further and alert the user to the presence of trend by highlighting all cases where the SDD/SD ratio is statistically significantly less than 1.0 - or, equivalently, the SDD/SD confidence interval lies entirely below 1.0. (This is the case with the approximate confidence intervals in Table A.2 for DO [0.17 - 0.76] and for temperature [0.04 - 0.76].) Such cases could then be investigated further, where appropriate, by time-series analysis.

The reason for recommending the inclusion of minimum and maximum percentage points is to encourage the user to appreciate that the sample minimum and maximum give little or no information about the position of the true, underlying minimum and maximum values (which are virtually certain to be much more extreme than the sample limits). The distinction is therefore an important one - and particularly so when considering, on the basis of past effluent quality data, what might constitute a realistic absolute limit.

The suggested selection of percentiles would primarily be of interest to more technical users of summaries. One of their functions, for example, would be to provide a comprehensive quantitative description of a histogram or probability plot of the data.

RULE 6: Orientation of multi-determinand tables

In the (common) case in which the statistical summary covers more than one determinand, the information for any one determinand should preferably be arranged in a column rather than a row.

This rule is illustrated in Table A.1 and also (slightly adapted) in Table A.2.

This rule represents a departure from the practice currently followed by some (though not all) NRA regions, and is a practical consequence of Rule 3 - that summary statistics should be accompanied by confidence limits. The main problem with a 'horizontal' layout is space. Even the Standard summary (see Table A.1) contains as many as 14 entries per determinand, and it would be very difficult to cram all these into a single row - even in 132-column, or landscape, mode. A columnar layout, on the other hand, offers ample 'vertical' scope for expansion.

Having each determinand's statistics arranged by column actually brings two further benefits. The first is that, as all the statistics for a particular determinand tend to be of the same order of magnitude and to require the same number of decimal places, they

will 'line up' neatly down the page. In contrast, an arrangement by row would produce a more jagged appearance because of changes in the scale of measurement (often very pronounced) from one determinand to another.

The second point is that between-determinand comparisons are required much less often than between-statistics comparisons for the same determinand (e.g. 95%ile v. mean; maximum v. minimum; SDD v. st.dev.). As it is far easier to scan down a column of figures than along a row, this is another reason for arranging tables so that the columns provide the principal dimension of comparison.

On their more detailed aspects, however, the Standard and Full summary layouts should be taken as guides rather than hard-and-fast rules. Indeed, it is actually an advantage to modify the content or style of the layout according to the particular application, as this provides further visual clues that speed the reader's recognition and hence appreciation of the summary.

RULE 7: Estimating percentiles

The statistical method used for estimating percentiles should always explicitly be stated. A non-parametric method is to be preferred - especially for general-purpose summaries - provided sufficiently many samples are available. Where a parametric method is used, the summary should indicate in a footnote the evidence for the assumed distributional model. More detailed guidance is available in Code of Practice CoP/PLE on 'Methods for estimating percentiles'.

Past experience has shown that percentile estimation has great potential for confusion and misunderstanding. This can introduce serious errors - especially when the number of data values is small. Rule 7 thus serves two purposes: first, to steer users towards the safer fall-back position of non-parametric methods; and secondly, to ensure that enough statistical information is given in the summary (whatever method is used) for users to gauge the soundness of any percentile values that are presented.

Percentile estimation is an extensive topic that is the subject of a separate Code of Practice, CoP/PLE. For convenience, however, the most commonly required expressions for estimating percentiles are reproduced in Part C of the present Code of Practice.

The performances of both parametric and non-parametric percentile methods are of particular importance in the context of river quality. To provide a ready means of evaluating the effect of sampling error on river Class, the PC program ARCTIC_SEAL (Assessing River Class Taking Into Consideration Sampling Error Against Limits) has been developed. This is available as the Test Data Facility procedure ARC.

RULE 8: Handling less-than and greater-than values

If the data set contains any 'censored' values, the methods given in Code of Practice CoP/LTV on 'Methods for handling less-than values and greater-than values' should be applied as appropriate.

Less-than or greater-than values in the data will introduce an additional degree of uncertainty in the summary. That is why it is important (a) for their presence to be flagged, and (b) for the relevant approach given in CoP LTV/1.3 to be applied.

Where less-thans are the problem, the recommended approach will usually involve calculating each required statistic twice - with less-thans being replaced first by their face value(s), and then by zero. This will produce two values bracketing the result that would have been obtained had actual measurements been available instead of less-thans.

Examples of how the Standard and Full summaries could be modified to cope with censored data are shown in Tables B.1 and B.2.

Table B.1 - Illustration of the Standard summary when the data contains less-than and greater-than values

Site name: COKETOWN STW
Date span of summary: 1/1/1983 to 31/12/1987
Selection criteria : Site Code: LDE012
: Purpose Codes: Q23/Q24/R01/R19

Determinand	S.Solids	BOD (5day)	Amm.Nit.
Units	mg/l	mg/l	mg/l
No.of values	258	257	258
No.of <'s	0	2	21
No.of >'s	0	6	0
Mean (<'s=0)	17.28	11.84	2.23
90% CI	16.16 - 18.40	11.24 - 12.44	2.03 - 2.43
Mean (<'s=L)		11.89	2.30
90% CI		11.29 - 12.49	2.10 - 2.49
St.dev(<'s=0)	10.90	5.84	1.93
90% CI	10.16 - 11.75	5.45 - 6.30	1.80 - 2.08
St.dev(<'s=L)		5.84	1.87
90% CI		5.45 - 6.30	1.74 - 2.01
Minimum	3.0 (.4%ile)	<1.5 (.4%ile)	<.10 (.4%ile)
95%ile	35.34	23.04	6.00
90% CI	31.0 - 51.4	20.8 - 25.8	5.4 - 7.6
Maximum	93.3 (99.6%ile)	>26.3 (99.6%ile)	11.00 (99.6%ile)

Note: 1. Percentiles are estimated by the Weibull non-parametric method.
2. 90% confidence intervals for standard deviations assume Normality, and so are approximate.

Table B.2 - Illustration of the Full summary when the data contains less-than and greater-than values

Site name: COKETOWN STW
Date span of summary: 1/1/1983 to 31/12/1987
Selection criteria : Site Code: LDE012
: Purpose Codes: Q23/Q24/R01/R19
: NGR: SD 547309

Determinand: BOD (5day) Units:
No. of values: 257
No. of <'s : 2
No. of >'s : 6

Parameter	Estimate	90% conf.int.
Mean (<'s=0):	11.84	11.24 - 12.44
Mean (<'s=L):	11.89	11.29 - 12.49
St.d.(<'s=0):	5.84	5.45 - 6.30
St.d.(<'s=L):	5.84	5.45 - 6.30
CoV (<'s=0):	.49	.46 - .53
CoV (<'s=L):	.49	.46 - .53
SDD/SD ratio:	.85	.79 - .91

%ile	Estimate:	90% conf.int.
5 (<'s=0):	4.3 :	3.6 - 5.4
5 (<'s=L):	4.2 :	3.2 - 5.3
50 :	10.6 :	10.1 - 11.2
80 :	16.7 :	15.2 - 19.1
90 :	20.5 :	20.0 - 23.0
95 :	25.1 :	22.6 - 36.8
99 :	>18.2 :	>10.4 - ???

Minimum :	<1.5 :	(.4 %ile)
Maximum :	>26.3 :	(99.6 %ile)

Determinand: Amm.Nit. Units:
No. of values: 258
No. of <'s : 21
No. of >'s : 0

Parameter	Estimate	90% conf.int.
Mean (<'s=0):	2.229	2.037 - 2.421
Mean (<'s=L):	2.296	2.104 - 2.488
St.d.(<'s=0):	1.867	1.741 - 2.013
St.d.(<'s=L):	1.865	1.740 - 2.012
CoV (<'s=0):	.84	.78 - .90
CoV (<'s=L):	.81	.75 - .87
SDD/SD ratio:	.81	.75 - .87

%ile	Estimate:	90% conf.int.
5 (<=0):	.00 :	.00 - .00
5 (<=L):	1.00 :	1.00 - .10
50 :	1.70 :	1.50 - 1.90
80 :	3.40 :	3.20 - 3.81
90 :	4.70 :	4.20 - 5.90
95 :	6.00 :	5.40 - 7.60
99 :	9.85 :	7.90 - ???

Minimum :	<.10 :	(.4 %ile)
Maximum :	11.00 :	(99.6 %ile)

Note: 1. 'SDD/SD ratio' denotes the ratio of the 'Successive Difference Deviation' to the conventional standard deviation. Ratios signif. less than 1.0 are a pointer to systematic time trends.
2. Percentiles are estimated by the Weibull non-parametric method.
3. Sample min. and max. values estimate the %iles shown in brackets.
4. '90% CI' denotes '90% confidence interval'. The CIs for standard deviations, coeffs of variation and SDD/SD assume Normality and so are only approximate.

PART C - TECHNICAL DETAILS

Table C.1 lists the data from which the summaries previously shown in Tables A.1 and A.2 were derived. To make it easier to follow through the derivation of the non-parametric percentile estimates and confidence limits, the data values are listed in increasing order.

Readers wishing to check their understanding of the statistical details in this Code of Practice - or to confirm that their existing summary routines produce the correct answer - may find it useful to rework some of the statistics using these test data sets.

Table C.1 - Listings of the data used for Tables A.1 and A.2

Determinand	Data values (ranked in increasing order)							
BOD(ATU)	1.50	1.50	1.70	1.70	1.70	1.90	1.90	2.00
	2.10	2.30	2.50	2.50	2.60	2.70	2.70	2.70
	2.80	2.80	2.80	2.90	2.90	3.20	3.40	3.40
	3.40	3.50	3.50	3.50	3.70	3.90	3.90	4.40
	4.40	4.70	4.70	4.80	5.00			
Amm.Nit.	1.02	1.10	1.17	1.41	1.43	1.90	2.03	2.23
	2.52	2.64	2.67	2.99	3.40	3.52	3.74	3.75
	3.90	4.05	4.05	4.21	4.41	4.53	4.63	4.76
	4.83	4.93	5.36	6.10	6.72	7.04	7.59	7.67
	7.73	7.82	7.86	8.82	14.50			
DO (%)	8.0	8.0	8.0	9.0	10.0	12.3	16.0	17.2
	18.0	28.0	28.4	32.1	38.0	48.0	48.6	49.7
	50.0	51.0	53.5	54.0	54.0	56.4	57.0	60.0
	62.0	62.0	64.0	66.0	67.0	70.0	73.0	73.0
	76.7	80.0	84.0	96.0	120.0			
Temperature	2.0	5.0	7.0	9.0	9.2	9.5	9.8	14.0
	14.0	15.0	16.0	16.0	16.0	16.0	17.0	17.0
	18.0	18.0	18.0	18.0	19.0	19.0	19.0	19.0
	21.0	21.0	21.0	21.0	22.0	22.0	22.0	23.0
	23.0							

For each of the parameters recommended in this Code of Practice, the following pages give expressions for calculating:

(i) the summary statistic that provides the best <u>estimate</u> of that parameter; and

(ii) a confidence interval (in some cases approximate) for the <u>true</u> value of the parameter estimated by that summary statistic.

Notation

The following terminology will be helpful:

n	no of determinand values
x(1), x(2), ..., x(n)	determinand values (in chronological order)
Σ[...]	denotes summation of the quantity inside [...], from i = 1 to n except where otherwise stated
$\bar{x}$	mean of the n values (defined below)
s	standard deviation of the n values (defined below)
√[...]	denotes the square root of the quantity inside [...]
t	0.95 quantile of t-statistic with the indicated degrees of freedom
Chi(05)	0.05 quantile of chi-squared statistic with the indicated degrees of freedom
Chi(95)	0.95 quantile of chi-squared statistic with the indicated degrees of freedom

Mean

Estimate $\bar{x} = \Sigma[x(i)]/n.$

Lower 90% conf. limit = $\bar{x} - ts/\sqrt{(n)}$, and

Upper 90% conf. limit = $\bar{x} + ts/\sqrt{(n)}$, where

s is calculated as below, and

degrees of freedom are n-1.

Standard deviation

Estimate $s = \sqrt{[\ \Sigma[(x(i) - \bar{x})^2]/(n-1)\ ]}.$

Lower 90% conf. limit ~ $s\sqrt{[\ (n-1)/Chi(95)\ ]}$, and

Upper 90% conf. limit ~ $s\sqrt{[\ (n-1)/Chi(05)\ ]}$.

see Note 2

Degrees of freedom are n-1.

Coefficient of variation

Estimate $CoV = s/(\bar{x})$.

Lower 90% conf. limit ~ $CoV[1 - t/\sqrt{(2n)}]$, and \
Upper 90% conf. limit ~ $CoV[1 + t/\sqrt{(2n)}]$. / see Note 2

Degrees of freedom are n-1.

SDD/SD ratio

Estimate $Rat = SDD/s$, where

$$SDD = \sqrt{\left[\sum_{i=2}^{n} [(x(i) - x(i-1))^2]/(2(n-1)) \right]}.$$

Lower 90% conf. limit ~ $\sqrt{[Rat^2 - Del]}$, and \
Upper 90% conf. limit ~ $\sqrt{[Rat^2 + Del]}$, / see Note 2

where $Del = t/\sqrt{[t^2 + n - 2]}$.

Degrees of freedom are n-2.

Percentiles

Suppose we wish to estimate the P-Percentile (or '%ile' for short). First, the x's should be sorted into increasing order. So x(1) now denotes the minimum, and x(n) the maximum. Next, calculate:

$p = P/100$,

$q = p(n+1)$,

r = integer part of q, and

$d = q - r$.

The P%ile can then be estimated by:

Estimate $P\%ile = [1-d].x(r) + d.x(r+1)$.

(Note: for certain combinations of P and n, no solution is possible.)

Now to calculate non-parametric confidence limits, we first need the binomial probabilities for the binomial distribution B(n,p). Suppose these are denoted by Pr(0), Pr(1),..., Pr(n). Each term can be calculated from:

$$Pr(r) = \frac{n!}{(n-r)!r!} p^r (1-p)^{(n-r)}, \quad r = 0,\ldots n.$$

Next, the two integers v_{max} and w_{min} are required:

(i) v_{max} is defined as the maximum value of v such that

$$\sum_{i=0}^{v} [\ Pr(i)\ ] \text{ is } \leq 0.05.$$

(For low percentiles, even a value of v as little as 0 may not satisfy this inequality.)

(ii) w_{min} is defined as the minimum value of w such that

$$\sum_{i=0}^{w} [\ Pr(i)\ ] \text{ is } \geq 0.95, \text{ and}$$

$$w_{min} < n.$$

(For high percentiles, it is not always possible to find a value of w less than n.)

The confidence limits can then be estimated (assuming both v_{max} and w_{min} can be obtained) by the following order statistics:

Lower 90% conf. limit = $x(v_{max}+1)$, and

Upper 90% conf. limit = $x(w_{min}+1)$.

General comments

1. The percentile estimates and associated confidence intervals are calculated by non-parametric methods - that is, they make no assumptions about the shape of the underlying statistical distribution. Details of parametric methods can be found in CoP/PLE - 'Methods for estimating percentiles'.

2. For the remaining parameters, the calculation of confidence intervals does make the assumption of underlying Normality. For the sample mean, this is usually a fairly safe assumption because of the Normalising effect of the Central Limit Theorem (but see the cautionary comments in CoP/EML on Estimating mean load). For the other three parameters, however - the standard deviation, the coefficient of variation, and the SDD/SD ratio - the calculations are more sensitive to departures from Normality. Moreover, the statistical theory becomes more complicated when the underlying population is non-Normal, and it is not possible to give useful general guidance. Instead, therefore, the proposal is to continue to use the Normal-based formulas but to ensure that the resulting confidence limits are labelled 'approximate'. This then ensures that the users are made aware of the uncertainty in the parameter in question; the fact that the correct confidence coefficient is not exactly 90% is of secondary concern.

3. A full account of the statistical methodology used here in Part C can be found in WRc's Sampling Handbook. Further practical guidance on non-parametric percentile issues can be obtained by using the ZEBRA package.

METHODS FOR ESTIMATING MEAN LOAD

This Code of Practice aims to ensure that the problems associated with load estimation are fully appreciated, and that estimation of the mean load of any substance transported by a river or effluent is handled in an appropriate manner by the NRA.

First, Part A sets out all the necessary algebraic definitions, and then presents the rules. These are then discussed in Part B. Finally, Part C gives the statistical details and also presents four illustrative examples.

PART A - DEFINITIONS AND RULES

Some algebraic notation is needed to define and discuss the various options for load estimation. To help make this as readable as possible, we have adopted the following convention. **Lower-case** symbols are used to denote all statistics calculated from a (limited) number of grab samples, whilst upper-case symbols are used for quantities calculated from a complete continuous record over the period of interest (and so regarded as being free from sampling error).

Definitions: grab-sample statistics

Instantaneous concentration..... c_i
The concentration of the substance of interest at any instant i.

Observed mean concentration..... $\bar{c}$
The arithmetic mean of the observed instantaneous concentrations..

Instantaneous flow..... q_i
The volume passing per unit time at any instant i.

Observed mean flow..... $\bar{q}$
The arithmetic mean of the observed instantaneous flows.

Instantaneous load..... l_i
The instantaneous rate of mass transfer of the substance of interest at any instant i - obtained by multiplying c_i and q_i.

Observed mean load..... $\bar{l}$
The arithmetic mean of the observed instantaneous loads.

Definitions: continuous-record statistics

Total duration of period..... T
T is measured in whatever time units are most appropriate for the scale of the particular application. For an overall period of one day, for example, T might have the value 24 (hours), 1440 (minutes) or 86400 (seconds).

True mean flow over a period..... $\bar{Q}$
The quantity V/T, where V is the total volume passing during the period.

True mean load over a period..... $\bar{L}$
The true average rate of mass transfer of the substance of interest - defined by the quantity M/T, where M is the total mass transferred over the period.

RULE 1: Using continuous monitoring data

Where it is important to have a reliable estimate of load, continuous monitoring data is essential. Rule 1a or 1b should be used according to whether the continuous monitoring is flow- or time-proportional. In either case, the resulting estimate can usually be assumed to have negligible statistical sampling error (although it will still be subject, of course, to instrument and analytical error).

RULE 1a: Flow-proportional continuous monitoring data

Where monitoring frequency is proportional to flow, and there is also a reliable record of mean flow over the whole period, mean load should be estimated by:

Method 2•: Mean load = $\bar{c}.\bar{Q}$

RULE 1b: Time-proportional continuous monitoring data

Where both concentration and flow data values are monitored at regular time intervals, mean load should be estimated by:

Method 3•: Mean load = $\bar{I}$

• The methods need to be numbered 2 and 3 (rather than 1 and 2) so that they match up with the relevant pair in the sequence of four methods defined in Rules 2a and 2b.

RULE 2: Using grab sampling data

Generally it will be necessary to estimate mean load from a limited number of grab samples. It is possible (though very unusual) for samples to be collected according to a flow-based regime - for example, a grab sample might be taken every 50Ml on average. In those circumstances Rule 2a should then be used. More usually, however, the grab sampling regime will be time-based - in which case Rule 2b should be used.

In either case it is important to appreciate that, because of the skewness so often characteristic of both flows and concentrations, the statistical sampling error can be very great. This issue is addressed by Rules 3, 4 and 5.

RULE 2a: Flow-based grab sampling data

Where flow-based grab sampling data provides the sole source of information, mean load should be estimated by:

Method 1: Mean load = $\bar{c}.\bar{q}$

Where, in addition, continuous flow records are available, an improved estimate is generally provided by:

Method 2: Mean load = $\bar{c}.\bar{Q}$

RULE 2b: Time-based grab sampling data

Where time-based grab sampling data provides the sole source of information, mean load should be estimated by:

Method 3: Mean load = $\bar{I}$

Where, in addition, continuous flow records are available, an improved estimate is generally provided by:

Method 4: Mean load = $\bar{I}.(\bar{Q}/\bar{q})$

RULE 3: Confidence limits

Because of the special difficulties arising in load estimation, it is particularly important always to give confidence limits showing the uncertainty surrounding any load estimate (see CoP/SSS on 'Presenting summary statistics'). In the absence of a clear-cut model for the shape of the instantaneous load probability distribution, confidence limits should be based on the log-Normal distribution using one of the methods described in Part C.

RULE 4: Representing high-flow events

Where high-flow events (whether in rivers or in effluents) are not adequately represented in the data, there is a risk of the mean load estimate being very severely biased. The bias may be either positive or negative depending on the shape and strength of the relationship between concentration and flow. Where this risk seriously jeopardises the objective of the exercise, it could be worth supplementing the routine sampling programme with a small number of special surveys.

RULE 5: Using additional information

Where a large proportion of the total load is transported in a relatively small proportion of the time, p, and the number of samples available is only of the order of 1/p, cases will commonly arise in which the data contains no extreme values. Not only will the resulting estimates be biased: the data will also produce misleadingly narrow confidence limits. Depending on the context, therefore, it can be prudent to use more realistic information from elsewhere on the likely underlying variability (e.g. from more comprehensive data on other similar rivers or discharges). Part C provides a table that would help in applying such an approach.

RULE 6: Exploiting relationships between concentration and flow

Any known or expected relationship between concentration and flow should be exploited wherever possible. For important applications, statistical modelling could be used to quantify the specific relationship for that site, leading to a more accurate estimate of load: this would be particularly useful where an extended flow record was available. More simply, even a rough understanding of the shape of the concentration-flow relationship will give a clear pointer as to which of Methods 1 to 4 is the most appropriate.

RULE 7: Alternative method for riverine load estimation

Mean load can almost always be estimated more easily and precisely for a discharge than for a river, as:

* polluting substances will generally be present at much higher concentrations and so present a less demanding analytical task; and
* the discharge will pass through a confined channel and so its flow is likely to be easier to measure.

In cases, therefore, where the substance of interest transported by a river derives only or mainly from a small number of point sources, it will often be better to derive an estimate of mean load indirectly from discharge data rather than directly by sampling the river.

PART B: BACKGROUND

Historically, the water industry has found load estimation a particularly troublesome task. The main difficulty has been that, unlike most other monitoring objectives, load estimation calls for both concentration and flow data. For rivers, other problems have been the considerable skewness exhibited by flow, and, for some determinands, inadequate limits of detection. Uncertainty over the most appropriate estimation formula to use has been a further complication.

It is particularly important, therefore, given the growing concern in recent years about the amounts of toxic substances and other materials discharged to coastal waters, that the NRA establishes sound methods of load estimation and ensures that these are applied consistently across the regions. With the increasing emphasis on load-based consents, moreover, there is also a particular need within the NRA for a better understanding of loads from effluent discharges.

RULE 1: Using continuous monitoring data

Where it is important to have a reliable estimate of load, continuous monitoring data is essential. Rule 1a or 1b should be used according to whether the continuous monitoring is flow- or time-proportional. In either case, the resulting estimate can usually be assumed to have negligible statistical sampling error (although it will still be subject, of course, to instrument and analytical error).

RULE 1a: Flow-proportional continuous monitoring data

Where monitoring frequency is proportional to flow, and there is also a reliable record of mean flow over the whole period, mean load should be estimated by:

Method 2: Mean load = $\bar{c}.\bar{Q}$

RULE 1b: Time-proportional continuous monitoring data

Where both concentration and flow data values are monitored at regular time intervals, mean load should be estimated by:

Method 3: Mean load = $\bar{I}$

Note that the grab-sample symbols $\bar{c}$ and $\bar{I}$ are used in Rules 1a and 1b (rather than $\bar{C}$ and $\bar{L}$) because the concentrations produced by continuous monitoring are still, technically, grab samples (albeit in very great quantity). In practice, however, they will usually be so close to the true values that sampling error can be ignored.

Some readers may find it surprising that there is not just one single formula for the continuous monitoring case. The reason why two formulas are needed is that the characteristics of the data do remain intrinsically different according to whether the values were obtained at equal volume increments (Rule 1a) or at equal time increments (Rule 1b). The essence of the difference is that with Rule 1a, the flow-weighting of concentration is performed automatically by the physical mechanism generating the samples, whilst in Rule 1b it has to be built in explicitly at the calculation stage.

RULE 2: Using grab sampling data

Commonly it will be necessary to estimate mean load from a limited number of grab samples. It is possible (though very unusual) for samples to be collected according to a flow-based regime - for example, a grab sample might be taken every 50Ml on average. In those circumstances Rule 2a should then be used. More usually, however, the grab sampling regime will be time-based - in which case Rule 2b should be used.

In either case it is important to appreciate that, because of the skewness so often characteristic of both flows and concentrations, the statistical sampling error can be very great. This issue is addressed by Rules 3, 4 and 5.

RULE 2a: Flow-based grab sampling data

Where flow-based grab sampling data provides the sole source of information, mean load should be estimated by:

Method 1: Mean load = $\bar{c}.\bar{q}$

Where, in addition, continuous flow records are available, an improved estimate is generally provided by:

Method 2: Mean load = $\bar{c}.\bar{Q}$

RULE 2b: Time-based grab sampling data

Where time-based grab sampling data provides the sole source of information, mean load should be estimated by:

Method 3: Mean load = $\bar{I}$

Where, in addition, continuous flow records are available, an improved estimate is generally provided by:

Method 4: Mean load = $\bar{I}.(\bar{Q}/\bar{q})$

Method 3 is the estimation formula used under ParCom for estimating loads of Red List substances.

Method 4 refers to the situation where the concentration data has been obtained from grab sampling but a continuous and complete record of flow happens to be available. The factor $\bar{Q}/\bar{q}$ in the Method 4 formula thus provides a correction to the Method 3 formula which will, in most circumstances, improve its precision.

Illustrative examples

Given the extreme skewness typically found in the distributions of flow and concentration, it is not obvious what the effects of sampling error on load estimation might be. For this reason we have produced the four illustrative examples described in detail in Part C. In each example we define a particular hypothetical pattern of flow and concentration variations in a river. We then show how each of the four estimation methods defined in Rules 2a and 2b would perform when applied to the sample values that would be generated by a programme of random time-based grab sampling from such a river.

The four hypothetical rivers have the following essential characteristics:

Example 1 - constant flow all year;
high concentration for 10% of year.

Example 2 - constant concentration all year;
high flow for 10% of year.

Example 3 - constant load all year;
high flow for 10% of the year.

Example 4 - high flow and high concentration
for 10% of year.

The discussion in Part C of the performance of Methods 1 to 4 reaches five main conclusions:

a. None of the four methods gives consistently good results.

b. Where flow is constant, all four methods give identical results for any given year's samples, but all are sensitive to the number of samples that happen to be taken at times of high concentration.

c. Where concentration is expected to remain roughly constant as flow varies, it is best to use Method 2 or Method 4.

d. Where load is expected to remain roughly constant as flow varies, it is best to use Method 3.

e. If the purpose is to estimate total load averaged over a number of years or across a number of rivers, Method 3 is the best as it is the only one of the four which consistently gives an unbiased result.

Practical studies

The examples in Part C, though very simple, are instructive in the way they echo conclusions reached from a number of extensive practical studies. Walling and Webb (1985) describe a very interesting simulation investigation of the magnitude of the errors involved when estimating mean load from a small number of samples. Their study was based on a two-year sequence of flow and suspended solids data obtained from a continuous monitoring station on the River Exe. They considered three common sampling frequencies - weekly, fortnightly and monthly - and four commonly-quoted load estimation methods (Methods 1 to 4 as defined in Rules 2a and 2b). For each sampling frequency in turn, they selected fifty different regular subsets of the full data set. They then evaluated the performance of each estimation method by applying it to the fifty subsets and calculating the variability and the mean bias of the resulting fifty estimates.

The Walling and Webb study found that Methods 1 and 2 gave fairly precise estimates - that is, the variation between estimates from replicate subsets was small. Unfortunately they were also badly biased, with load estimates showing a marked tendency to under-estimate the true mean load: for example, the results were on average only 38% of the true mean for weekly sampling and 25% for monthly. In contrast, Methods 3 and 4 showed little bias (as would be expected on theoretical grounds), but produced much greater dispersion. For example, some individual estimates were less than 10% or more than 300% of the true load.

The failure of methods based on grab sampling through time to account fully for the variations in concentrations and flow was confirmed in two more recent studies. WRc, in collaboration with Essex University, investigated heavy metals (Harrison et al. 1989) and pesticides (van Dijk et al, 1991) in the River Thames. Using the two methods of calculation based on instantaneous load values (Methods 3 and 4), estimates of total load were obtained and compared with the actual total load obtained by continuous monitoring. Many of the confidence intervals were so wide as to render the load estimates themselves of little practical use.

In conclusion, therefore, each of the available options for calculating load from a limited number of samples has potentially serious drawbacks. Methods 1 and 2 give repeatable results but suffer from an unknown and possibly large bias, whilst Methods 3 and 4 give unbiased results on average but can produce a large error in individual cases. Thus there is no single best method: the choice should take account of the objectives of the exercise, together with any available prior knowledge about the nature of flow and concentration variations in the sampled river or effluent.

On balance, however, we believe that relative freedom from bias is a more essential property than good repeatability, particularly for objectives concerned with aggregating results over years or across regions. As a general guide, therefore, Methods 3 and 4 are to be preferred - provided that due attention is paid to the various aspects of sampling error now addressed in Rules 3 to 5 following.

RULE 3: Confidence limits

Because of the special difficulties arising in load estimation, it is particularly important always to give confidence limits showing the uncertainty surrounding any load estimate (see CoP/SSS on 'Presenting summary statistics'). In the absence of a clear-cut model for the shape of the instantaneous load probability distribution, confidence limits should be based on the log-Normal distribution using one of the methods described in Part C.

For calculating confidence limits around means, the advice given in CoP/SSS is that Normality can generally be assumed because of the Central Limit Theorem. (This states, roughly, that the uncertainty in the mean of a set of sample values will be approximately Normal even when the individual values are far from being Normally distributed.)

For the special case of load estimation, however, this is an unreliable assumption as the distribution of instantaneous load can be so highly skewed. Thus, whilst mean load will certainly have a distribution nearer to Normal than that of the individual instantaneous loads, the approximation to Normality may not be very close - particularly if the number of samples is small.

For this reason we recommend as the default the approximate procedure described in Part C based on the log-Normal model.

RULE 4: Representing high-flow events

Where high-flow events (whether in rivers or in effluents) are not adequately represented in the data, there is a risk of the mean load estimate being very severely biased. The bias may be either positive or negative depending on the shape and strength of the relationship between concentration and flow. Where this risk seriously jeopardises the objective of the exercise, it could be worth supplementing the routine sampling programme with a small number of special surveys.

The problem of high-flow events is especially critical in rivers where the substance of interest is associated with particulates, since the transport of particulates is by nature episodic and dominated by flood events. In these circumstances, a large proportion of the total load may be discharged during relatively short periods - with the statistical consequences illustrated by Example 4 in Part C.

In the previously-mentioned study of the River Thames (van Dijk et al), for example, about 80% of the total observed annual load for a number of pesticides was discharged during a period of eight weeks of high flow in the winter. Moreover, no observations could be made during four weeks at the height of the Thames floods of 1990, and so it is likely that a still higher proportion of the total loads passed during this period.

The episodic nature of suspended sediment transport and the dominance of flood events in such transport was also clearly demonstrated by Walling and Webb (1985), who observed that 60% of the overall suspended sediment load was transported during only 2% of the time.

RULE 5: Using additional information

Where a large proportion of the total load is transported in a relatively small proportion of the time, p, and the number of samples available is only of the order of 1/p, cases will commonly arise in which the data contains no extreme values. Not only will the resulting estimates be biased: the data will also produce misleadingly narrow confidence limits. Depending on the context, therefore, it can be prudent to use more realistic information from elsewhere on the likely underlying variability (e.g. from more comprehensive data on other similar rivers or discharges). Part C provides a table that would help in applying such an approach.

Suppose, for example, that the major part of the load is transported by the high flows occurring during 8% of the year (so p = 0.08.) If the number of samples is around 1/p = 12 - i.e. monthly - it can be calculated that such high-flow events would be missed entirely more than one year in three.

In these circumstances, it is likely - particularly when the sampling dates are fixed in advance - that virtually all samples will be collected during periods of comparatively low flow. Very occasionally, samples will be collected during episodes of high flow and associated high sediment transport. Thus, for the many substances readily adsorbed by sediments, infrequent sampling is likely to lead to a gross under-estimation of total load, but the inclusion by chance of one or two samples with high concentrations will produce very different load estimates to those obtained without such values. Comparisons between different years or between rivers will thus be very dependent on the inclusion or exclusion by chance of one or two peak concentrations.

One practical way of lessening the risks of erroneous comparisons in such situations, therefore, is to 'borrow' more realistic confidence limits from other similar but more extensive data sets. Such an approach is likely to become more feasible as experience of load estimation develops, and a better understanding is obtained of the variability of estimates for different substances in given circumstances.

RULE 6: Exploiting relationships between concentration and flow

Any known or expected relationship between concentration and flow should be exploited wherever possible. For important applications, statistical modelling could be used to quantify the specific relationship for that site, leading to a more accurate estimate of load: this would be particularly useful where an extended flow record was available. More simply, even a rough understanding of the shape of the concentration-flow relationship will give a clear pointer as to which of Methods 1 to 4 is the most appropriate.

As with Rule 5, the aim here is to use whatever knowledge may be available to squeeze the most out of the data. A good example of a determinand showing a well-established relationship with flow is chloride; phosphate and sulphate are other likely candidates.

RULE 7: Alternative method for riverine load estimation

Mean load can almost always be estimated more easily and precisely for a discharge than for a river, as:

* **polluting substances will generally be present at much higher concentrations and so present a less demanding analytical task; and**
* **the discharge will pass through a confined channel and so its flow is likely to be easier to measure.**

In cases, therefore, where the substance of interest transported by a river derives only or mainly from a small number of point sources, it will often be better to derive an estimate of mean load indirectly from discharge data rather than directly by sampling the river.

This approach is in fact already used in some regions: for example, inputs to the North Sea are based on data from Harmonised Monitoring points and major discharges below them. It will become an increasingly feasible option as more dischargers install flow-measurement devices in response to the need to demonstrate compliance with load-based consents. It is likely to be useful, moreover, even when matters are complicated by the existence of other unmonitored or diffuse-source inputs. Given the wide confidence limits typically arising with river-based estimates of load, even a partial estimate of the contribution from discharges might well provide a tighter lower bound on the total load than that implied by riverine data alone.

One drawback with the indirect method of load estimation is that the calculation of confidence limits becomes too complicated for useful general guidance to be given in this Code of Practice. Anyone wishing to use this approach, therefore, is advised to seek the advice of a statistician.

REFERENCES

WALLING D E and WEBB B W (1985) Estimating the discharge of contaminants to coastal waters by rivers: some cautionary comments. Marine Pollution Bulletin, 16(12), 488-492.

HARRISON R M, THOROGOOD G A and LACEY R F (1990) Comparison of load estimation from grab sampls and continuous flow proportional sampling. Report PRS 2383-M, WRc, Medmenham.

VAN DIJK P A H, SAGE A and HARRISON R M (1991) Variability of pesticides in river water and its effects on estimation of load. Report NR2656, WRc, Medmenham.

PART C - TECHNICAL DETAILS

C.1 CONFIDENCE LIMITS

We start with the statistical details for Method 3 as this is the recommended approach likely to be used most often in practice.

Method 3

Suppose that M is the mean and S the standard deviation of the observed instantaneous loads data. The coefficient of variation, C, can hence be calculated by $C = S/M$. On the assumption that load is log-Normally distributed, the estimated standard deviation of log(load), s, can be determined from:

$$s^2 = \ln[1 + C^2].$$ (Note: 'ln' denotes 'log to base e'.)

Moreover, observed mean load $\bar{I}$ is related to m and s, the estimated mean and standard deviation of ln(load), by:

$$\bar{I} = \mathrm{EXP}[m + 0.5s^2].$$

Now the standard error of $[m + 0.5s^2]$ is approximately equal to:

$$E = \sqrt{[s^2/n + (0.25)2s^4/n]}.$$

An approximate 90% confidence interval for $\ln(\bar{I})$ is therefore given by:

$$\ln[\bar{I}] \pm tE$$

where t is the Student's t statistic for the appropriate degrees of freedom (namely n-1) and desired confidence level.

Approximate multiplicative limits around the estimate $\bar{I}$, therefore, are:

$$\mathrm{EXP}[\pm tE].$$

Example

Suppose the estimated coefficient of variation, C, is 1.2, and the estimate of mean load has been obtained from n = 24 samples.

Then $s^2 = \ln(1 + 1.44) = 0.892$

so $E = \sqrt{[.892/24 + 0.5(.892)(.892)/24]}$
$= \sqrt{[0.05374]}$
$= 0.232.$

With 24-1 = 23 degrees of freedom, the t value for 90% confidence is 1.71. Multiplicative 90% limits are therefore:

EXP[±1.71(0.232)]
= EXP[±0.397]

viz 0.67 and 1.49.
---- ----

These calculations have been repeated for a variety of typical coefficients of variation and numbers of samples; the results are presented in Table C.1.

Table C.1 - Multipliers giving approximate 90% confidence intervals for the mean of a log-Normally distributed determinand

No of samples		Observed coefficient of variation 0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8	2.0
4	lower:	0.63	0.50	0.40	0.32	0.26	0.22	0.18	0.16	0.13
	upper:	1.60	2.01	2.52	3.11	3.80	4.57	5.44	6.38	7.41
6	lower:	0.72	0.61	0.52	0.45	0.39	0.34	0.30	0.27	0.25
	upper:	1.39	1.63	1.91	2.22	2.55	2.91	3.28	3.67	4.08
12	lower:	0.81	0.73	0.66	0.61	0.55	0.51	0.47	0.44	0.41
	upper:	1.23	1.36	1.50	1.65	1.80	1.96	2.11	2.27	2.42
24	lower:	0.87	0.81	0.76	0.71	0.67	0.64	0.60	0.58	0.55
	upper:	1.15	1.23	1.32	1.40	1.49	1.57	1.66	1.74	1.82
30	lower:	0.88	0.83	0.78	0.74	0.70	0.67	0.64	0.61	0.59
	upper:	1.13	1.20	1.28	1.35	1.42	1.49	1.56	1.63	1.70
40	lower:	0.90	0.85	0.81	0.77	0.74	0.71	0.68	0.66	0.63
	upper:	1.11	1.17	1.23	1.29	1.35	1.41	1.47	1.52	1.58
50	lower:	0.91	0.87	0.83	0.79	0.76	0.74	0.71	0.69	0.67
	upper:	1.10	1.15	1.21	1.26	1.31	1.36	1.41	1.45	1.50
60	lower:	0.92	0.88	0.84	0.81	0.78	0.76	0.73	0.71	0.69
	upper:	1.09	1.14	1.18	1.23	1.28	1.32	1.36	1.41	1.44
80	lower:	0.93	0.89	0.86	0.83	0.81	0.79	0.76	0.74	0.73
	upper:	1.08	1.12	1.16	1.20	1.24	1.27	1.31	1.34	1.37
100	lower:	0.94	0.91	0.88	0.85	0.83	0.81	0.79	0.77	0.75
	upper:	1.07	1.10	1.14	1.17	1.21	1.24	1.27	1.30	1.33
150	lower:	0.95	0.92	0.90	0.88	0.86	0.84	0.82	0.81	0.79
	upper:	1.06	1.08	1.11	1.14	1.17	1.19	1.21	1.24	1.26
200	lower:	0.95	0.93	0.91	0.89	0.88	0.86	0.85	0.83	0.82
	upper:	1.05	1.07	1.10	1.12	1.14	1.16	1.18	1.20	1.22

Variant of Method 3 when C is known

The method described above can be modified for the case where the coefficient of variation is known (or can be assumed) rather than estimated from the data.

First, on the assumption that load is log-Normally distributed, calculate σ, the standard deviation of ln(load) implied by the given value of C. This is:

$$\sigma^2 = \ln[1 + C^2].$$ (Note: 'ln' denotes 'log to base e'.)

The standard error of $[m + 0.5\sigma^2]$ is simply:

$$E = \sqrt{[\sigma^2/n]}.$$

An approximate 90% confidence interval for $\ln(\bar{I})$ is therefore given by:

$$\ln[\bar{I}] \pm 1.65E$$

(As C is assumed to be known, we use the standard Normal variate for 90% confidence, 1.65, rather than Student's t.)

Approximate multiplicative limits around the estimate $\bar{I}$, therefore, are:

$$EXP[\pm 1.65E].$$

Example

Suppose the coefficient of variation, C, is reliably believed to be 1.2, and the estimate of mean load has been obtained from n = 24 samples.

Then $\sigma^2 = \ln(1 + 1.44) = 0.892$

so $E = \sqrt{[0.892/24]}$
$= 0.193.$

Thus approximate multiplicative limits around the estimate $\bar{I}$ are:

$$EXP[\pm 1.65(0.193)]$$
$$= EXP[\pm 0.318]$$

viz 0.73 and 1.37.

These calculations have been repeated for a variety of typical coefficients of variation and numbers of samples; the results are presented in Table C.2.

Table C.2 - Multipliers giving approximate 90% confidence intervals for the mean of a log-Normally distributed determinand when the variability is assumed known

No of samples		Assumed coefficient of variation 0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8	2.0
4	lower:	0.73	0.63	0.56	0.50	0.46	0.42	0.40	0.37	0.35
	upper:	1.37	1.58	1.78	1.98	2.17	2.36	2.53	2.69	2.84
6	lower:	0.77	0.69	0.62	0.57	0.53	0.50	0.47	0.45	0.43
	upper:	1.30	1.45	1.60	1.75	1.89	2.01	2.13	2.24	2.34
12	lower:	0.83	0.77	0.72	0.67	0.64	0.61	0.59	0.57	0.55
	upper:	1.20	1.30	1.40	1.48	1.57	1.64	1.71	1.77	1.83
24	lower:	0.88	0.83	0.79	0.76	0.73	0.70	0.68	0.67	0.65
	upper:	1.14	1.20	1.27	1.32	1.37	1.42	1.46	1.50	1.53
30	lower:	0.89	0.85	0.81	0.78	0.75	0.73	0.71	0.70	0.68
	upper:	1.12	1.18	1.24	1.28	1.33	1.37	1.40	1.43	1.46
40	lower:	0.90	0.87	0.83	0.81	0.78	0.76	0.75	0.73	0.72
	upper:	1.11	1.16	1.20	1.24	1.28	1.31	1.34	1.37	1.39
50	lower:	0.91	0.88	0.85	0.82	0.80	0.78	0.77	0.76	0.74
	upper:	1.09	1.14	1.18	1.21	1.25	1.27	1.30	1.32	1.34
60	lower:	0.92	0.89	0.86	0.84	0.82	0.80	0.79	0.77	0.76
	upper:	1.09	1.12	1.16	1.19	1.22	1.25	1.27	1.29	1.31
80	lower:	0.93	0.90	0.88	0.86	0.84	0.83	0.81	0.80	0.79
	upper:	1.07	1.11	1.14	1.17	1.19	1.21	1.23	1.25	1.26
100	lower:	0.94	0.91	0.89	0.87	0.86	0.84	0.83	0.82	0.81
	upper:	1.07	1.10	1.12	1.15	1.17	1.19	1.20	1.22	1.23
150	lower:	0.95	0.93	0.91	0.89	0.88	0.87	0.86	0.85	0.84
	upper:	1.05	1.08	1.10	1.12	1.14	1.15	1.16	1.18	1.19
200	lower:	0.96	0.94	0.92	0.91	0.90	0.89	0.88	0.87	0.86
	upper:	1.05	1.07	1.09	1.10	1.12	1.13	1.14	1.15	1.16

Note: the approximate multipliers in this table are calculated on the assumption that C, the coefficient of variation, is known (or can be assumed). If instead C is estimated from the data, Table C.1 should be used.

Method 4

The statistical behaviour of mean load estimates obtained by Method 4 is much more complicated than for Method 3, as the estimate now involves the ratio of two correlated quantities, $\bar{l}$ and $\bar{q}$. A detailed treatment therefore goes beyond the scope of this Code of Practice. For further discussion of the calculation of confidence limits for Method 4, the reader is referred to Van Dijk et al (1991). For detailed advice on how to proceed in a particular case, the advice of a statistician should be sought.

Methods 1 and 2

Similar remarks apply to estimates obtained by Method 1 - the statistical details for which are again complicated by involving a combination of correlated quantities. For Method 2, however, the procedure described in detail for Method 3 can be used, as it applies quite generally to any log-Normally distributed determinand. The only other point to note when applying the method to $\bar{c}.\bar{Q}$ rather than $\bar{l}$ is that $\bar{Q}$ should be regarded as a multiplicative constant.

C.2 WORKED EXAMPLES

This section presents the details of four worked examples. These have four main aims:

(i) to illuminate the arithmetic 'nuts and bolts' of each estimation formula, and so give the reader the opportunity to work through and check his or her understanding of the practical details (important if that person is subsequently responsible for specifying the details of a load estimation calculation routine);

(ii) to demonstrate how, even for one given set of sample values, different methods can produce very different load estimates;

(iii) to demonstrate, for any one estimation method, the extent to which the answer can vary from year to year; and

(iv) to demonstrate how the findings in (ii) and (iii) can themselves be very dependent on the nature of variations in flow and concentration in the river or effluent being sampled.

The four hypothetical rivers

The hypothetical rivers represented by the four examples have been designed to explore the consequences of four markedly different combinations of flow and/or concentration variability. To aid comparisons, the true mean load is arranged to have the same value (9.0 units) in each case.

Example 1: **flow stays constant** throughout the year; concentration is high for 10% of the year.

Example 2: **concentration stays constant** throughout the year; flow is high for 10% of the year.

Example 3: load stays constant throughout the year;
flow is high for 10% of the year.

Example 4: flow and concentration are correlated,
both being high for the same 10% of the year.

The load estimation methods

Four methods for estimating mean loads are examined. These are the four methods defined during the course of Rules 1 and 2. (They are also the principal four options studied by Walling and Webb.)

Method 1: Mean load = $\bar{c}.\bar{q}$

Method 2: Mean load = $\bar{c}.\bar{Q}$

Method 3: Mean load = $\bar{I}$

Method 4: Mean load = $\bar{I}.(\bar{Q}/\bar{q})$

Results

The four examples are presented in boxes on pages 18 to 23. We recommend that the reader works through Example 1 first; then reads the conclusions drawn about that example below; and then works similarly through Examples 2, 3 and 4.

Conclusions

Example 1: flow stays constant throughout the year;
concentration is high for 10% of the year.

For this river the four methods perform identically to one another, and all four produce results that average out, in the long run, at the true mean load. Individual results, however, are very sensitive to the data outcome happening to arise in any one year. For each method, the estimated mean load can range from 3.0 (in years when no high concentrations are selected) to 33.0 (in the rare cases when as many as three of the six samples are taken during the high concentration period).

Example 2: concentration stays constant throughout the year;
flow is high for 10% of the year.

Methods 2 and 4 give the true result for all data outcomes, whereas Methods 1 and 3 are highly susceptible to the number of samples happening to be taken during high flow periods. But in the long term, all four methods again average out at the true value.

Example 3: load stays constant throughout the year; flow is high for 10% of the year.

Method 3 gives the true result whatever combination of data happens to arise - and so of course is unbiased. The other three methods, however, are sensitive to the number of samples taken at high flow, and also have a positive bias - that is, they tend to over-estimate the true mean load.

Example 4: flow and concentration are correlated, both being high for the same 10% of the year.

For this river, Method 3 performs worst in one respect and best in another. It produces the estimates that vary the most wildly according to the number of high-load events that happen to be sampled; and yet it is the one method which in the long run generates estimates that average out to the correct mean load. In contrast, Method 2 produces the least variable results over the four data outcomes, but is badly biased, with an expected value of 5.76 rather than 9.0 - 36% too low.

Example 1: Constant flow

Suppose a river has constant flow but two levels of concentration:
Flow = 3 Conc = 1 (so Load = 3) for 90% of the year
Flow = 3 Conc = 21 (so Load = 63) for 10% of the year
Then
True mean flow $\bar{Q} = 0.90(3.00) + 0.10(3.00) = 2.7 + 0.3 = 3.00$
and
True mean load $\bar{L} = 0.90(3.00) + 0.10(63.00) = 2.7 + 6.3 = \boxed{9.00}$

Now suppose six samples are taken at random through the year. As high concentrations occur for only 10% of the time, it is quite likely that none of the six samples happens to catch a high concentration. In fact, the chance of this happening is just over 50% (the exact probability is $0.9^6 = 53.1\%$); so we have called this 'Outcome 1'. The next most likely situation (with prob. = 35.4%) is that just one sample produces a high concentration; this is called Outcome 2.

Outcomes 3 and 4 then describe the decreasingly likely cases in which two, then three of the six samples pick up high concentrations. Thus Outcome 3 would occur about one year in ten, whilst Outcome 4 would occur only about one year in 70.

The probabilities of Outcomes 1 to 4 add up to 99.8%. Thus, whilst in principle there might be four, five or even six high concentrations in the six samples, the chance of any of these still more extreme outcomes actually occurring is so slight (0.2%) that it can be ignored for the purposes of the example.

(continued on next page)

Outcome 1..... (Prob: 53.1%)			Outcome 2..... (Prob: 35.4%)			Outcome 3..... (Prob: 9.85%)			Outcome 4..... (Prob: 1.45%)		
flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l
3	1	3	3	21	63	3	21	63	3	21	63
3	1	3	3	1	3	3	21	63	3	21	63
3	1	3	3	1	3	3	1	3	3	21	63
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3

Now we can calculate the annual mean for each det'd for each Outcome...

$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$
3.00	1.00	3.00	3.00	4.33	13.0	3.00	7.67	23.0	3.00	11.0	33.0

... and so produce the estimates of mean load by Methods 1 to 4:

1: $\bar{c}.\bar{q}$	3.00	13.00	23.00	33.00
2: $\bar{c}.\bar{Q}$	3.00	13.00	23.00	33.00
3: $\bar{l}$	3.00	13.00	23.00	33.00
4: $\bar{l}(\bar{Q}/\bar{q})$	3.00	13.00	23.00	33.00

Finally, we can average 'horizontally' across the four Outcomes, weighting each Outcome according to its occurrence probability. This shows us, for each load estimation method, what the long-run average would be if the method were used repeatedly on this hypothetical river.

	Method	Weighted sum					Expected value of mean load
1:	$\bar{c}.\bar{q}$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)					
		= 1.59	+ 4.61	+ 2.27	+ 0.48	=	9.00
2:	$\bar{c}.\bar{Q}$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)					
		= 1.59	+ 4.61	+ 2.27	+ 0.48	=	9.00
3:	$\bar{l}$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)					
		= 1.59	+ 4.61	+ 2.27	+ 0.48	=	9.00
4:	$\bar{l}(\bar{Q}/\bar{q})$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)					
		= 1.59	+ 4.61	+ 2.27	+ 0.48	=	9.00

Example 2: Constant concentration

Suppose a river has constant concentration but two levels of flow:
Flow = 1 Conc = 3 (so Load = 3) for 90% of the year
Flow = 21 Conc = 3 (so Load = 63) for 10% of the year
Then
True mean flow $\bar{Q}$ = 0.90(1.00) + 0.10(21.00) = 0.9 + 2.1 = 3.00
and
True mean load $\bar{L}$ = 0.90(3.00) + 0.10(63.00) = 2.7 + 6.3 = **9.00**

Now suppose six samples are taken at random through the year. By the same argument as in Example 1, there is a chance of just over 50% that all six samples fall in the 'normal' period of the year: this constitutes Outcome 1. Outcomes 2, 3 and 4 then refer to the cases in which one, two and three of the six samples pick up high flows.

Outcome 1..... (Prob: 53.1%)			Outcome 2..... (Prob: 35.4%)			Outcome 3..... (Prob: 9.85%)			Outcome 4..... (Prob: 1.45%)		
flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l
1	3	3	21	3	63	21	3	63	21	3	63
1	3	3	1	3	3	21	3	63	21	3	63
1	3	3	1	3	3	1	3	3	21	3	63
1	3	3	1	3	3	1	3	3	1	3	3
1	3	3	1	3	3	1	3	3	1	3	3
1	3	3	1	3	3	1	3	3	1	3	3

Now we can calculate the annual mean for each det'd for each Outcome...

$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$
1.00	3.00	3.00	4.33	3.00	13.0	7.67	3.00	23.0	11.0	3.00	33.0

... and so produce the estimates of mean load by Methods 1 to 4:

1: $\bar{c}.\bar{q}$	3.00	13.00	23.00	33.00
2: $\bar{c}.\bar{Q}$	9.00	9.00	9.00	9.00
3: $\bar{I}$	3.00	13.00	23.00	33.00
4: $\bar{I}(\bar{Q}/\bar{q})$	9.00	9.00	9.00	9.00

Finally, we can calc. a weighted average across the four Outcomes...

Method	Weighted sum				Expected value of mean load
1: $\bar{c}.\bar{q}$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)				
	= 1.59	+ 4.61	+ 2.27	+ 0.48	= 9.00
2: $\bar{c}.\bar{Q}$	0.531(9) + 0.354(9) + 0.0985(9) + 0.0145(9)				
	= 4.78	+ 3.19	+ 0.89	+ 0.13	= 9.00
3: $\bar{I}$	0.531(3) + 0.354(13) + 0.0985(23) + 0.0145(33)				
	= 1.59	+ 4.61	+ 2.27	+ 0.48	= 9.00
4: $\bar{I}(\bar{Q}/\bar{q})$	0.531(9) + 0.354(9) + 0.0985(9) + 0.0145(9)				
	= 4.78	+ 3.19	+ 0.89	+ 0.13	= 9.00

Example 3: Constant load

Suppose a river has constant load but two levels of flow:
Flow = 3 Conc = 3 (so Load = 9) for 90% of the year
Flow = 9 Conc = 1 (so Load = 9) for 10% of the year
Then
True mean flow $\bar{Q}$ = 0.90(3.00) + 0.10(9.00) = 2.7 + 0.9 = 3.60
and
True mean load $\bar{L}$ = 0.90(9.00) + 0.10(9.00) = 8.1 + 0.9 = 9.00

Now suppose six samples are taken at random through the year. By the same argument as in Examples 1 and 2, the respective chances of the six samples picking up zero, one, two or three high flows are as given below under Outcomes 1, 2, 3 and 4.

Outcome 1..... (Prob: 53.1%)			Outcome 2..... (Prob: 35.4%)			Outcome 3..... (Prob: 9.85%)			Outcome 4..... (Prob: 1.45%)		
flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l
3	3	9	9	1	9	9	1	9	9	1	9
3	3	9	3	3	9	9	1	9	9	1	9
3	3	9	3	3	9	3	3	9	9	1	9
3	3	9	3	3	9	3	3	9	3	3	9
3	3	9	3	3	9	3	3	9	3	3	9
3	3	9	3	3	9	3	3	9	3	3	9

Now we can calculate the annual mean for each det'd for each Outcome...

$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$	$\bar{q}$	$\bar{c}$	$\bar{l}$
3.00	3.00	9.00	4.00	2.67	9.00	5.00	2.33	9.00	6.00	2.00	9.00

... and so produce the estimates of mean load by Methods 1 to 4:

1: $\bar{c}.\bar{q}$	9.00	10.67	11.67	12.00
2: $\bar{c}.\bar{Q}$	10.80	9.60	8.40	7.20
3: $\bar{l}$	9.00	9.00	9.00	9.00
4: $\bar{l}(\bar{Q}/\bar{q})$	10.80	8.10	6.48	5.40

Finally, we can average 'horizontally' across the four Outcomes, weighting each Outcome according to its occurrence probability...

Method	Weighted sum	Expected value of mean load
1: $\bar{c}.\bar{q}$	0.531(9) + 0.354(10.67) + 0.0985(11.67) + 0.0145(12) = 4.78 + 3.78 + 1.15 + 0.17 =	9.90
2: $\bar{c}.\bar{Q}$	0.531(10.8) + 0.354(9.6) + 0.0985(8.4) + 0.0145(7.2) = 5.74 + 3.40 + 0.83 + 0.10 =	10.08
3: $\bar{l}$	0.531(9) + 0.354(9) + 0.0985(9) + 0.0145(9) = 4.78 + 3.19 + 0.89 + 0.13 =	9.00
4: $\bar{l}(\bar{Q}/\bar{q})$	0.531(10.8) + 0.354(8.1) + 0.0985(6.48) + 0.0145(5.4) = 5.74 + 2.87 + 0.64 + 0.08 =	9.33

Example 4: High flow associated with high concentration

Suppose a river has high concentrations associated with high flows:
Flow = 3 Conc = 1 (so Load = 3) for 90% of the year
Flow = 21 Conc = 3 (so Load = 63) for 10% of the year
Then
True mean flow $\overline{Q}$ = 0.90(3.00) + 0.10(21.00) = 2.7 + 2.1 = 4.80
and
True mean load $\overline{L}$ = 0.90(3.00) + 0.10(63.00) = 2.7 + 6.3 = **9.00**

Now suppose ten samples are taken at random through the year. Here there are more possible outcomes than in Examples 1, 2 and 3, and so the probabilities are spread more diffusely between them. In particular, Outcomes 1 and 2 (picking up zero and one high flows respectively) can each be expected to arise about one year in three.

Outcome 1..... (Prob: 34.9%)			Outcome 2..... (Prob: 38.7%)			Outcome 3..... (Prob: 19.4%)			Outcome 4..... (Prob: 5.74%)		
flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l	flow q	conc c	load l
3	1	3	21	3	63	21	3	63	21	3	63
3	1	3	3	1	3	21	3	63	21	3	63
3	1	3	3	1	3	3	1	3	21	3	63
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3
3	1	3	3	1	3	3	1	3	3	1	3

Now we can calculate the annual mean for each det'd for each Outcome...

$\overline{q}$	$\overline{c}$	$\overline{l}$	$\overline{q}$	$\overline{c}$	$\overline{l}$	$\overline{q}$	$\overline{c}$	$\overline{l}$	$\overline{q}$	$\overline{c}$	$\overline{l}$
3.00	1.00	3.00	4.80	1.20	9.00	6.60	1.40	15.0	8.40	1.60	21.0

... and so produce the estimates of mean load by Methods 1 to 4:

1: $\overline{c}.\overline{q}$	3.00	5.76	9.24	13.44
2: $\overline{c}.\overline{Q}$	4.80	5.76	6.72	7.68
3: $\overline{l}$	3.00	9.00	15.00	21.00
4: $\overline{l}(\overline{Q}/\overline{q})$	4.80	9.00	10.91	12.00

(continued on next page)

Finally, we can average 'horizontally' across the four Outcomes, weighting each Outcome according to its occurrence probability...

	Method	**Weighted sum**	**Expected value of mean load**
1:	$\bar{c}.\bar{q}$	0.349(3) + 0.387(5.76) + 0.194(9.24) + 0.0574(13.44) = 1.05 + 2.23 + 1.79 + 0.77 =	6.08
2:	$\bar{c}.\bar{Q}$	0.349(4.8) + 0.387(5.76) + 0.194(6.72) + 0.0574(7.68) = 1.67 + 2.23 + 1.30 + 0.44 =	5.76
3:	$\bar{I}$	0.349(3) + 0.387(9) + 0.194(15) + 0.0574(21) = 1.05 + 3.49 + 2.91 + 1.21 =	9.00
4:	$\bar{I}(\bar{Q}/\bar{q})$	0.349(4.8) + 0.387(9) + 0.194(10.91) + 0.0574(12) = 1.67 + 3.49 + 2.11 + 0.69 =	8.13

USING THE PROTOTYPE TEST DATA FACILITY (TDF)

PART A - STEPS TO BE FOLLOWED

The Test Data Facility (TDF) is a standard operating protocol with supporting computer software that enables an NRA region to evaluate any proposed new method of data analysis quickly and easily using its own selected data sets. This Code of Practice provides guidance on the use of the prototype TDF system that has been produced. A companion Code of Practice (CoP/TDFd) deals with the development of procedures for inclusion in the TDF.

In designing and constructing the TDF, the main aim has been to ensure the greatest possible ease of use. Only usage will show the extent to which this has been achieved. Accordingly, we do ask every region to try out the TDF and report on any problems encountered, as feedback based on practical experience is the surest way in which any weak links can be identified and strengthened.

We define the following terms for use in this Code of Practice:

a. **TDF site:** any regional office or laboratory at which a TDF has been established.

b. **TDF PC:** any microcomputer on which the TDF system has been set up. This should be a PC-compatible machine, and have a maths co-processor. In view of the potentially heavy workload to which the TDF could be subjected, a 386 machine would be desirable. As the TDF occupies well under one Mbyte on the hard disk, there is no requirement for a dedicated PC; any suitable existing machine offering an adequate amount of hands-on time could be used.

c. **TDF manager:** the person in charge of setting up and maintaining the TDF at a particular TDF site.

d. **TDF user:** any other individual authorised to use the TDF.

The steps for using the TDF are stated here in Part A without elaboration. Part B then discusses and illustrates the steps in more detail.

STEP 1: Protocol for assembling the data

A protocol should be established at the TDF site for:

+ extracting any desired collection of data files from the quality archive;
+ downloading these to the TDF PC as 'AARDVARK-type' files; and
+ creating the corresponding set of '.CTL' control files.

STEP 2: Setting up the TDF on a PC

On the nominated PC, the TDF manager should create the subdirectory C:\TDF, and then, from the TDF diskette supplied, copy into C:\TDF the following files:

+ TDFInit.EXE;
+ TDFData.EXE;
+ Now1.DAT; and
+ TDF.BAT.

STEP 3: Adding a new procedure to the TDF

Any new procedure (QQQ, say) will be provided on a diskette containing two files - QQQRead.Me, and QQQ.EXE. The TDF manager should copy QQQ.EXE to C:\TDF, and then consult QQQRead.Me for details of the procedure and any special operating requirements.

STEP 4: Assembling the data for a TDF run

Using the protocol outlined in Step 1, the TDF user should extract from the quality archive the required test data files; download these to the TDF PC; and set up their AARDVARK-type control files.

STEP 5: Interactive or batch run

The TDF user should next decide whether the TDF run is to be a batch or an interactive run. When a new procedure is being tested, it is generally prudent to begin with the interactive option and apply QQQ to just a few data files, to check that the instructions in QQQRead.Me have been understood and followed correctly. For interactive runs, the user should proceed directly to Step 7; for batch runs, Steps 6a and 6b should first be followed.

STEP 6a: Naming the batch input file for a TDF batch run

The 'batch input' file holds details of the files to be processed in the current batch run. It is sensible for the TDF user to make the file name reflect both the procedure name (QQQ) and the run number (ij) of that particular application of the procedure. Thus, a suitable name might for example be 'ARC07.DAT'.

STEP 6b: Creating the batch input file for a TDF batch run

Next, the TDF user should create the batch input file according to the instructions given and illustrated in Table A.1.

Table A.1 - Contents of the batch input file

The batch input file contains two lines for each data set to be submitted to QQQ:

Line 1 contains the title of the data file (including the '.DAT'); and

Line 2 contains:
* the no. of determinands (detds) required;
* their positions in the file specified on the preceding line; and
* whether or not each detd is to be logged (0=no, 1=yes).

Illustration:

```
COP\R02B001.DAT
  7    1 2  3  4  5  7 11     0 0 0 1 1 1 1
D:\TEMMS\WICK.DAT
  4    4 2 3 6    0 1 1 1
```

Thus the instructions given by this file to the TDF are:

+ First, read file R02B001.DAT from \COP on the current drive; pick off seven detds, namely the first five plus detds 7 and 11; and log the last four of these.
+ Then read file WICK.DAT from \TEMMS on drive D; pick off detds 4, 2, 3 and 6 (in that order); and log all but the first detd.

STEP 7: Running a TDF application

To run any TDF application of procedure QQQ, the TDF user should type the command

TDF QQQ (press ENTER),

and then respond appropriately to the prompts from the PC screen.

STEP 8: Handling the output from a TDF run

From a batch run, the output is sent to a file in the C:\TDF directory having the same root name as the batch input file. For interactive runs, the output is simply named TDF.OUT. In either case, if the output is required as a permanent record, it is advisable before any subsequent TDF run for the user to:

+ print the file;
+ rename it; or
+ transfer it to some more appropriate storage location.

USING THE PROTOTYPE TEST DATA FACILITY (TDF)

PART B - BACKGROUND

This Code of Practice provides guidance on the use of the prototype Test Data Facility (TDF) system. As Part A describes, the TDF is a standard operating protocol with supporting computer software that enables an NRA region to evaluate any proposed new method of data analysis quickly and easily using its own selected data sets.

The terms 'TDF site', 'TDF PC', 'TDF manager', and 'TDF user' are used in the following discussion. These are more or less self-explanatory, but the reader may wish to consult the formal definitions given at the start of Part A.

STEP 1: Protocol for assembling the data

A protocol should be established at the TDF site for:

- **extracting any desired collection of data files from the quality archive;**
- **downloading these to the TDF PC as 'AARDVARK-type' files; and**
- **creating the corresponding set of '.CTL' control files.**

The TDF cannot function without data, and so the requirements set out in Step 1 are vital. Potentially, there are many useful ways in which data can be stored electronically, extracted, transferred and submitted to other software packages. This topic, indeed, is planned to be addressed by a future Code of Practice. In the meantime, the present 'low-tech' proposal has been made in the knowledge that all NRA regions are users of WRc's AARDVARK package and so will already have developed some form of procedure for getting data into the (fairly flexible) format required by AARDVARK.

For the convenience of those who do not have direct experience of AARDVARK, that section of the AARDVARK User Guide dealing with the required formats for data and control files has been reproduced in Part C of this Code of Practice.

STEP 2: Setting up the TDF on a PC

On the nominated PC, the TDF manager should create the subdirectory C:\TDF, and then, from the TDF diskette supplied, copy into C:\TDF the following files:

- **TDFInit.EXE;**
- **TDFData.EXE;**
- **Now1.DAT; and**
- **TDF.BAT.**

Once a system for assembling data has been established, setting up the TDF itself is very straightforward - as Step 2 indicates. Just four files need to be transferred to any convenient PC (which should have a maths co-processor). These are:

+ TDFInit.EXE - the executable code for setting up the details of the current TDF run;
+ TDFData.EXE - the executable code for inputting any particular data set to the TDF;
+ Now1.DAT - a small housekeeping file; and
+ TDF.BAT - a file of DOS batch commands.

The TDF.BAT file is a key component of the TDF, as it provides the control-loop mechanism by which one data set at a time is read into the TDF and passed across to the desired procedure for analysis. A listing of TDF.BAT is given in Table B.1 at the end of Part B.

STEP 3: Adding a new procedure to the TDF

Any new procedure (QQQ, say) will be provided on a diskette containing two files - QQQRead.Me, and QQQ.EXE. The TDF manager should copy QQQ.EXE to C:\TDF, and then consult QQQRead.Me for details of the procedure and any special operating requirements.

Useful new data-interpretation procedures will from time to time be developed by scientists or statisticians in the NRA or elsewhere. For those developing such procedures for use in the TDF, the steps to be followed are covered in a companion (CoP/TDFd) to the present Code of Practice.

From the standpoint of the TDF user, however, the only thing that matters is how the procedure is delivered to the TDF site. All procedures are given a three-letter name. For example, the program ARCTIC_SEAL (developed earlier in the study) was given the abbreviation ARC for use in the TDF. In the following discussion, the letters 'QQQ' will be used to denote any particular procedure of interest.

Thus, as Step 3 indicates, any new procedure QQQ would be sent to the TDF site on a diskette containing just the following two files:

+ QQQRead.Me - a text file providing those at the TDF site with any necessary background comments and instructions for applying procedure QQQ; and
+ QQQ.EXE - a file containing the executable code for carrying out procedure QQQ on any given data set.

STEP 4: Assembling the data for a TDF run

Using the protocol outlined in Step 1, the TDF user should extract from the quality archive the required test data files; download these to the TDF PC; and set up their AARDVARK-type control files.

Suppose the TDF user wishes to apply procedure QQQ to a particular collection of regional data sets. The details are exactly the same whether QQQ is a newly-acquired procedure being tried for the first time, or a procedure already well established on the TDF.

Incidentally, it may be that a collection of data files has already been assembled for some previous TDF run. If so, Step 4 can be bypassed.

STEP 5: Interactive or batch run

The TDF user should next decide whether the TDF run is to be a batch or an interactive run. When a new procedure is being tested, it is generally prudent to begin with the interactive option and apply QQQ to just a few data files, to check that the instructions in QQQRead.Me have been understood and followed correctly. For interactive runs, the user should proceed directly to Step 7; for batch runs, Steps 6a and 6b should first be followed.

Step 5 advises the TDF user to do some interactive exploration of the procedure before embarking on a substantial batch run. This will often be useful, for example, in deciding which types of data sets it would be most profitable to include in the evaluation of QQQ.

STEP 6a: Naming the batch input file for a TDF batch run

The 'batch input' file holds details of the files to be processed in the current batch run. It is sensible for the TDF user to make the file name reflect both the procedure name (QQQ) and the run number (ij) of that particular application of the procedure. Thus, a suitable name might for example be 'ARC07.DAT'.

It is important to maintain a proper log of the TDF investigations made and the results obtained from them. Step 6a provides one possible system that would help TDF users to keep track of their batch runs.

STEP 6b: Creating the batch input file for a TDF batch run

Next, the TDF user should create the batch input file according to the instructions given and illustrated in Table A.1.

As Table A.1 in Part A shows, there are two lines in the batch input file for every data set to be analysed: the first contains the name of the data file, and the second gives the number of required determinands, their locations, and whether or not each is to be logged.

In situations where the desired collection of data sets has already been the subject of some previous TDF run, a suitable batch input file will probably already be in existence. The TDF user can take advantage of this convenient short cut, but should be sure to work with an appropriately renamed copy of the batch input file, for the 'traceability' reasons discussed earlier.

STEP 7: Running a TDF application

To run any TDF application of procedure QQQ, the TDF user should type the command

TDF QQQ (press ENTER),

and then respond appropriately to the prompts from the PC screen.

During any TDF run, the responses called for will depend on whether it is an interactive or a batch run. The command that actually sets the run going, however, is the same in either case.

STEP 8: Handling the output from a TDF run

From a batch run, the output is sent to a file in the C:\TDF directory having the same root name as the batch input file. For interactive runs, the output is simply named TDF.OUT. In either case, if the output is required as a permanent record, it is advisable before any subsequent TDF run for the user to:

- **print the file;**
- **rename it; or**
- **transfer it to some more appropriate storage location.**

A batch output is given the same identification code as that of the corresponding batch input file, so that the two files can be kept together - whether in hard-copy form or as computer files - for future reference. Thus, when the naming convention proposed in Step 6a has been adopted, the output file will be named QQQij.OUT - where QQQ is the procedure name and ij is the run number for that application of the procedure.

The output from an interactive run is simply given the scratch name TDF.OUT. It is particularly important, therefore, to decide promptly after each run whether or not the output is to be saved, as it will be overwritten by the output from any subsequent TDF run.

Table B.1 - Listing of the TDF.BAT file

```
:
:  This is the file TDF.BAT
:  ------------------------
:
:  Date of current version: 21-Aug-1991
:
:  There is one input parameter:
:  %1 supplies the 3-letter name of the routine.
:
:----------Initialisation of TDF run---------------
:
ECHO OFF
IF EXIST TDFCMN.DAT    DEL TDFCMN.DAT
IF EXIST CONTROL.DAT   DEL CONTROL.DAT
IF EXIST NEXT.DAT      DEL NEXT.DAT
COPY NOW1.DAT NOW.DAT
:
TDFINIT
:
:----------Start of data loop-----------------
:TOP
:
TDFDATA
:
IF EXIST TDFCMN.DAT  GOTO MORE
GOTO BOTTM
:MORE
DEL NOW.DAT
RENAME NEXT.DAT NOW.DAT
:
%1
:
DEL TDFCMN.DAT
:
GOTO TOP
:----------End of data loop-------------------
:
:BOTTM
:
:============ end of file TDF.BAT ================
:
```

PART C - SETTING UP 'AARDVARK-TYPE' DATA FILES

The description that follows is a shortened version of Chapter 4 of WRc's AARDVARK User Guide, to which the reader is referred for further detail (including advice on trouble-shooting).

AARDVARK requires two input files:

* the data file itself; and
* a control file, telling AARDVARK everything it needs to know about the layout of the data file.

We describe these two types of file in the following two sections.

The data file

Name of data file

The data file may have any name of the form xxxxxxxx.DAT, where xxxxxxxx is any combination of letters and numbers (but starting with a letter) up to eight characters long. Examples of valid names are AVON8088.DAT, FINNBARR.DAT, Q0073872.DAT, and FIELD.DAT (the file illustrated in Table C.1).

Structure of data file

At the start of the data file there may be any number of rows of text information - title of the data set, determinand names, units of measurements, comments, and so on. In FIELD.DAT, for example, there are four rows of text: the data set title, the determinand names, and then two rows of comments.

After this initial header information, the file must contain a number of 'fields', or 'vertical blocks', of data. Of these,

* Fields 1 - 3 contain the sample dates (day, month & year in any specified order) moving forward through time; and then
* Fields 4, 5, 6, etc contain the data - one field per determinand.

There is no need for the fields to be adjoining. Thus, if the data file happens to contain other fields that are not of relevance, it is easy to specify in the control file that they should be skipped over.

AARDVARK requires the fields of interest to be in fixed format (as described shortly); but within that general constraint there is complete flexibility as to the choice of field width and spacing.

Table C.1 - Example of AARDVARK-type data and control files

Portion of data file FIELD.DAT

```
Field Raynes Sewage Treatment Works
                                        S.S(105)    BOD(ATU)  Amm Nit
  Data provided by Herbert Wardrobe of Central WA on 9-v-86
  Sent on mag.tape in format WQ17/32b (Option IV)
 09/01/73 1330 ---T  21 KL003     28.000                3.200
 14/02/73 1215 ---T  22 IL025     32.500                7.000
 01/03/73 1200 ---T  22 NC080     57.000                8.100
 02/04/73 1100 ---T  22 NC105     52.000                6.200
 03/05/73 1440 ---T  22 JC062     59.600                6.300
 19/06/73 1430 ---T  11 MC065     23.200                2.400
 03/07/73 1315 ---T  11 MC086     22.800                3.000
 30/07/73 1015 ---T  11 NC199     29.000                0.900
 05/09/73 1345 ---T  11 JC124     18.000                1.800
 :
 :
 :
 27/11/84 1200 D--T  11 PW592     22.000     12.800     2.700
 01/04/85 1445 ---D 522 00000     56.000     20.000     4.500
 15/05/85 1715 ---D 522 00000     36.000     18.400     1.600
 11/06/85 1445 ---D 521 00000     25.000     11.000     3.400
 04/07/85 0920 ---D 522 00000     33.000      9.000     4.100
 13/08/85 0920 ---D 521 00000     20.000      6.000     2.300
 11/09/85 1010 ---D 521 00000     17.000      8.600     2.400
 16/10/85 1420 ---D 521 00000     21.000      7.300     2.200
 28/10/85 1400 ---D 521 00000     19.000     10.000     6.600
 28/11/85 1410 ---D 521 00000     24.000      8.500     7.500
 21/01/86 0001 ---D 521 00000     25.000     11.500     1.900
 06/02/86 0930 ---D 521 00000     42.000     11.600     0.900
 03/03/86 1015 ---D 521 00000     26.000     23.500    10.300
 19/03/86 0845 ---D 521 00000     40.000     16.000     1.700
                                  999.       999.       999.
```

Control file FIELD.CTL

```
Field Raynes Sewage Treatment Works
  3
S.S(105)
BOD(ATU)
Amm Nit
4   1 2 3  999  0 -1 .5
(1X,I2,1X,I2,1X,I2, 20X, 3F10.0)
```

End of data file

There is the choice of:

* letting AARDVARK detect the end of the data file automatically; or
* using a row of dummy determinand values ("999.", for example) to flag the end of the data file.

Missing values

Gaps in the data file, known as 'missing values', most commonly arise because a particular sample was not analysed for the full set of determinands. There is the choice of representing missing values by:

* a zero;
* any other convenient numerical value - for example, "-99";
* an asterisk; or
* a blank.

Less-than values

Some data values may be reported as being less than the analytical limit of detection. These may be flagged in the data file by:

* the symbol "<"; or
* negative numbers (so, for example, "-0.05" would stand for "less than 0.05").

The control file

The control file contains all the supplementary information that AARDVARK needs to make sense of the data file.

Name of control file

The control file must have the same name as the corresponding data file, but be followed by the extension 'CTL'; and it must be stored in the same directory. For example, a data file called

\KING\PENGWYN.DAT

would have an associated control file called PENGWYN.CTL which was stored in the \KING sub-directory.

Structure of control file

In working through the following description, the reader will find it helpful to refer at each stage to the FIELD.CTL example given in Table C.1.

Row 1:	Title of data set - may be up to 40 characters long.
Row 2:	Number of determinands (nD) to be read from the data file.
Next nD rows:	Determinand titles - up to 12 characters long, one title per row.
Following row:	Eight 'data control' quantities, separated by spaces, giving detailed instructions about the various conventions to be adopted when reading the data file. These are defined below, in the section called 'Data Control row of control file'.
Final row:	The format by which AARDVARK is to read the required data from the data file. This is explained below in the section called 'Data Format row of control file'.

Data Control row of control file

The last-but-one row of the control file contains eight control quantities. These are defined as follows:

1. Number of rows in the data file to be skipped before the data starts.

2. Number of field containing Day (1, 2 or 3).
3. Number of field containing Month (1, 2 or 3).
4. Number of field containing Year (1, 2 or 3).

5. The end-of-file indicator. This may be either:
   "0" for automatic end-of-file detection; or
   any convenient non-zero quantity (e.g. "98789.") by which the end of the file is signalled: this must appear in every determinand field of the data file.

6. The missing values (MVs) indicator, defined as follows:
   "0" if MVs are represented by zero or blank in data file;
   "1" if MVs represented by "*" in data file; or
   any convenient non-zero quantity (e.g. "-99") by which MVs have been flagged in the data file.

7. The less-than indicator. This can take one of three possible values:
   "0" if there are no less-than values in the data file;
   "-1" if all negative data values are to be interpreted as less-than values; or
   "-2" if the symbol "<" is used to represent less-than values in the data file.

8. The factor by which less-than values are to be multiplied. For example:
   "0.5" would cause "less-than 0.02" to be replaced by 0.01;
   "0" would replace all less-than values by zero. (This, in conjunction with a "0" choice for the MV indicator, would then allow all less-than values to be ignored.)

Thus, in the example of FIELD.CTL, the Data Control row indicates that:

* the first 4 rows of the data file are to be skipped;

* the day, month and year values are in fields 1, 2 and 3;

* the end of file is flagged by a row of 999s;

* missing values appear as zeros or blanks;

* less-than values (if any) will be flagged by negative entries;

* where less-than values do occur, they should be replaced by 0.5 times the limit value.

Data Format row of control file

The final row of the control file specifies the format by which the required data is to be read from the data file. The characters (and) must always be the first and last to appear in the format statement, and may not be used elsewhere in the statement.

The rules governing what may be put inside the brackets are broadly the same as those applying to Fortran format statements. Take the example given in Table C.1 for FIELD.CTL, namely:

```
(1X,I2,1X,I2,1X,I2, 20X, 3F10.0).
```

This is shorthand for:

```
1X..... skip the first column;
I2..... read a 2-digit Integer;
1X..... skip a column (so as to jump over the "/" symbol);
I2..... read a 2-digit Integer;
1X..... skip another column;
I2..... read a 2-digit Integer;
20X.... skip 20 columns (so as to jump over various unwanted
          fields on the file);
3F10.0.. read 3 data values each occupying a field-width of 10
          columns (the decimal point is handled automatically
          when AARDVARK reads in the data).
```

Another example taken from the AARDVARK User Guide, for MOSS.CTL, is:

```
(1X,I2,2I3,1X,   F3.0,5F6.0,6X,9F6.0,6X,3F6.0).
```

This is shorthand for:

```
1X..... skip the first column;
I2..... read a 2-digit Integer;
2I3.... read 2 Integers from consecutive fields of width 3;
1X..... skip one column;
F3.0... read one data value occupying a field-width of 3;
5F6.0.. read 5 data values each of field-width 6;
6X..... skip the next 6 columns;
9F6.0.. read the next 9 data values each of field-width 6;
6X..... skip the next 6 columns;
3F6.0.. read the final 3 data values each of field-width 6.
```

DEVELOPING SOFTWARE FOR THE TEST DATA FACILITY

The Test Data Facility (TDF) is a standard operating protocol with supporting computer software that enables an NRA region to evaluate any proposed new method of data analysis quickly and easily using its own selected data sets. Guidance on using the TDF is provided by the Code of Practice note CoP/TDFu.

The present note is a technical supplement to that Code of Practice, and describes how to develop software applications for the TDF. It will be of interest to anyone with programming skills who wishes (or has been asked) to make a particular statistical or data-handling procedure more widely available around the NRA regions. The assumed programming language is Fortran. However, there is nothing in principle to prevent a developer from using another more convenient language.

We define the following terms for use in this Code of Practice:

a. **TDF site:** any regional office or laboratory at which a TDF has been established.

b. **TDF PC:** any microcomputer on which the TDF software has been set up. This should be a PC-compatible machine, and have a maths co-processor. In view of the potentially heavy workload to which the TDF could be subjected, a 386 machine would be desirable. As the TDF software occupies well under one Mbyte on the hard disk, there is no requirement for a dedicated PC; any suitable existing machine offering an adequate amount of hands-on time could be used.

c. **TDF manager:** the person in charge of setting up and maintaining the TDF at a particular TDF site.

d. **TDF user:** any other individual authorised to use the TDF.

e. **TDF developer:** any individual with computer programming skills who is developing a software application for use in the TDF.

The steps involved in developing software for use in the TDF are stated in Part A without elaboration. In the detailed discussion that follows in Part B, the required development steps are illustrated with the TDF application 'MOT' - a procedure which carries out the parametric test for multiple outliers described in CoP/OLR. The Fortran code used for the application is listed in Part C, and is also provided on diskette.

PART A - STEPS TO BE FOLLOWED

STEP 1: Choose a name (QQQ) for the procedure

Decide on a suitable three-letter abbreviation for the application. The letters 'QQQ' will be used to denote the user-defined name in the following description.

STEP 2: Develop the Fortran code to carry out the procedure

Develop a collection of Fortran subroutines to carry out the required data analysis procedure. Data should be passed to the subroutines through the argument list of a primary subroutine entitled QQQCalc, and the subroutines should be assembled in a file entitled QQQCalc.FOR. An example of the required structure is shown in Listing C.1.

STEP 3: Write the Fortran code that connects QQQ with the TDF

To enable the procedure QQQ to communicate with the TDF, two specialised software modules are needed: the calling program QQQ, and the control parameter subroutine QQQCon. These should be prepared as described in Steps 3a and 3b.

STEP 3a: Write the calling program QQQ

The calling program QQQ has three main functions:

+ to read each successive data file passed to it by the TDF;
+ to pick out from the current file the relevant data for each determinand in turn; and then
+ to call QQQCalc.

QQQ should be written in accordance with the requirements illustrated in Listing C.2, and stored in a file called QQQ.FOR.

STEP 3b: Write the control parameter subroutine QQQCon

The QQQCon subroutine allows the TDF user interactively to specify any desired control parameters (e.g. date range) for the current run. It should be written in accordance with the requirements illustrated in Listing C.3, and stored in a file called QQQCon.FOR.

STEP 4: Produce the file QQQ.EXE

Next, the following modules should be compiled and linked to produce the executable code in file QQQ.EXE:

+ QQQ } - Customized versions of the standard calling program
+ QQQCon } and control parameter subroutine (see Listings C.2 and C.3);
+ QQQCalc - the subroutines provided by the TDF developer (see Listing C.1); and
+ DatCom - a standard module consisting of two utility subroutines - GetData and FillCMN - used unchanged by all procedures (see Listing C.4).

STEP 5: Establish that procedure QQQ performs correctly

Procedure QQQ is now ready for testing, and should be run on a selection of data sets sufficiently varied to establish that the routine is working correctly. One convenient way of trying out QQQ is for the developer to use a TDF that has previously been installed on his or her own PC (see CoP/TDFu).

STEP 6: Establish that procedure QQQ runs correctly under TDF operation

Where this has not already been accomplished during Step 5, the developer should test procedure QQQ on a TDF to check that the data and control transfer mechanisms are functioning correctly.

STEP 7: Prepare user notes for procedure QQQ

Next, the developer should prepare a concise guide to the procedure and write this to text file QQQRead.Me. This should contain any necessary background comments about the purpose and scope of the procedure, and provide clear instructions for users at TDF sites on how it should be applied.

STEP 8: Circulate procedure QQQ

The procedure is now ready for general use. To make QQQ available to any particular TDF site, copy files QQQ.EXE and QQQRead.Me to a diskette, and send this to the TDF manager at the site.

PART B - bACKGROUND

The sequence of steps described below provides one assured method of developing a procedure for use with the TDF. This does not necessarily provide the best or most efficient development route. However, with a little experience of preparing procedures for the TDF, developers will soon see how the method can be improved or simplified to suit their particular circumstances. It will also become evident, to those who would prefer to use some other programming language, precisely what functions of the present Fortran system would need to be reproduced in any alternative system.

STEP 1: Choose a name (QQQ) for the procedure

Decide on a suitable three-letter abbreviation for the application. The letters 'QQQ' will be used to denote the user-defined name in the following description.

The illustrative application used in the discussion that follows has been named MOT - short for Multiple Outlier Test. A full account of the test can be found in the Code of Practice on handling outliers (CoP/OLR).

STEP 2: Develop the Fortran code to carry out the procedure

Develop a collection of Fortran subroutines to carry out the required data analysis procedure. Entry to the subroutines should be through the argument list of a primary subroutine entitled QQQCalc, and the subroutines should be assembled on a file entitled QQQCalc.FOR. An example of the required structure is shown in Listing C.1.

An earlier draft of this Code of Practice contained an artificially simple application, the idea being to keep the example listings as short as possible. Subsequently, however, we decided that it was preferable to use a real application of genuine practical value, even at the expense of making the example a little more complicated. This accounts in part for why the code for MOTCalc and its associated subroutines runs to five pages.

We believe it is unnecessary for the present note to provide a complete and detailed description of MOT. The code is adequately commented, and the logic should be fairly comprehensible to anyone who is familiar with Fortran. We will, however, briefly discuss one aspect that is a key component of all TDF applications, namely the method by which we supply MOTCalc with its required inputs.

All the data required by MOTCalc is passed through a parameter list in the CALL MOTCalc statement. As Listings C.1 and C.2 indicate, there are eight input parameters:

X..... a one-dimensional array containing the values for the current determinand;
NObs..... the number of values in X;
jDay.....}
jMon.....} three one-dimensional arrays containing the day, month and year of each value in X;
jYr.....}
Freshold... a user-defined parameter telling MOTCalc where to position the threshold above which data values are classed as 'possible outliers';
Screen..... a LOGICAL variable, also user-defined, indicating whether or not screen output is required; and
Logg..... a LOGICAL variable, determined automatically within MOT, indicating whether or not the current determinand consists of logged data.

This specific example gives a good pointer to what might typically appear in a QQQCalc argument list. The parameters X and NObs will be required for virtually all applications. The logical variable Screen will also be needed whenever output is produced by QQQCalc (or one of its associated subroutines). The date parameters (jDay, jMon and jYr) are rather more specialised: they will be needed only if the analysis that QQQCalc performs needs to know the temporal ordering of the data - as for example in a time series analysis.

Other input parameters will be essentially specific to the particular application - as, for example, the parameter Freshold here in MOT.

STEP 3: Write the Fortran code that connects QQQ with the TDF

To enable the procedure QQQ to communicate with the TDF, two specialised software modules are needed: QQQ, and QQQCon. These should be prepared as described in Steps 3a and 3b.

To develop QQQ and QQQCon for a new application, we have found that the easiest method is to take as a starting point a copy of the corresponding QQQ and QQQCon files from some previous application, and to modify these as required. The discussion under Steps 3a and 3b following gives practical details of how to go about this.

STEP 3a: Write the calling program QQQ

The calling program QQQ has three main functions:

+ to read each successive data file passed to it by the TDF;
+ to pick out from the current file the relevant data for each determinand in turn; and then
+ to call QQQCalc.

QQQ should be written in accordance with the requirements illustrated in Listing C.2, and stored in a file called QQQ.FOR.

The calling program MOT, shown in Listing C.2, is typical of all TDF calling programs: it is relatively short, and its structure and purpose are straightforward. The main components to note are:

+ the call to the utility subroutine GetData to obtain the current data set;
+ the determinand loop (DO 50) within which the arrays X, jDay, jMon and jYr are filled with the relevant data for the current determinand; and then
+ the MOTCalc call to apply the procedure to the data in X.

Only a few of the lines are in fact obligatory: these are the ones that we have flagged with comments such as:

<--- must have this line

Usually, however, the developer will find that, with only minor editing, nearly all of the code can usefully be pressed into service for the new application.

STEP 3b: Write the control parameter subroutine QQQCon

The QQQCon subroutine allows the TDF user interactively to specify any desired control parameters (e.g. date range) for the current run. It should be written in accordance with the requirements illustrated in Listing C.3, and stored in a file called QQQCon.FOR.

The subroutine MOTCon shown in Listing C.3 is typical of all QQQCon subroutines. It is short, and also particularly easy to customize - as witness the small number of lines flagged:

<--- specific to applicn

Non-trivial changes are called for, in fact, only when there is a need to accommodate additional control information. Here, for example, we wish the user to be able to specify the 'Freshold' parameter to be used for all data sets in the current run. Incorporating additional information such as this has ramifications in three areas of the subroutine:

+ first, the appropriate code is needed to prompt for that input on the first pass through MOTCon and then to write it <u>to</u> the control file CONTROL.DAT (unit 4);
+ next, that additional information must be read <u>from</u> the control file on subsequent passes through MOTCon; and
+ finally, the named COMMON block /MOTCMN/ must be set up so that the value of Freshold can be passed through to the calling program MOT (see Listing C.2).

Otherwise, all that is required to customize QQQCon for any particular procedure is to substitute for QQQ the required three-letter name in just a couple of places.

One other subroutine needs a similar minor adjustment: this is the 'dummy' subroutine GetCon listed in the last part of Listing C.3. Although GetCon has no function other than to call MOTCom, it does serve a useful purpose: it provides a standard means of calling the variably-named QQQCon from within the standard utility subroutine GetData in file DatCom.FOR (see Listing C.4). This ensures that GetData can be used unchanged from one application to another, and so is one less chunk of code for the developer to have to bother about.

STEP 4: Produce the file QQQ.EXE

Next, the following modules should be compiled and linked to produce the executable code of file QQQ.EXE:

+ QQQ } - Customized versions of the standard calling program
+ QQQCon } and control parameter subroutine (see Listings C.2 and C.3);
+ QQQCalc - the subroutines provided by the TDF developer (see Listing C.1); and
+ DatCom - a standard module consisting of two utility subroutines - GetData and FillCMN - used unchanged by all procedures (see Listing C.4).

This stage should present few problems (provided the developer's code is error-free). One possible difficulty that might arise is where a particular feature of the Microsoft Fortran we have used in writing MOT is not recognised by the developer's own Fortran. We have tried to minimise the risk of this happening, however, by keeping fairly closely to standard Fortran-77.

STEP 5: Establish that procedure QQQ performs correctly

Procedure QQQ is now ready for testing, and should be run on a selection of data sets sufficiently varied to establish that the routine is working correctly. One convenient way of trying out QQQ is for the developer to use a TDF that has previously been installed on his or her own PC (see CoP/TDFu).

STEP 6: Establish that procedure QQQ runs correctly under TDF operation

Where this has not already been accomplished during Step 5, the developer should test procedure QQQ on a TDF to check that the data and control transfer mechanisms are functioning correctly.

Where existing software is not being developed from scratch but merely being modified for use with the TDF, it will presumably already have received adequate testing in its previous existence. In these circumstances it will almost certainly be sensible for the developer to proceed at once to a testing of the entire system - that is, the procedure itself and the logic that couples it with the TDF.

STEP 7: Prepare user notes for procedure QQQ

Next, the developer should prepare a concise guide to the procedure and write this to text file QQQRead.Me. This should contain any necessary background comments about the purpose and scope of the procedure, and provide clear instructions for users at TDF sites on how it should be applied.

The user notes should not provide too much detail. A full account of the technique will usually be available elsewhere - in another Code of Practice, for example, or in WRc's Sampling Handbook. Similarly, the NRA's reasons for wishing to apply a particular procedure - as part of a research project, say, or to provide information to assist a national group - are better conveyed directly by the Project Leader or Group Chairman concerned.

Thus, after providing the context and giving a reference to the statistical details, the user notes should focus on the operational computing or data-handling aspects that will be of immediate relevance to the user.

The user notes for the MOT application are shown in Listing B.1.

Listing B.1 - Contents of the file MOTRead.Me

```
|-------------------------------------------------------------|
| Notes on using the MOT procedure in your Test Data Facility |
|-------------------------------------------------------------|

MOT carries out the Multiple Outlier Test on any specified set of
determinand values, as described in the Code of Practice on 'Methods
for detecting outliers' (CoP/OLR).

The test assumes that determinand values are drawn at random from an
underlying Normal distribution.  This is not as restrictive as it sounds,
because you can alternatively assume log-Normality by applying the test to
the logs of the determinand values.  So when you're setting up the batch
data file, it's a good idea to specify the log transformation option for all
determinands you think are likely to be right-skewed (in other words, just
about everything except DO, pH and temperature).  If you're not sure how to
go about this, consult the general guidance note for TDF users (CoP/TDFu).

Two other points to note in using MOT.  First, MOT uses a multiplier of 1.0
when dealing with 'less-than' values.  (In an earlier version of MOT we
discarded less-thans; but with experience we think that the present rule
is preferable.)  Secondly, you'll be prompted for a 'threshold' value.
Unless you want specifically to exclude all but the really out-rageous-liers,
just use the value 3.0 (the precise choice isn't critical).

Please contact WRc if you have any problems.  Otherwise, good luck!
```

STEP 8: Circulate procedure QQQ

The procedure is now ready for general use. To make QQQ available to any particular TDF site, copy files QQQ.EXE and QQQRead.Me to a diskette, and send this to the TDF manager at the site.

It will often be appropriate for the distribution disk to be sent just to the NRA officer who has asked for this particular procedure to be made available, and for that individual then to be responsible for circulating the procedure to the appropriate TDF sites.

PART C - TECHNICAL DETAILS

C.1 Listings of the MOT code and output

A full listing of the MOT procedure is provided in Listings C.1 to C.4. An example of the output from MOT is given in Listing C.5.

C.2 Files used by the TDF

The TDF uses a total of nine files. At the heart of the TDF is the batch file TDF.BAT shown in Listing C.6. This consists of a sequence of DOS commands which handle the alternate passing of control between program TDFData (which assembles the data from any specified data file) and the QQQ-related software which then applies procedure QQQ to that data.

Details of the other eight files are given in Tables C.1 and C.2. Table C.1 lists their contents, whilst Table C.2 charts the life history of each file, showing within which programs or subroutines each file is created, opened, read from, written to, appended to, and/or closed.

C.3 Data limitations

Number of data values and determinands
The TDF can accommodate up to 2000 data values and up to 12 determinands.

Missing-values
Depending on the TDF site, one of several conventions may be in use for representing missing values. This is no problem to the TDF, however, as the users at any particular TDF site will automatically construct their 'AARD.CTL' files so as to take account of whatever their local missing-value convention may be.

This 'decoding' takes place in program TDFData. All missing values are then written to the RAW data array as "-99." values to provide a standard convention for all developers.

'Less-than' values
Similarly, variations between one TDF site and another in the data storage convention for less-than values are coped with automatically in TDFData. The TDF then ignores whatever rule the user may have specified in the 'AARD.CTL' file, and instead follows the rule:

a value of '<L' is written to the RAW array as -L.

The advantage of this (assuming there are no genuine negative values in the data set) is that the developer can still recognise less-than values in the data set and apply whatever rule is most appropriate for the particular application. In MOT, for example, we have chosen to use a multiple of 1.0 for less-thans. This could not have been done if, say, the '0.5L' rule had been applied before the data ever reached MOT.

Listing C.1 - MOTCalc and its associated subroutines

```
C
      SUBROUTINE MOTCalc(X,NObs,jD,jM,jY,Freshold,Screen,Logg)
C                ========
C     Date of current version:  26-Oct-1991
C
C     Incoming: X.......... array containing data values for current detd;
C               NObs....... no. of values in X;
C               jD........}
C               jM........} arrays containing day, month and year nos of
C               jY........} each value in X;
C               Freshold... Standardized deviate defining 'possible outliers'
C                           threshold;
C               Screen..... LOGICAL variable set to TRUE if screen output is
C                           required;
C               Logg....... LOGICAL variable set to TRUE if current detd has
C                           been logged.
C
      DIMENSION X(NObs),jD(NObs),jM(NObs),jY(NObs),
     +           iSusp(50), Susp(50), iOmit(2000)
      LOGICAL Screen, SigLo01,SigHi01, Hunting,Logg
      CHARACTER  LoHi*4, F100*40

      F100 = '(16X,I2,''/'',I2,''/'',I2,F13.#,3X,A4)'

      CHARACTER  LoHi*4

C     Calc. overall mean & st.dev.
      S = 0
      DO 1 i = 1,NObs
         S = S + X(i)
    1 CONTINUE
      Ave = S/NObs

      SS = 0
      DO 2 i = 1,NObs
         SS = SS + (X(i) - Ave)**2
    2 CONTINUE
      StD = SQRT(SS/(NObs-1))
      nSusp = 0
      nFound = 0
      IF(StD.LT.0.00001) THEN
         PRINT *, ' Constant data! '
         GO TO 99
      END IF

C     Find out how many data values are more than "Freshold" st.devs
C     from the mean.  Save their locations in order, from least to most
C     extreme, in array iSusp.  Also fill up iOmit array with 0s & 1s...
C
      DO 3 i = 1,NObs
         t = ABS((X(i) - Ave)/StD)
         iOmit(i) = 0
         IF(t.GT.Freshold) THEN
            iOmit(i) = 1
            nSusp = nSusp + 1
```

```
            IF(nSusp.GT.50) THEN
               WRITE(6,110)
               WRITE(7,110)
  110          FORMAT(' Sorry!  Threshold was too low.',
     +                ' Try again with higher threshold...')
               RETURN
            END IF

            jSlot = nSusp
            IF(nSusp.GT.1) THEN
               Hunting = .TRUE.
               DO 4 j = 1,nSusp-1
                  IF(Susp(j).LT.t) GO TO 4
                  IF(Hunting) jSlot = j
                  Hunting = .FALSE.
    4          CONTINUE
               IF(jSlot.LT.nSusp) THEN
                  DO 5 j = nSusp-1,jSlot,-1
                     Susp(j+1) = Susp(j)
                     iSusp(j+1) = iSusp(j)
    5             CONTINUE
               END IF
            END IF
            iSusp(jSlot) = i
            Susp(jSlot) = t
         END IF
    3  CONTINUE
       IF(nSusp.EQ.0) GO TO 99

C      Do the single-outlier test up to nSusp times.  Once an outlier HAS
C      been identified, it's assumed that all more extreme data values are
C      outliers too, so CalcOT can be skipped...
C
       iAdd = 0
   10  iAdd = iAdd + 1
       nOmit = nSusp - iAdd
       jAdd = iSusp(iAdd)
       iOmit(jAdd) = 0
       IF(nFound.EQ.0) THEN

C         -----------
          CALL CalcOT(NObs,X,nOmit,iOmit,tmin,tmax,itmin,itmax,
C         -----------
     +                                        tc,SigLo01,SigHi01)

          IF(SigLo01) THEN
             LoHi = 'low'
             nFound = nFound + 1
             ZZZ = X(itmin)
             IF(Logg) ZZZ = 10**ZZZ
             nDP = 3
             IF(ABS(ZZZ).GT.1.) nDP = 2
             IF(ABS(ZZZ).GT.100.) nDP = 1
             WRITE(F100(27:27),FMT='(I1)') nDP
             IF(Screen)
     +       WRITE(6,FMT=F100) jD(itmin),jM(itmin),jY(itmin),ZZZ, LoHi
             WRITE(7,FMT=F100) jD(itmin),jM(itmin),jY(itmin),ZZZ, LoHi
          END IF
```

```
      IF(SigHi01) THEN
         LoHi = 'high'
         nFound = nFound + 1
         ZZZ = X(itmax)
         IF(Logg) ZZZ = 10**ZZZ
         nDP = 3
         IF(ABS(ZZZ).GT.1.) nDP = 2
         IF(ABS(ZZZ).GT.100.) nDP = 1

         WRITE(F100(27:27),FMT='(I1)') nDP
         IF(Screen)
     +      WRITE(6,FMT=F100) jD(itmax),jM(itmax),jY(itmax),ZZZ, LoHi
         WRITE(7,FMT=F100) jD(itmax),jM(itmax),jY(itmax),ZZZ, LoHi
      END IF

      ELSE

         nFound = nFound + 1
         LoHi = 'high'
         ZZZ = X(jAdd)
         IF(ZZZ.LT.Ave) LoHi = 'low'
         IF(Logg) ZZZ = 10**ZZZ
         nDP = 3
         IF(ABS(ZZZ).GT.1.) nDP = 2
         IF(ABS(ZZZ).GT.100.) nDP = 1

         WRITE(F100(27:27),FMT='(I1)') nDP
         IF(Screen)
     +      WRITE(6,FMT=F100) jD(jAdd),jM(jAdd),jY(jAdd),ZZZ, LoHi
         WRITE(7,FMT=F100) jD(jAdd),jM(jAdd),jY(jAdd),ZZZ, LoHi

      END IF

      IF(iAdd.LT.nSusp) GO TO 10

   99 IF(nFound.EQ.0) THEN
         IF(Screen) WRITE(6,102)
         WRITE(7,102)
  102    FORMAT(35X,'No outliers found.')
      END IF

      RETURN
      END
C=======================================================
```

```
C
      SUBROUTINE CalcOT(NObs,X,nOmit,iOmit,tmin,tmax,itmin,itmax,
C                       ======
     +                                        tcrit,SigLo01,SigHi01)
C
C     Date of current version:  24-Oct-1991
C
C     Incoming: X........ array containing data values (including suspects).
C               NObs..... total no. of data values in X.
C               nOmit.... no. of suspects in X.
C               iOmit.... array containing 1 for each X value to be omitted,
C                                 and 0 otherwise.
C
C     Outgoing: tmin..... test statistic for minimum data value.
C               tmax..... test statistic for maximum data value.
C               itmin.... obs.no. of minimum value.
C               itmax.... obs.no. of maximum value.
C               tcrit.... 1% critical point for test statistic.
C               SigLo01.. LOGICAL variable set to TRUE if tmin is
C                                       significant at 1% level.
C               SigHi01.. LOGICAL variable set to TRUE if tmax is
C                                       significant at 1% level.
C
C     CalcOT carries out the Normality-based outlier test described in
C     CoP/OLR.  Both the minimum & the maximum are tested at the one-sided
C     1% significance level, and the LOGICAL variables SigLo01 and SigHi01
C     set accordingly.  If there are fewer than 4 obs or the st.dev. is zero,
C     the tests are not performed.
C
      DIMENSION X(NObs), iOmit(NObs)
      LOGICAL SigLo01,SigHi01

      SigLo01 = .FALSE.
      SigHi01 = .FALSE.
      NuN = NObs - nOmit
      IF(NuN.GE.4) THEN

C     Determine critical test statistic.

C     -----------
      CALL OTCrit(NuN,tCrit)
C     -----------

C     Calc. min, mean and max.
         tmin = 999999.
         tmax = -999.
         S = 0.
         DO 1 i = 1,NObs
            IF(iOmit(i).EQ.1) GO TO 1
            S = S + X(i)
            IF(X(i).LT.tmin) THEN
               tmin = X(i)
               itmin = i
            END IF
            IF(X(i).GT.tmax) THEN
               tmax = X(i)
               itmax = i
            END IF
    1    CONTINUE
```

```
         Ave = S/NuN

C        Calc. st.dev.
         SS = 0.
         DO 2 i = 1,NObs
            IF(iOmit(i).EQ.1) GO TO 2
            SS = SS + (X(i) - Ave)**2
    2    CONTINUE

         IF(SS.GT.0.000001) THEN
            StDev = SQRT(SS/(NuN-1))
            tmin = (tmin-Ave)/StDev
            IF(ABS(tmin).GT.tCrit) SigLo01 = .TRUE.
            tmax = (tmax-Ave)/StDev
            IF(tmax.GT.tCrit) SigHi01 = .TRUE.
         END IF

      END IF

      RETURN
      END
C
C=======================================================
C
C
      SUBROUTINE OTCrit(NObs,tCrit)
C                ======
C     Incoming: NObs
C     Outgoing: tCrit
C
C     OTTCrit evaluates an empirical approximation to the 1% critical points
C     for the Normality-based outlier test described in CoP/OLR.
C
C     Date of current version:  18-Oct-1991
C
      XN = NObs
      IF(NObs.LE.10) THEN
         tCrit = EXP(1.2026 - 3.259/XN + 0.182/XN**2)
      ELSEIF(NObs.LE.20) THEN
         tCrit = EXP(1.2577 - 4.208/XN + 4.255/XN**2)
      ELSE
         tCrit = EXP(1.3670 - 9.190/XN + 60.52/XN**2)
      END IF
      RETURN
      END
C=======================================================
```

Listing C.2 - The calling program MOT

```
C
      PROGRAM MOT
C     ===
C     This is an example of a TDF application.
C     MOT carries out the Normality-based Multiple Outlier Test as described
C     in Code of Practice note CoP/OLR.
C
C     Date of current version:  20-Apr-1992
C
      COMMON                                                  <--- must have this
     +          ID(2000),IM(2000),IY(2000),RAW(2000,12),       COMMON block
C       Day, month & year, and RAW detd values, for up to 2000 obsns.
C
     +          Ndets,                NRows,
C       No.of detds (= columns) & observations (= rows) in RAW array.
C
     +          NofX(12),
C       No.of valid observns in each column of the RAW array.
C
     +  NObs,                  IY1,        IY2,
C       No.of obsns for        Beginning & End years
C       current data set.      for the current analysis
C
     +  DTit,                           WTit1, WTit2,
C       12-char labels for the         Two 70-col strings for the
C       full set of upto 12 detds.     grand heading of the data set.
C
     +  WTit
C       40-char string extracted from WTit1(1:40)
C
      CHARACTER   DTit(12)*12, WTit1*70, WTit2*70,     <--- must have this line
     +            WTit*40
C
C     MOTCMN is needed to pick up control info. from MOTCon.
      COMMON /MOTCMN/Freshold                          <--- must have this line
                                                            if add'l control
                                                            info. is required
                                                            for this applicn
      DIMENSION X(2000), jDay(2000),jMon(2000),jYr(2000)

      LOGICAL Batch,Screen,Logg                        <--- must have this line

C     ------------
      CALL GetData(Batch,Screen,Next)                  <--- must have this line
C     ------------
C
      IF(Screen) WRITE(6,100) WTit, Freshold
      WRITE(7,100) WTit, Freshold
  100 FORMAT(////' |',60('-'),'|'/
     +  ' |     Output from the Multiple Outlier Test program MOT',
     +                                                    6X,'|'/
     +  ' |     Site name: ',A40,4X,'|'/' |',60('-'),'|'//
     +  '       Note: 1. The test assumes Normality.'/
     +  '             2. A multiplier of 1.0 is used for less-thans.'/
     +  '             3. The "tmax" threshold used is',F6.2//)
```

```
      IF(Screen) WRITE(6,110)
      WRITE(7,110)
  110 FORMAT(///'                  Date of      Value              '/
     +         ' Determinand      sample     (unlogged)   Comment'/
     +         ' ------------   --------   ----------   -------')
C
C-----Loop through each detd in turn...------------
C
      DO 50 jD = 1,NDets
         NN = NRows
         NObs = 0
         Logg = .FALSE.
         IF(DTit(jD)(1:3).EQ.'Log') Logg = .TRUE.

         DO 60 i = 1,NN
            Why  = RAW(i,jD)
C           Deal with any less-thans of unlogged data using a
C           multiplier of 1.0...
            IF(Why.LT.0. .AND. Why.GT.-99. .AND. .NOT.Logg) Why = -1.0*Why

            IF(Why.GT.-99. .AND. IY(i).GE.IY1 .AND. IY(i).LE.IY2) THEN

C              Value is valid and is in the required date range, so
C              include it in the X array and save its date info. in the
C                                                  jDay, jMon & jYr arrays.
               NObs = NObs + 1
               jDay(NObs) = ID(i)
               jMon(NObs) = IM(i)
               jYr(NObs) = IY(i)
               X(NObs) = Why
            END IF
   60    CONTINUE
C
         IF(Screen) WRITE(6,111) DTit(jD)
         WRITE(7,111) DTit(jD)
  111    FORMAT(1X,A12)

         IF(NObs.LT.4) THEN
            WRITE(6,112) NObs
            WRITE(7,112) NObs
  112       FORMAT(15X,'Too few data values! (N =',I2,')' )
         ELSE
C           ------------
            CALL MOTCalc(X,NObs,jDay,jMon,jYr,Freshold,Screen,Logg) <------ must
C           ------------                                              have this line

         END IF
   50 CONTINUE
C-----End of detd loop...-------------------------

      WRITE(7,120) Next                                  <--- must have this line
  120 FORMAT(/2X,15('=-=-')/I6)                          <--- must have this line
      CLOSE (7)                                          <--- must have this line

      STOP ' '
      END
C=============================================================
```

Listing C.3 - The control parameter subroutine MOTCon

```
C
      SUBROUTINE MOTCon(OutFil,Next)                    <--- specific to applicn
C               ======
C     This obtains whatever control information is reqd. for running MOT.
C
C     Incoming: OutFil..... name of output file (previously set up by TDF)
C               Next........ number of next data set to be processed
C
C     Date of current version:  24-Apr-1992
C
      COMMON
     +          ID(2000),IM(2000),IY(2000),RAW(2000,12),
C       Day, month & year, and RAW detd values, for up to 2000 obsns.
C
     +          Ndets,                NRows,
C       No.of detds (= columns) & observations (= rows) in RAW array.
     +          NofX(12),
C       No.of valid observns in each column of the RAW array.
     +  NObs,                  IY1,        IY2,
C       No.of obsns for        Beginning & End years
C       current data set.      for the current analysis
C
     +  DTit,                             WTit1, WTit2,
C       12-char labels for the            Two 70-col strings for the
C       full set of upto 12 detds.        grand heading of the data set.
     +  WTit
C       40-char string extracted from WTit1(1:40)
C
      CHARACTER  DTit(12)*12, WTit1*70, WTit2*70, WTit*40
C
      COMMON /MOTCMN/ Freshold                          <--- specific to applicn
      CHARACTER OutFil*12, ZZZZ*4

      IF(Next.EQ.2) THEN
C         Prompt for reqd control information
C         -----------------------------------
          WRITE(6,300)
  300     FORMAT(////
     +     '    Running program MOT'/                  <--- specific to applicn
     +     '    -------------------'/)

    1     WRITE(6,301)
  301     FORMAT(
     +    '   Please input the earliest and latest years for which'/
     +    '   you''d like data to be included  (e.g. 85 92)... ')
          READ *, IY1, IY2

          IF(IY1.GT.IY2) THEN
              PRINT *, ' Uh??? '
              GO TO 1
          END IF
          WRITE(4,*) IY1,IY2
```

```
C          Additional control input required...                          }
                                                                         }
           WRITE(6,302)                                                  }
  302      FORMAT(/                                                      }
     + '    You also need to tell MOT what threshold value'/            } specific
     + '    of the "tmax" statistic it should use for defining'/        }
     + '    suspicious data values.  By way of guidance, here are'/     }
     + '    the tmax values needed (for various sample numbers) for'/}
     + '    identified outliers to be significant at the 1% level:'/ }
     +/'    No. of  values:      6     12     20     50    100    '/  }  to
     + '              tmax:    1.94   2.55   2.88   3.34   3.60  '// }
     + '    ... so what tmax value would you like?'/                    }
     + '                      (if in doubt, try 3.0)... ')              }
                                                                         }
           READ *, Freshold                                              } applicn
           WRITE(4,*) Freshold                                           }
           OPEN(7,FILE=OutFil)

       ELSE

C          Get reqd control info. from previously created file...
C          ------------------------------------------------
           READ(4,*) IY1,IY2
           READ(4,*) Freshold                          <--- specific to applicn
C
           OPEN(7,FILE=OutFil,STATUS='OLD')

C          Wind on to just before end of file...
    10     READ(7,305) ZZZZ
  305      FORMAT(2X,A4)
           IF(ZZZZ.EQ.'=-=-') THEN
              READ(7,306) iZZZZ
  306         FORMAT(I6)
              IF(iZZZZ.EQ.(Next-1)) GO TO 11
           END IF
           GO TO 10
    11     CONTINUE
       END IF
       CLOSE (4)
       RETURN
       END
C
C=======================================================================
C
       SUBROUTINE GetCon(OutFil,Next)
C                 ======
C      This simply calls MOTCon.
C
C      Date of current version: 24-Oct-1991
C
       CHARACTER OutFil*12
C
C      -----------
       CALL MOTCon(OutFil,Next)                    <--- specific to applicn
C      -----------

       RETURN
       END
```

Listing C.4 - The standard utilities file DatCom

```
C
      SUBROUTINE GetData(Batch,Screen,Next)
C                ========
C     Date of current version:  25-Mar-1992
C
      COMMON
     +          ID(2000),IM(2000),IY(2000),RAW(2000,12),
C       Day, month & year, and RAW detd values, for up to 2000 obsns.
C
     +          Ndets,                NRows,
C       No.of detds (= columns) & observations (= rows) in RAW array.
C
     +          NofX(12),
C       No.of valid observns in each column of the RAW array.
C
     +  NObs,                  IY1,        IY2,
C       No.of obsns for        Beginning & End years
C       current data set.      for the current analysis
C
     +  DTit,                             WTit1, WTit2,
C       12-char labels for the            Two 70-col strings for the
C       full set of upto 12 detds.        grand heading of the data set.
C
     +  WTit
C       40-char string extracted from WTit1(1:40)
C
      CHARACTER  DTit(12)*12, WTit1*70, WTit2*70, WTit*40
C
C
      CHARACTER BatFil*12, OutFil*12

      LOGICAL Batch,Screen

      OPEN(4,FILE='CONTROL.DAT',STATUS='OLD')
      READ(4,110) BatFil
  110 FORMAT(A12)
      READ(4,*) Batch,Screen

C     Create name of Output file...
      IF(Batch) THEN
          OutFil = BatFil(1:8)//'.OUT'
      ELSE
          OutFil = 'TDF.OUT'
      END IF

C     Fill up contents of TDF.CMN...

      CALL FillCMN
C     ============
```

```
C     See whether it's the first pass...
      OPEN(11,FILE='Now.DAT',STATUS='OLD')
      READ(11,*) Next
      CLOSE (11)
C     A 'Next' value of 2 means that the prog. is on the first pass...

C     Get control information...

      CALL GetCon(OutFil,Next)
C     ==========

      RETURN
      END
C=====================================================================
C
      SUBROUTINE FillCMN
C                *******
C     This fills up contents of TDF.CMN...
C     Date of current version:  26-Jun-1991
C
      COMMON
     +          ID(2000),IM(2000),IY(2000),RAW(2000,12),
C       Day, month & year, and RAW detd values, for up to 2000 obsns.
C
     +          Ndets,               NRows,
C       No.of detds (= columns) & observations (= rows) in RAW array.
C
     +          NofX(12),
C       No.of valid observns in each column of the RAW array.
C
     +  NObs,                  IY1,        IY2,
C       No.of obsns for        Beginning & End years
C       current data set.      for the current analysis
C
     +  DTit,                           WTit1, WTit2,
C       12-char labels for the         Two 70-col strings for the
C       full set of upto 12 detds.     grand heading of the data set.
     +  WTit
C       40-char string extracted from WTit1(1:40)
C
      CHARACTER   DTit(12)*12, WTit1*70, WTit2*70, WTit*40
C
      OPEN(21,FILE='TDFCMN.DAT',STATUS='OLD')

      READ(21,*) Ndets,NRows

      READ(21,100) WTit
  100 FORMAT(A70)
      READ(21,101) (DTit(j),j=1,NDets)
  101 FORMAT(5(2X,A12))

      DO 1 i = 1,NRows
            READ(21,*) ID(i),IM(i),IY(i),(RAW(i,j),j=1,NDets)
    1 CONTINUE

      RETURN
      END
C=====================================================================
```

Listing C.5 - Example of the output from procedure MOT

```
-------------------------------------------------------------
     Output from the Multiple Outlier Test program MOT
     Site name: Langham Raw Water
-------------------------------------------------------------

      Note: 1. The test assumes Normality.
            2. A multiplier of 1.0 is used for less-thans.
            3. The "tmax" threshold used is  3.00

                Date of      Value
Determinand      sample    (unlogged)   Comment
------------    --------   ----------   -------
Raw Condtvty
                                        no outliers found

Raw pH
                                        no outliers found

Log_Raw Alka
                 7/10/82        198.0   low
                 2/12/83        194.0   low
                12/ 5/82        194.0   low
Log_Raw Amm_
                12/ 8/85        1.000   low
Raw Nitrate
                                        no outliers found

=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

-------------------------------------------------------------
     Output from the Multiple Outlier Test program MOT
     Site name: Mole at Wick Farm, Horley
-------------------------------------------------------------

      Note: 1. The test assumes Normality.
            2. A multiplier of 1.0 is used for less-thans.
            3. The "tmax" threshold used is  3.00

                Date of      Value
Determinand      sample    (unlogged)   Comment
------------    --------   ----------   -------
Hr of samplg
                31/ 7/89        21.00   high
                19/ 6/89        21.00   high
                 1/ 8/89        1.000   low
                31/ 7/89        23.00   high
Log_BOD(ATU)
                23/11/82         .400   low
                25/ 1/90        33.70   high
Log_Amm.Nit.
                                        no outliers found

DO (% sat)
                                        no outliers found

=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

Listing C.6 - Contents of the batch file TDF.BAT

```
:
:  This is the file TDF.BAT
:  ------------------------
:
:  Date of current version: 21-Aug-1991
:
:  There is one input parameter:
:  %1 supplies the 3-letter name of the routine.
:
:----------Initialisation of TDF run---------------
:
ECHO OFF
IF EXIST TDFCMN.DAT   DEL TDFCMN.DAT
IF EXIST CONTROL.DAT  DEL CONTROL.DAT
IF EXIST NEXT.DAT     DEL NEXT.DAT
COPY NOW1.DAT NOW.DAT
:
TDFINIT
:
:----------Start of data loop-----------------
:TOP
:
TDFDATA
:
IF EXIST TDFCMN.DAT  GOTO MORE
GOTO BOTTM
:MORE
DEL NOW.DAT
RENAME NEXT.DAT NOW.DAT
:
%1
:
DEL TDFCMN.DAT
:
GOTO TOP
:----------End of data loop-------------------
:
:BOTTM
:
:============ end of file TDF.BAT ================
:
```

Table C.1 - Description of the main files used by the TDF

Unit no.	File name	Contents
4	CONTROL.DAT	'QQQij.BAT' (the name of the batch data file); the LOGICAL variables Batch & Screen; the start & end years to be covered by the analyses
1	'AARD.DAT'	The current AARDVARK-type data set
2	'AARD.CTL'	Control file for the current AARDVARK-type data set
11	NOW.DAT	Sequence no. of current data set - overwritten by NEXT.DAT as soon as the current data set has been read by ReadData
12	NEXT.DAT	Sequence no. of next data set to be read
22	'BatFil'	Row 1: filename of the 1st data set to be analysed, & Row 2: details of the detds reqd from that data set Row 3: \ details of the 2nd data set; Row 4: / Row 5: \ details of the 3rd data set; Row 6: / and so on.
21	TDFCMN.DAT	The data 'drained' out of TDF COMMON by subroutine DrainCMN, and then read into QQQ COMMON by subroutine FillCMN
7	'OutFil'	The sequential output for all data sets in current TDF run - named 'QQQij.OUT' for batch runs, and 'TDF.OUT' for interactive runs.

Table C.2 - File usage and control during a run of the TDF

File no.	File name		TDF system software: Program TDFInit	TDF system software: Subroutines of prog. TDFData... ReadData	TDF system software: Subroutines of prog. TDFData... DrainCMN	QQQ-related software: Standard subroutines GetData	QQQ-related software: Standard subroutines FillCMN	QQQ-related software: Subroutine QQQCon: First call	QQQ-related software: Subroutine QQQCon: Subseq. calls	QQQ-related software: Prog.QQQ or related subroutine
4	CONTROL.DAT		Cr/Wr/Cl	Op/Re/Cl		Op/Re		Ap/Cl	Re/Cl	
1	'AARD.DAT'	a		Op/Re/Cl						
2	'AARD.CTL'	a		Op/Re/Cl						
11	NOW.DAT	b		Op/Re/Cl		Op/Re/Cl				
12	NEXT.DAT			Cr/Wr/Cl						
22	'QQQij.DAT'	a		Op/Re/Cl						
21	TDFCMN.DAT				Cr/Wr/Cl		Op/Re/Cl			
7	'OutFil'	c						Cr	Op/En	Wr/Cl

Note: a These files are set up and named by the user prior to the TDF run.
b This file is created by the TDF batch command logic (file TDF.BAT).
c This file is named automatically during the TDF run.

Key: Cr - create file Op - open file Re - read from file Wr - write to file
Ap - append to file En - go to end of file Cl - close file

GUIDELINES AND METHODS FOR DATA QUALITY CONTROL

This Code of Practice is in three parts. First, Part A gives a list of general rules that form an effective basis for any system of Data Quality Control (DQC). Part B then provides some background to DQC, and also explains each rule in detail. Finally, Part C describes various statistical tests and procedures that can be of practical value in screening data.

PART A - RULES

RULE 1: Incongruous data

Eliminate incongruous data by applying Rules 1a, 1b and 1c.

RULE 1a: Database structure

Structure any system used to store data so that it is possible to extract any subset of the data with the necessary fineness of detail.

RULE 1b: Archiving

When archiving data, use sufficiently precise referencing, including determinand code, purpose code, method code, date and time, to ensure that users may retrieve as finely detailed a subset as they require.

RULE 1c: Retrieving

When retrieving data for analysis, always have a clear objective, and make sure that sufficiently precise referencing is used to ensure that unwanted, misleading or irrelevant data is not included.

RULE 2: Data Quality Control

Develop a Data Quality Control (DQC) system according to the principles set out in Rules 2a to 2d.

RULE 2a: Procedures

Develop quality control procedures which, briefly but clearly, state what must be done - where, when and by whom - for every activity from sample collection, through transportation, preservation and analysis to data screening and archiving.

RULE 2b: Minimum Requirements

Tailor the quality control system to suit local clerical, hardware and software systems, but ensure as a minimum that the most common sources of error are guarded against (see Part B).

RULE 2c: Data screening

The methods of Part C of this Code of Practice, and those detailed in Code of Practice CoP/OLR on 'Methods for handling outliers' should be applied to all new data (and in due course to all existing data) to check for the presence of erroneous data values.

RULE 2d: Documentation

Ensure that up-to-date data quality procedure manuals are in place and easily available, and that staff are fully trained in using them.

GUIDELINES AND METHODS FOR DATA QUALITY CONTROL

PART B - BACKGROUND

B.1 ERRORS IN DATA SETS

Errors and outliers

This Code of Practice discusses methods of preventing the occurrence of erroneous values in data sets, and of detecting and correcting such values when they do occur.

One special kind of error is the 'outlier'. Whilst the techniques described here will succeed in detecting some types of outlier, there are other more specific outlier tests available. As the detection of outliers is a large subject in its own right, we have made this the subject of a separate Code of Practice (CoP/OLR) - which should, therefore, be regarded as a companion to the present document.

Sampling error and analytical error

'Sampling error' is a long-established term used to describe the inevitable variability inherent in any results derived from sampling. 'Analytical error' is a similarly familiar term describing the uncertainty associated with the results from a particular analytical method. The use of the word 'error' in these expressions is unfortunate, as in neither case should there be any connotations of 'mistake'.

We mention this point both by way of clarification, and also to emphasize that this Code of Practice is very much concerned with errors in the literal sense of the word.

Effects - why errors matter

The presence of undiscovered errors in a data set has many unwelcome consequences. Estimates of parameters and their confidence intervals will be wrong; hypothesis tests may give the wrong conclusion; erroneous Look-up Table failures will occur (leading to incorrect compliance verdicts); and misleading indications of trend may be given. Errors in data are particularly undesirable where sampling frequencies are low, as the uncertainties due to sampling and analysis alone are likely to be quite bad enough.

These consequences lead to attention (and capital) being diverted away from where it is really needed, thus reducing the pace of environmental improvement. Perhaps of more immediate concern is the fact that if any aspect of the data is questionable, then prosecutions against offenders become more difficult to sustain.

Causes - how errors arise

Errors can occur anywhere in the sometimes complex and ill-defined route between the collection of the physical sample and the archiving of the final analytical results. The error takes the form of an addition of 'noise' to the 'signal'. The source of the noise may be clerical error, software problems, or the effects of poor practice in sample collection, sample treatment and analysis.

Each of these factors is discussed in turn below.

Clerical errors

The following are major, commonly occurring causes of clerical error:

+ **poor procedures** - procedures may be poorly thought-out, poorly written or over-complex and prone to misinterpretation;

+ **unclear responsibility** - responsibilities and accountabilities may be unclear, and an operative may be quite unaware - perhaps because of inadequate training - that a particular part of the clerical procedure is one of his duties;

+ **oversight** - a step in the analysis may be overlooked, such as allowance for a dilution factor, or converting from imperial to metric units; the wrong units or units code may be used when recording the result;

+ **falsification** - for whatever reason, an operative may deliberately falsify a result;

+ **preference** - an operative may believe that he knows better than the result which the analysis gives, and may adjust it in line with his preference;

+ **accidental inversion** - this is the common phenomenon of digits being swapped: for example, 397.2 being wrongly written as 937.2;

+ **accidental repetition** - the same digit may be written twice: for example, 992.8 instead of 92.8.

Errors due to software faults

Software may be poorly written. A common way for this to happen is that a computing enthusiast working in a laboratory writes some handy programs to make his life a little easier, and later gives copies to colleagues. The programs, which were probably hurriedly written and may contain errors, then become a de facto standard.

One such problem encountered in reality arose through a laboratory data management system using a dummy 'placeholder' value, 9999, to 'save a place' in the computer file for data not yet available. This, combined with the absence of safeguards against archiving incomplete samples, meant that values of 9999 could find their way into the archive. Without the benefit of Data Quality Control, moreover, it was years before this particular problem came to light.

Errors due to sample collection, treatment, transport and analysis
Samples may not always be taken at the correct sampling point, and may be attributed to the wrong date and/or time. Situations abound where spatial variation and temporal variation are such that the concentration of a determinand can change substantially with a slight change in location, day of the week, or time of day.

If the actual technique used to collect the sample is itself inconsistent, this will introduce errors. Details such as what size of scoop, what size of bucket, whether a snap sample, flow-composite or time-composite sample is required, and how to clean and prepare sample bottles should all be clearly specified and consistently adhered to.

If a clear clerical procedure is not worked out and properly followed, then samplers will develop their own ad hoc methods. As a result, samples may be swapped or wrongly labelled.

Particularly now that laboratories are being reduced in number, samples may be subjected to extremes of temperature and long delays during transport. This may have severe effects on any determinands which are highly dependent on temperature or are unstable over time.

If a sample needs to be stored for a length of time after reaching the laboratory, it may need to be stored in specific conditions to ensure stability. If these conditions are not maintained, either through oversight or accident, the final analytical result may be spurious and give rise to errors.

There may be a calibration fault, such as a consistent bias, associated with a particular method or instrument. For example, a thermocouple may be known to give a temperature which is always two degrees high. If the operator is aware of it he can make allowance, and there is no problem; but if the correction has not been documented and written into the procedure, the operator is unaware of it and errors will result.

An inappropriate method may be used, or an appropriate method may be used wrongly, so that the analytical result can only be said to be 'less than X', or 'greater than X'. The result may then be recorded as X, thus causing an error. A common cause of this problem is the use of the wrong dilution factor during chemical analysis.

Analytical quality control (AQC) may be inadequate, and severe bias and imprecision can combine to give wildly varying results.

Errors due to electro-mechanical faults
These are computer breakdowns, and communications and telemetry faults. In modern systems, computer hardware is relatively reliable and line protocols used in communications and telemetry have a large measure of error detection and validation built in, using error-correcting codes. As a result, problems in this area are relatively rare.

B.2 PREVENTION - HOW TO AVOID OR DETECT ERRORS

Errors caused by the database system

Before proceeding to the question of erroneous data values, it is worthwhile considering what happens when a data set is retrieved from the archive or database, and analysed. Although the data values themselves may be quite valid, it is possible that, in a specific analysis, some members of the data set may not be appropriate to the task in hand. In this way a data value, although valid in its context, can become, in effect, an outlier when used out of context.

RULE 1: Incongruous data

Eliminate incongruous data by applying Rules 1a. 1b and 1c.

RULE 1a: Database structure

Structure any system used to store data so that it is possible to extract any subset of the data with the necessary fineness of detail.

Here, 'data quality' is measured by the database's completeness, consistency and reliability. The data structure must be relevant to users' decision making needs, and interrelationships among types of data must be well catered for. If the database system makes no allowance for storing data with the necessary fineness of detail, clearly it will not be possible subsequently to retrieve data to that degree of detail.

RULE 1b: Archiving

When archiving data, use sufficiently precise referencing, including determinand code, purpose code, method code, date and time, to ensure that users may retrieve as finely detailed a subset as they require.

If sufficient detail is not used when archiving the data, quite different types of data may become indistinguishable and a data set retrieved later may be quite inappropriate to the purpose for which it was obtained.

This may happen, for example, in the case of a data set obtained for the same determinand at each of several labs, one of which uses a method with poor precision. If lab code and method code are not correctly recorded when archiving such data, it will never be possible subsequently to distinguish either between labs or between methods.

RULE 1c: Retrieving

When retrieving data for analysis, always have a clear objective, and make sure that sufficiently precise referencing is used to ensure that unwanted, misleading or irrelevant data is not included.

It is always vital to have a clear objective when setting out to analyse data, and to ensure that the data retrieved is appropriate for that objective. Lack of sufficient care in this regard is one the most common causes of apparently erroneous data.

For example, to make a statement about the state of a particular sewage works effluent under routine conditions, only data derived from samples collected under routine conditions should be analysed. On the other hand, a statement about conditions during a particular pollution incident should be based solely on data derived from samples collected during the incident.

If the sample purpose is not specified - by use of a purpose code, for example - then samples collected for routine purposes will become indistinguishable from those taken as follow-up to a pollution incident. The data set later retrieved will then appear to contain errors.

Errors in the data

RULE 2: Data Quality Control

Develop a Data Quality Control (DQC) system according to the principles set out in Rules 2a to 2d.

RULE 2a: Procedures

Develop quality control procedures which, briefly but clearly, state what must be done - where, when and by whom - for every activity from sample collection, through transportation, preservation and analysis to data screening and archiving.

When considering the actual data values themselves, our objective must be to minimise the chance of an error occuring. Moreover, there must be a clear 'audit trail' which permits back-tracking through the life of a data value, so that the sources of any errors which *do* occur may be found. In achieving this state of affairs, the importance of having written data quality procedures cannot be overstated. These should preferably be constructed in accordance with the British Standard on Quality Systems, BS 5750, and should cover the whole chain of events from sample collection to data archiving.

RULE 2b: Minimum Requirements

Tailor the quality control system to suit local clerical, hardware and software systems, but ensure as a minimum that the most common sources of error are guarded against (see Part B).

Every region has a different organisational structure, a different set of clerical procedures and computer hardware and software systems, a different set of geo-demographic and logistical problems, different mixes of staff experience and training, and widely differing sets of water quality problems. This makes it impossible to prescribe a fixed single DQC procedure for all regions. It is, however, possible to give general guidelines, and we have identified the following nine features as essential elements of any DQC procedure.

1. The DQC procedure should list and define every step in every process from collecting the sample to archiving the results.

2. The DQC procedure should give the name and/or position of the 'designated officer' responsible for guaranteeing proper completion of each step in the procedure.

3. Anyone whose suspicions are aroused by an unusual data value should immediately contact the designated officer to whom responsibility for that particular data has been given by the DQC procedure.

4. Anyone who does not have, or cannot find, a copy of the DQC manual should request one, and not simply assume that sooner or later someone else will follow the correct procedure.

5. The DQC procedure should provide a clear audit trail mechanism whereby suspicious data values may be traced back to discover whether and where there is an error.

6. In the event of a suspected error, the designated officer should follow the audit trail back towards the source.

7. If the designated officer finds evidence that the data value is in error, he should take the appropriate action as follows:

   + if possible, correct the error (where, for example, it occurred through a miscalculation or a wrong dilution);
   + if the error cannot be corrected, then if possible re-analyse an unused portion of the sample;
   + if there is no portion of the original sample left, and not too much time has elapsed, then re-sample if feasible.

8. If a data value is known to be in error but no corrective action is possible, it should not be stored on the archive (except, possibly, as a 'missing value').

9. If the audit trail reveals no suggestion that the suspect value is in fact erroneous, then the value must be declared valid. The designated officer should take responsibility for its validity, and sign off the appropriate document accordingly.

RULE 2c: Data screening

The methods of Part C of this Code of Practice, and those detailed in Code of Practice CoP/OLR on 'Methods for handling outliers' should be applied to all new data (and in due course to all existing data) to check for the presence of erroneous data values.

The availability of written procedures will help reduce the number of errors reaching the archive. Nevertheless, some errors will still creep into the Archive unless the actual data values being recorded are first screened.

It is vital that screening occurs early in the life of a data value, and that a system of exception reporting should be used, so that exceptional and suspicious data items come to light before they are archived. Doing this provides the opportunity to correct errors while memories are still fresh and while any short-lived records or notes are still available. More importantly, any exceptions which are not errors, but rather reflect some genuine change, will come to light quickly, and can be speedily checked using all available statistical procedures.

Allocation of responsibility for the resolution of values highlighted in such an 'exception report' must be an integral part of the data quality procedures.

There are many techniques which may be used to highlight potential errors. Part C of this Code of Practice gives some general methods for screening data for inconsistent values; Code of Practice CoP/OLR on 'Methods for handling outliers' gives a further set of techniques for dealing with that particular type of error.

It is also wise, where practicable, to apply the same techniques to existing data - that is, data which was archived before the introduction of DQC. In most cases, because of the sheer volume of data involved, this would have to be done in a rolling programme over a long period.

RULE 2d: Documentation

Ensure that up-to-date data quality procedure manuals are in place and easily available, and that staff are fully trained in using them.

DQC procedures must not only define exactly who does exactly what, where and when; they must also be clear, brief, easily accessible, widely communicated, regularly updated, and faithfully followed. Proper document control - including the use of version numbers and the withdrawing of superseded versions - should be used, and the use of photcopied manuals should be forbidden. Adequate training in the procedures and the methods to which they refer must regularly be given.

GUIDELINES AND METHODS FOR DATA QUALITY CONTROL

PART C - TECHNIQUES FOR DATA SCREENING

C.1 ERRORS AND OUTLIERS

Many, if not most, values which arouse suspicion will be in the form of suspected outliers - that is, values which simply appear too extreme to be correct.

The study of outliers is a large and complex topic; and although the methods presented here will provide some protection against outliers, a separate Code of Practice (CoP/OLR) has been devoted to techniques which have been designed specifically to deal with them.

A good introduction to the use of statistical methods in quality control may be found in "Statistical process control - a practical guide" by John S Oakland.

C.2 THE USE OF CONTROL CHARTS IN DETECTING ERRORS

Conventional control charts

A basic tool in Quality Control is the 'Shewhart Chart' type of control chart proposed originally by Shewhart (1924). The control chart is a graphical device for presenting and helping to interpret the results obtained from repeated sampling of a process. Early applications of control charts were predominantly in the manufacturing industries (hence the word 'process'), but their use can readily be extended to the monitoring of quality in natural systems such as rivers or effluents.

In its simplest form, the control chart consists of a central horizontal line corresponding to the average or target value of the characteristic under investigation, together with upper and lower control limits beyond which only a small, stated, proportion of the points should fall when the process is running satisfactorily. These limits are derived by reference to the probability distribution which it is assumed the determinand follows when the process is satisfactory, or 'in control'.

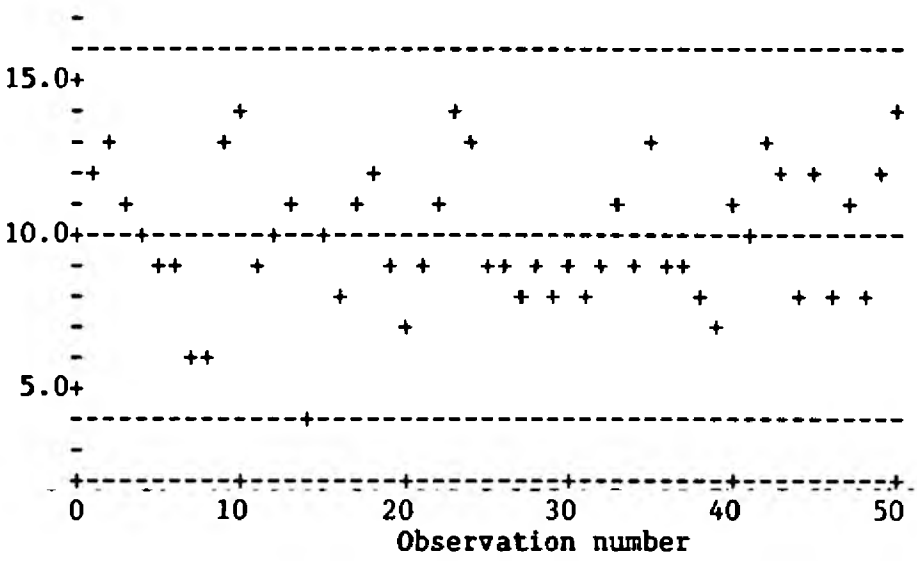
It is important to realise that, to use the chart to the full, it is not sufficient to respond only to points outside the limits. Various other features of the data can aso be used to trigger an alarm - for example, an unusually long sequence of data values all appearing on the same side of the target. In fact one can usefully apply quite a large repertoire of tests, covering a variety of departures from the assumed in-control model. (This makes it essential that the control chart is set up so that the tests can be carried out automatically using computer software.)

Full details of the construction, use and interpretation of control charts may be found in any good text on statistical process control (SPC), such as that cited above.

Example 1

Suppose that an in-control process is believed to follow a Normal distribution with mean 10 and standard deviation 2. A suitable control chart for such a process is shown in Figure C.1. The centre line shows the target mean of 10.0; the Upper and Lower control limits are set at mean plus or minus 3 standard deviations, i.e. 16.0 and 4.0. The choice of 3 standard deviations ensures that if the process remains in control, only about one point in 400 will appear above the Upper line or below the Lower line.

Figure C.1 - Typical Shewhart control chart for an 'in-control' process



Notice that the points are scattered haphazardly about the target mean, and that no point falls outside the control limits.

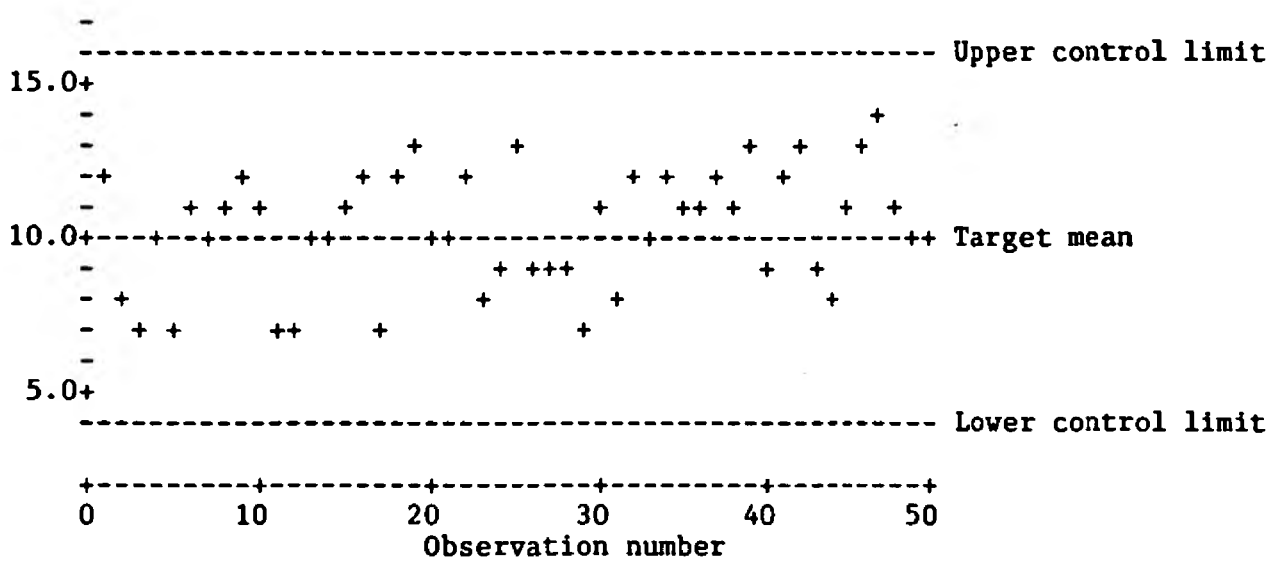
Example 2

Suppose now that the process does initially have mean 10.0 and standard deviation 2.0, but that after the 30th observation it is subjected to a persistent source of bias which causes the process mean to shift from 10.0 to 10.5. The control chart for this out-of-control process might appear as in Figure C.2.

Notice that the change in the process at observation 30 is not immediately apparent. On closer inspection, however, we can see that prior to the change, equal numbers of points (12 v. 12) lie above and below the target mean, whereas after observation 30 the majority of the points lie above the target (13 v. 4).

As we remarked earlier, this kind of assessment is awkward to do manually, and the best option is to use computer software to do the checking and significance testing.

Figure C.2 - Typical Shewhart control chart for an 'out-of-control' process



In these simple examples the underlying variability was arranged to be Normally distributed. In practice, this is unlikely to be the case, and a common problem with Shewhart charts is indeed in identifying the correct probability distribution upon which to base the positioning of the control limits.

One common solution to the problem is to plot not the individual data items but the means of small groups of observations. The advantage of doing this is that it makes the Normality assumption more tenable, thanks to the 'Central Limit Theorem' - a general statistical result which in essence states that 'means of samples from any distribution tend to follow a Normal distribution'.

Unfortunately, however, this has the effect that any exception does not come to light until the values which are to be averaged have all been collected. It also means that an erroneous data value is less likely to give rise to an exception, since its effect will be diluted by the other values in the group.

Fortunately, the distribution of determinands such as BOD, suspended solids and ammonia in final effluents can often be approximated by the log-Normal model. In particular cases where this can be shown to be a reasonable assumption, therefore, one may plot the logarithms of concentrations of these determinands on conventional Normal-based control charts.

Cusum charts

The cusum chart is an alternative type of scheme possessing several advantages over the conventional control chart - in particular its ability to detect small but consistent changes in mean level.

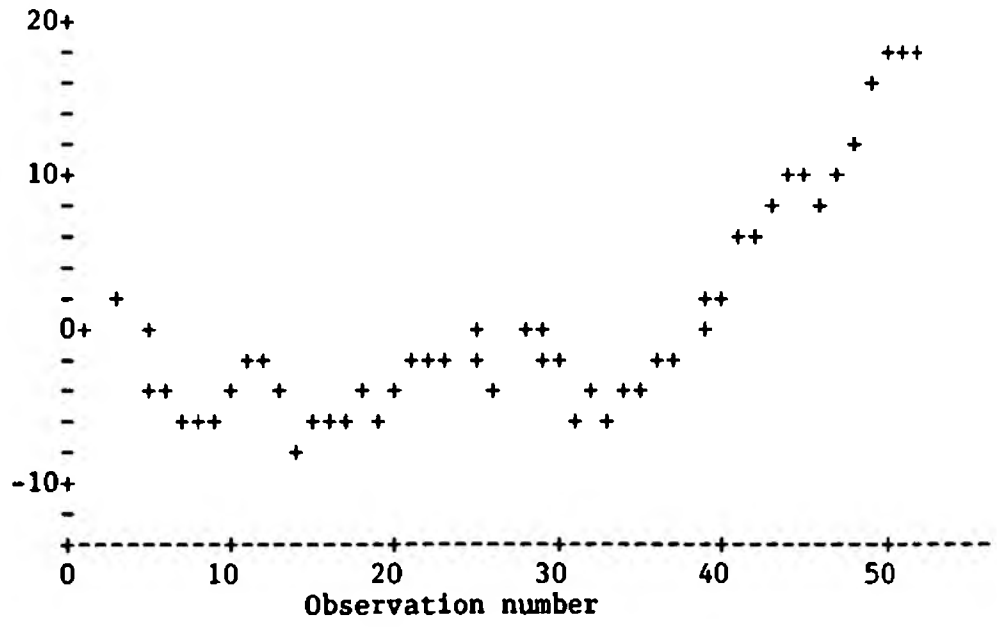
The cusum chart is constructed as follows. First, the deviations of the data values from some suitable target value (such as the overall mean) are calculated. The cusum then consists of a plot of the cumulative sum (hence the name) of those deviations against observation number - as illustrated in Figure C.3.

Changes in mean level (often difficult to spot in a time series plot because of the scatter between successive observations) are transformed by the cusum into changes in slope that are relatively much easier to pick out. Steadily rising slopes in the cusum indicate periods when the observations are on average above the cusum target; falling slopes indicate that the observations are on average below the cusum target. The steeper the slope, the greater the difference in level from the cusum target.

Full details of the construction, use and interpretation of cusum charts may be found in any good text on SPC methods.

If the underlying distribution and standard deviation of the process which generates the data are known, then the cusum chart can be used as a control chart to detect departures from a target mean. This is the basis on which the cusum example shown in Figure C.3 was constructed, using the same data as that plotted earlier in Figure C.2.

Figure C.3 - Typical cusum control chart for an 'out-of-control' process



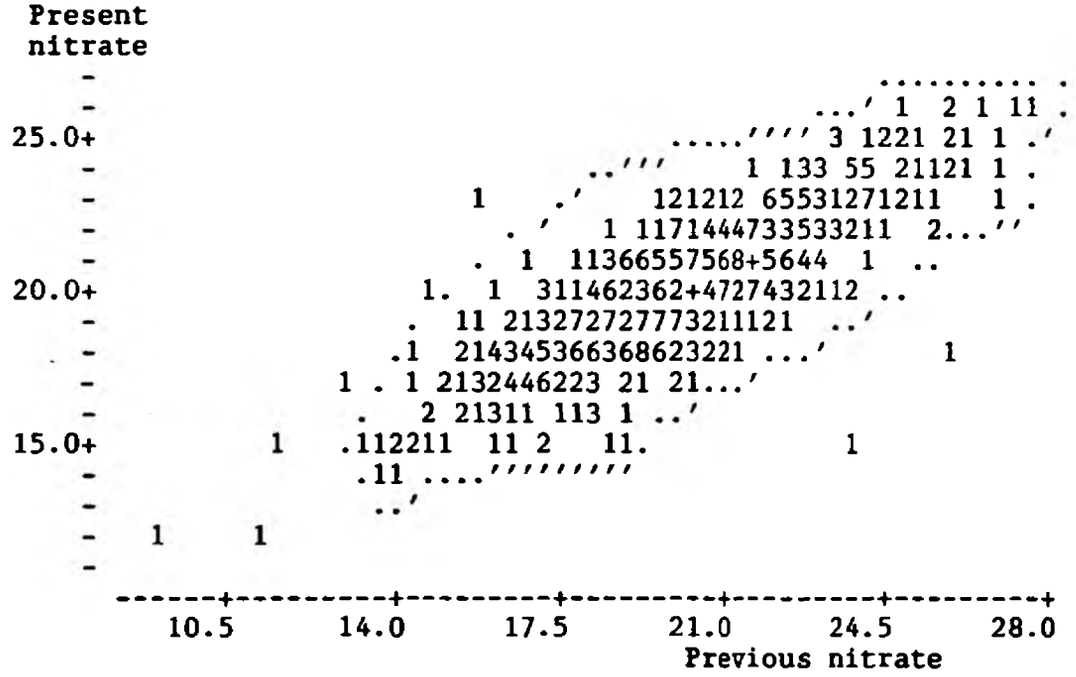
The cusum chart gives a very clear indication that there was a sudden change in mean shortly after observation 30. This rapid response to the onset of a small bias is in marked contrast to the more muted effect shown by Figure C.2, and illustrates the principal benefit of the cusum approach.

The occurrence of a change in slope does not necessarily mean that the change is statistically significant - that is, that a real change in the mean has taken place. In order to distinguish changes due to chance variation from those due to real changes in the process an appropriate significance test should be applied. Again, the details can be found in any text on cusum methods.

C.3 COMPARING AN OBSERVATION WITH PAST DATA

A useful and often revealing method of data screening is to take a long sequence of data and plot each value against the value which preceded it. An example is shown in Figure C.4 for a long series of historical data for a river sampling point.

Figure C.4 - Plot of present nitrate value against previous nitrate value



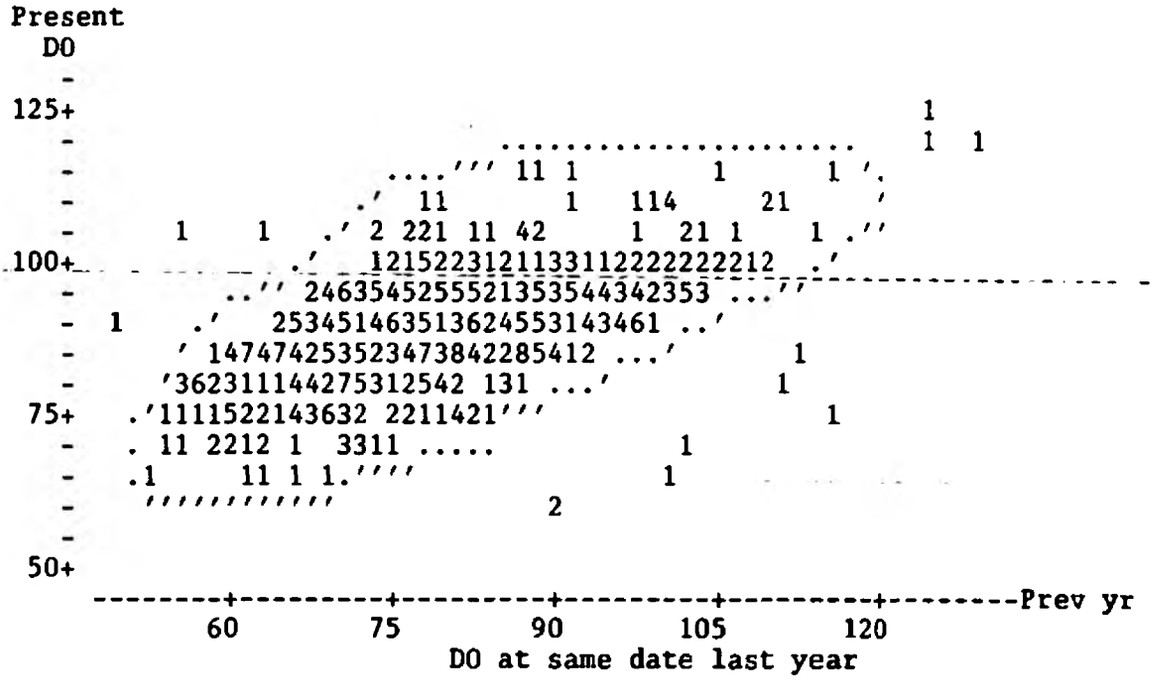
The numbers in the plot indicate how many data pairs occupy the same character position; the symbol '+' is used if this is greater than 9.

Most of the 500-odd points fall in the main 'lozenge', with only a few stragglers at some distance away. We can now use either a broad felt-tip pen or a mathematical contouring algorithm (the results will be much the same) to define a region of the plot such that any point outside it will be regarded as suspect. The dotted curve in the above plot indicates such a region.

Once defined, the region can readily be programmed into routine data-vetting software which precedes input to the archive. Alternatively it can be used as a free-standing procedure to check a data set for consistency after retrieval.

In a similar vein, if the data is such that a consistent seasonal pattern can be assumed, it could be productive to compare the latest observation with that from the same date in the previous year - as illustrated in Figure C.5.

Figure C.5 - Plot of current DO value against DO value at same date last year

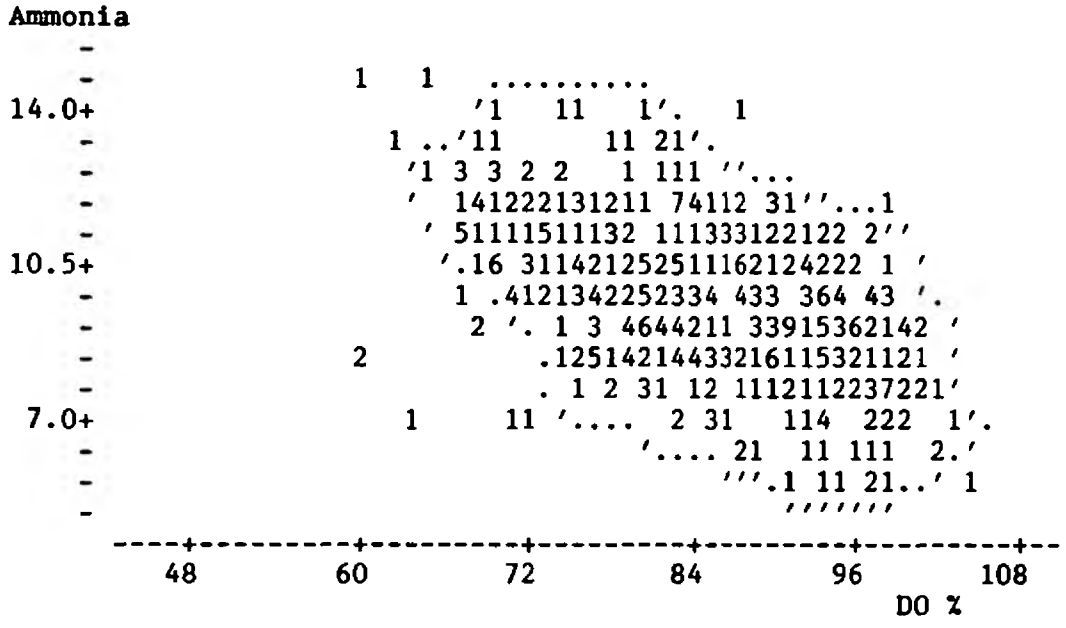


Again it is easy to construct an 'OK-region', and to use this as the basis of an error test - or, more correctly, as the basis of an exception report. Thus, the dozen or so points which lie outside the dotted boundary, had they come to light during routine data screening, should at least have raised suspicions and invoked the relevant DQC procedures to search for possible causes of exceptional data.

C.4 CROSS-DETERMINAND CHECKS

Pairs of determinands may be used in a similar way to the technique described in the previous section. Figure C.6 shows an example in which ammonia has been plotted against dissolved oxygen for a long run of river quality data.

Figure C.6 - Cross-determinand plot of ammonia against DO



The message here is the same as that of the previous section. Had the points outside the dotted region arisen in practice, they would certainly have merited investigation before being signed off and archived.