TIDEDA

for

WINDOWS®

REFERENCE MANUAL

S. M. THOMPSON

Tideda is a computer program for processing time-dependent data, particularly hydrological data. The software can be used to collect, edit, display and analyse such data. This manual is intended to teach first-time users and then become a comprehensive reference once experience has been gained.

NIWA Technical Report 88

August 2000

National Institute of Water and Atmospheric Research
P O Box 8602, Christchurch, New Zealand.

Published by NIWA
2000

Inquiries to:
Tideda, c/o NIWA,
PO Box 8602, Christchurch, New Zealand

ISSN 1174-2631
ISBN 0-478-08490-0

Citation: Thompson, S. M. 2000:
Windows Tideda reference manual.
NIWA Technical Report 88

The National Institute of Water and Atmospheric Research
is New Zealand's leading provider
of atmospheric, marine,
and freshwater science

Visit NIWA's website at http://www.niwa.cri.nz

PREFACE

The history of Tideda reflects the history of hydrological electronic data processing. The demise of chart recorders in favour of recorders with computer readable punched paper tape (1965) and magnetic media (1975) meant that field staff could not "see" the records they were responsible for collecting. Thus there was an immediate need for the computers to produce graphs. In New Zealand (N.Z.), graphs were initially print plots from a line printer (1965), then black lines on ungridded paper from a centrally located Calcomp plotter (1970), then coloured lines from local pen plotters (1980), then black lines of various thickness on gridded paper from laser printers plus colored lines on computer screens (1985) and now coloured lines on laser printers. Computers for field staff were typewriter terminals (1970), then CPM micro computers (1980) then IBM PCs or VAX computers with graphic screens (1985) and now Windows. The Windows interface is making the program much more accessible to new users who have no previous computer experience.

The Tideda program has prevailed in N.Z. because of a commitment to data already filed, and because it implements an important concept. Digital computers can only deal with discrete quantities, but this software automates the interpolation to represent continuous series that graph as wiggly lines.

In N.Z. we have collected continuous river flow records for hydroelectric design since 1905. We also have ~200 short rivers which flood in hazardous ways, and planning for mitigation of these hazards has prompted collection of many more records since 1941. Responsibility for hazard mitigation is with 14 regional authorities, and so there are many offices where Tideda is used. River habitat studies have been an important recent application of these records.

Government departments in Malaysia, Fiji, Papua New Guinea, Vietnam etc., where hydrology practice has evolved through a sequence like N.Z., have adopted Tideda. Private companies in Malaysia, Australia, Indonesia as well as N.Z. also use it.

M W Rodgers developed the first Windows version in 1995, and since then T.A. Hill, assisted by G.A. Halliburton, has been responsible for its maintenance and enhancement. Complementary applications continue to be developed and some of these applications may become new processes within the Tideda program. Others remain as independent programs that make use of Tideda files – notably Flosys for telemetry and Ricoda for river cross-sections.

S M Thompson, August 1999

Prefaces to previous Tideda reference manuals

1974

TIDEDA is the latest in a series of programs written in the Ministry of Works and Development for processing time-dependent data in general, and river flow data in particular. The following concepts in TIDEDA have been derived from these earlier programs. In 1960, H.F. Kennedy arranged series of values at unequally spaced times to aid the digitisation of data from charts. Subsequently, R.K. Howard introduced magnetic tape storage, arranged for stage-to-discharge ratings to be stored with the stage values to facilitate the editing process, and introduced complex data comprising the several measurements needed to derive flow at a power station. In 1966, P.J. Thompson represented time by the number of seconds since 1940, and used integer arithmetic to maintain precision.

We realised in 1969 that the CSMP system, produced by IBM in 1966, was an excellent aid in the implementation of mathematical models which exploit river flow measurements, and other measurements which are represented as time series. This application has been in view throughout the development of TIDEDA.

Since 1970, we have worked together. We would offer our thanks to the many other people who have provided essential parts of the program, particularly P.H. Mansfield, R.P. Ibbitt and B.C. Grant, who have contributed more than a year of programming effort, and to the many patient and adaptable users who comments have been greatly appreciated.

S.M. Thompson and G.R. Wrigley, September 1974

1985

TIDEDA is software for processing time-dependent data, particularly hydrological data, and was written by the New Zealand Ministry of Works and Development to run on a mainframe computer. Since 1970, a national hydrological archive containing 16,000 station-years of data has been complied under R.P. Ibbitt's supervision, and he has had a formative influence. Over the same period the most demanding applications have been for hydroelectric investigations in collaboration with I.G. Jowett and D. Ferguson who have also had a formative influence. The origins of this mainframe software are outlined in the 1974 preface, which is repeated on the following page.

Micro-TIDEDA is a development from that software and runs on a microcomputer with floppy disks. It is used for the same purposes as the mainframe system, both for data preparation prior to archiving and for applications. It is also for initial entry of data destined for the mainframe archive.

We thank the following people who have contributed more than a years effort on micro-TIDEDA software since its inception in 1981. S.C. Close of MWD, Christchurch demonstrated an early version that permitted direct access to storage in place of the earlier sequential methods. P.N. Pallister, a consultant at Micro-Controls & Systems Ltd, Lower Hutt produced the first CP/M version, and continued on the development of a simulation capability. A.G. Wood, now of Wormald Vigiliant, Christchurch, developed a comprehensive set of interfaces for data entry, processing and display.

S.M. Thompson and M.W. Rodgers, June 1985

1991

Computer processing of time-dependent data in general, and river flow data in particular, is done throughout New Zealand with the Tideda program. The current version evolved as described in prefaces to the 1974 and 1985 manuals which are repeated below. The principal improvements since 1985 are:

- use of graphics on visual display units and laser printers; and
- use of large (gigabyte) archives from a personal computer.

We thank the following people who have contributed substantially to maintenance and development since 1985. T.A. Hill has been responsible for user support. G. A. Halliburton wrote the digitising software. R.D. Henderson and R.A. Woods have willingly and effectively tested new features before release, and C. Holmes has edited this manual.

M.W. Rodgers and S.M. Thompson, March 1991

CONTENTS

CHAPTER 1 INTRODUCTION

What Tideda is for

The Tideda computer program deals with data that are naturally recorded as a time sequence. Examples are river flow, rain fall, air temperature and electric power generation. The lack of an elaborate data structure in such sequences makes the processing methods simple. It is the large quantity of data, the unusual time units (year, day, hour, etc.) and requirements for interpolation that justify the use of a specially designed computer program.

Although Tideda has been used primarily for hydrology and meteorology, and the examples in this manual reflect that, the terminology used is more general and suitable for use in other disciplines.

The program accepts data from a variety of automatic recording instruments. Once entered into the system, data are stored in a binary format designed to facilitate checking, correcting and analysis.

What Tideda does

Data entry from various devices

We can capture records from a keyboard, a chart digitiser in a comprehensive range of chart formats, a wide range of computer readable recorder and disk file formats, and directly by telemetry. There are processes within Tideda or associated applications for all these. The main associated applications are for telemetry including radio telemetry from field recorders and Internet communications with other computers.

Ratings

There is provision for storage with the series, and routine application, of non-linear transformations to data. In relation to hydrological data, this feature is used for the representation of river stage-to-discharge rating curves that vary over time as the channel changes.

Gaps

Temporary failures of field-measuring equipment cause gaps between periods of continuous series. They are flagged in the corresponding filed sequence of discrete values so that the normal interpolation rule is not applied. Several methods are available for dealing with gaps in graphs, tables and calculations. For example synthetic data that are made to fill a gap can be kept in a separate file so they are not confused with measurements, and that file can be linked using an "extended directory" so that we can read a seamless, complete record.

Data management and manipulation

We can check, then correct or delete data, rearrange the content of files to suit new applications or to more efficiently use storage space.

Graphical and tabular displays

We can produce high quality graphs, and a choice of processes treats different kinds of data in appropriate ways.

We can tabulate means, extremes and totals computed hourly, daily, weekly, monthly or annually. These text tables are not stored, but are prepared only when required from binary files. They are used directly in documents and indirectly after transfer to other computer applications.

Macro languages

Tideda has two macro languages:

- a "simulation" language (SIM) for algebraic calculations;
- a "Tideda script file " language (TSF) for recording a sequence of processes so that a single instruction can rerun the whole sequence.

Linking Tideda files to other programs

A toolbox of software can be provided in a "dynamic linked library" for directly accessing the contents of Tideda binary files from other Windows applications. For example Visual Basic macros in the Microsoft Excel spreadsheet program can make calls to this library which read and write Tideda files.

How this book is organised

This manual is for people who know what they want to do and are seeking advice about how. Chapter 2 in tutorial style, leads us through a sequence of related examples, designed for a first time user. Chapter 13 is also in tutorial style and is for experienced users. You might find an example in a tutorial that matches what you want to do. The rest of this manual is a comprehensive reference and index.

A banner at the top of the computer window contains buttons which open menus, and each menu lists a set processes. Names of menus and processes are always written in capitals, e.g., "the **FILE** menu's process **OPEN INPUT FILE**".

Each process offers options. These are on dialog sheets where their purpose is indicated in the way they are grouped with other options and of course by their label. When the manual offers additional explanation about an option, the option name appears with the label capitalised and **Bold**.

Although its main applications have been hydrological, Tideda avoids hydrological terminology and uses adjectives for the kind of data such as: Increment for rain, Instant for stage and Histogram for values that are constant over a time step and for averages. The noun distribution is used rather than the phrase "flow duration curve". This vocabulary focuses attention on the methods in an abstract way rather than on a particular application, with the advantage that once the small vocabulary is mastered we can deduce what will happen without reference to this manual. It is possible to process in the same way all data that have the same abstract requirements, e.g., gate openings and monthly averages are both Histogram data in that the value is constant over the time interval between the discrete filed values.

The Windows version of Tideda retains, as options for experienced users, both the command and menu user interfaces of previous versions.

PRIOR EXPERIENCE

Tideda's Windows interface has many menus and on a first encounter the purpose of some of them may not be clear. Fortunately there are Windows conventions that assist interpretation. Thus the first three menus are **FILE**, **EDIT** and **VIEW** and the last is **WINDOW**. These menus contain what an experienced Windows user expects.

- If you are familiar with Windows but have no experience of Tideda, then follow the tutorial in chapter 2 using data from the small Tideda file supplied with the Tideda software. This should give you some idea of what Tideda is about. The last chapter ORGANISATION OF DATA has descriptions of the kinds of data that can be stored with Tideda and the organisation of the data files, and definitions of the capitalised words in this manual such as Site, Kind, Histogram, Increment, Attribute, etc., which in Tideda have precise meanings.
- If you are unfamiliar with Windows but an experienced Tideda user, the tutorial in chapter 2 explains how to use the Windows interface to use data from an existing Tideda file. You could use this as your guide to Windows Tideda while processing some of your own data.
- Those who are unfamiliar with both Windows and Tideda, should first look elsewhere for some help with learning to use a Windows interface.

AGENT

This manual mentions items available from your Tideda agent. Enquire at the following address to find out who this is:

Tideda, c/o NIWA, P O Box 8602, Christchurch, New Zealand

telephone +64 3 348 8987 fax +64 3 348 5548

email t.hill@niwa.cri.nz or g.halliburton@niwa.cri.nz

ACKNOWLEDGEMENTS

Data used in some of the examples is provided by Environment Canterbury. Other data was collected by NIWA under the FRST funded *Information on New Zealand's Freshwaters* programme.

C. Kilroy and T. Hill assisted with the formatting and editing of this manual.

CHAPTER 2 BEGINNERS' TUTORIAL

Starting Tideda for the first time

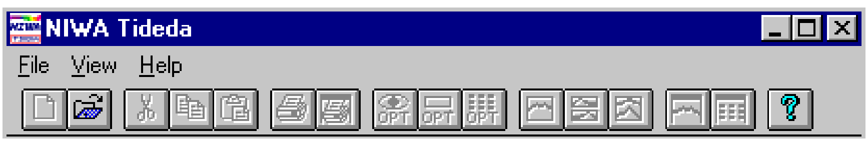
PREVIOUS SESSION SETTINGS

Tideda stores options and settings used in the previous session in two files. Options that are common to all Tideda versions are in a file called Timouse.dat. It is located in the Working folder specified in the **PREFERENCES** dialog described on page 2.2. This is not a text file and can not be directly viewed or edited by the user. This is the same as the Timouse.dat file used by the DOS version of Tideda. The second file is named Td32.ini and is located in the Windows or Winnt directory. This is a text file and contains settings specific to the Windows version.

OPEN THE PROGRAM

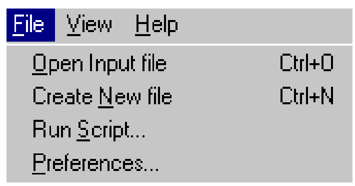
To run the Tideda program double click on this icon on our Desktop >

This will open the program with the following banner :



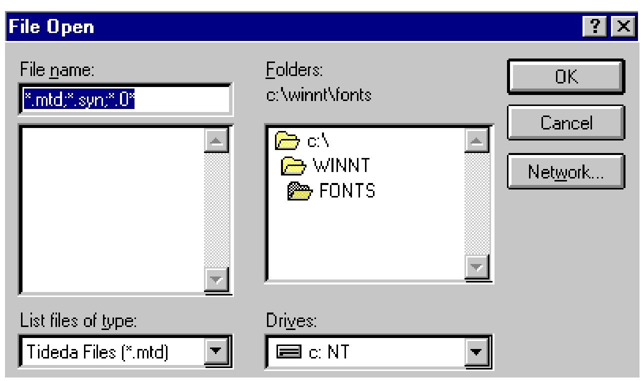
OPEN A FILE WINDOW

Click **FILE** and this drop down menu appears >

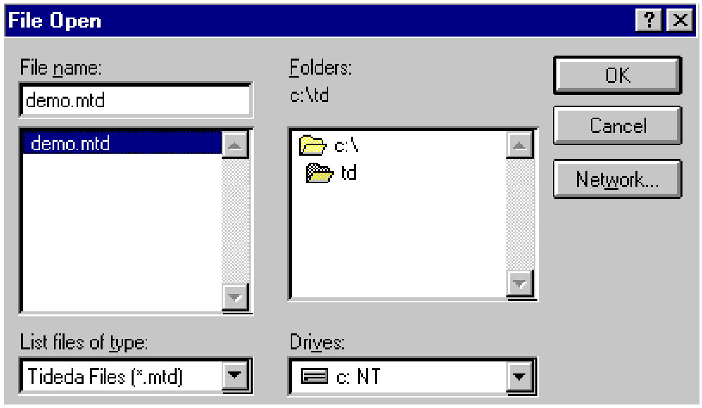


The easiest way to start using Tideda is to open an existing file. For this reason two files are supplied with the program, Demo.mtd and Demo.att. If we put them in a folder called c:\td we will be able to reproduce the illustrations in this tutorial.

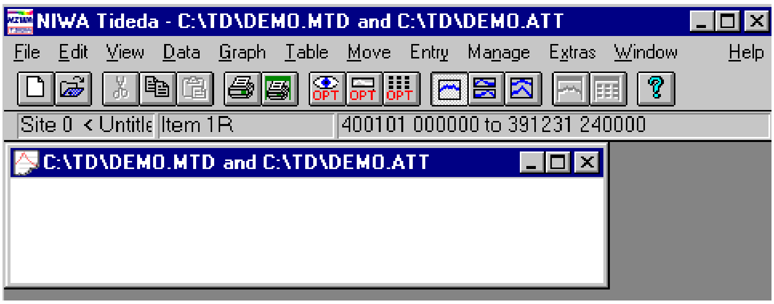
Click **Open Input file**, to get this dialog >



In the **Folders** box: double click on the root folder name e.g., **c:**, double click on the particular folder name required, e.g., **td**, then click on the file name required, e.g., **demo.mtd**, to get the dialog to look like this >



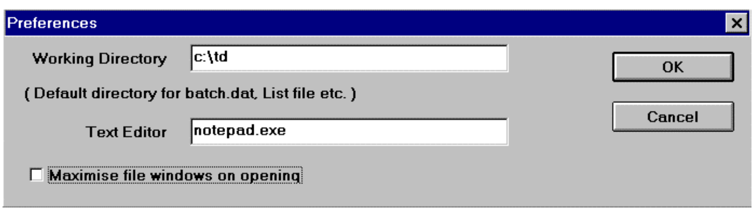
Click **OK** and a File window will open under the program banner >



The second blue bar shows the name of the input and Attribute files attached to the File window. Results from those files will be presented in this File window and can be kept there when a second File window is opened, and then returned to later in the same session. However in this tutorial we will use only one File window.

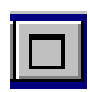
PREFERENCES

Click **FILE**, click **PREFERENCES** to get this dialog >



To reproduce the illustrations in this tutorial we set the preferences as shown above, and click **OK**. We have already suggested we create c:\td for the demo files.

It may be advisable to **Maximise file windows on opening**, especially if we are using a small screen. However to better see what we are doing, we will not make this automatic, but use this button at top right of the window >



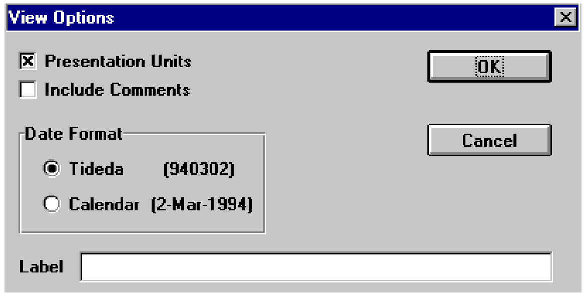
VIEW

Click **VIEW**, click **OPTIONS** to get the this dialog >

We select **Presentation units** so that the scaling factors in our Attribute file are applied to convert data from File units.

We select **Date Format Tideda**, just a personal preference.

Click **OK**.



Exporting a series (with gaps) to another program

Our next example is specific to some data in the Demo.mtd file. We export 8 days of Whataroa River flow data as text to an EXCEL spreadsheet. The particular 8 days we choose include a gap in the record so that we see what happens at gaps. Thus we illustrate:

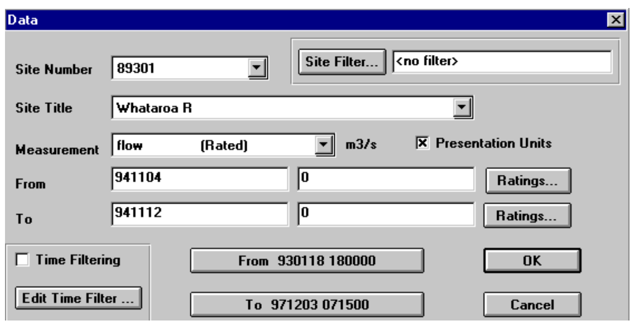
- How a particular series is selected.
- How gaps can be marked.
- How to transfer data from TIDEDA to an EXCEL spreadsheet.

Some gap marking options are a recent enhancement to TIDEDA. We think their importance for maintaining accuracy and credibility warrants the emphasis we give them here.

DATA SELECTION

We will examine 8 days from the Whataroa River record in the input file Demo.mtd.

Click **DATA** in the program banner,
to get this dialog >
Set **Site Title**: Whataroa R.,
Measurement:
flow (Rated)
From: 941104 0
To: 941112 0.
Click **OK**.



We can select data we wish to process before we select a process, as we have just done, by clicking the **DATA** button in the program banner. Conversely we can select a process before the data then click the **DATA** button in the process dialog. In this manual when describing a particular process we generally do the latter to emphasise that process.

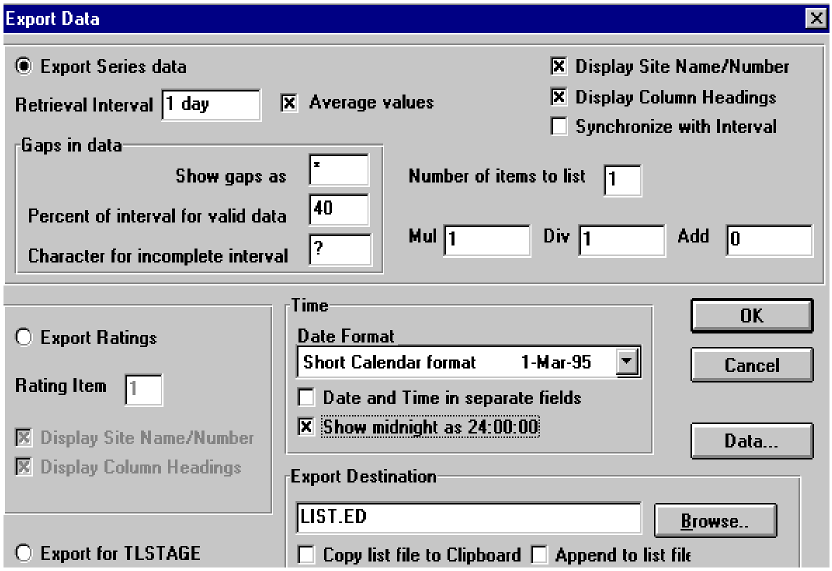
Just for our information at this stage we can use the **MANAGEMENT** menu's **GAP** dialog to get a report on the gaps in the record and it includes the following:

```
Gap from  941108 71500 to 941110 121500 of 2.21 days
```

EXPORT FROM TIDEDA

The **TABLE** menu's process **EXPORT** lists a time series at equal time steps to a text file, one time per line.

Click **TABLE** in the program banner, click **EXPORT,** then set the dialog thus >



In the **Gaps in data** box on the dialog, we set **Percent of interval for valid data** = 40 to cause the program to interpolate across gaps in the record to complete any intervals which are at least 40% measured. We have also specified that a * will be reported for intervals <40% measured, and the values reported for <100% measured will be followed by ?

Click **OK** in the **EXPORT** dialog and the File window will then contain:

```
~~~ Export data to text file ~~~
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R
From  941103 240000 to  941111 240000
Output to LIST.ED
```

To see the contents of LIST.ED, one way is to go to the **EDIT** menu, click **LIST FILE**, and click the **EDIT** button. The editor will open a window containing:

```
Site 89301 Whataroa R
Date,flow m3/s
4-Nov-94 24:00:00,82.5
5-Nov-94 24:00:00,288.0
6-Nov-94 24:00:00,1169.7
7-Nov-94 24:00:00,1216.4
8-Nov-94 24:00:00,*
9-Nov-94 24:00:00,*
10-Nov-94 24:00:00,350.8?
11-Nov-94 24:00:00,187.6
```

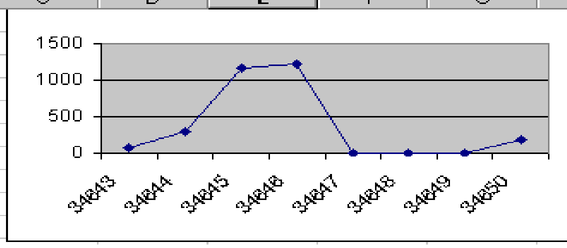
This list reports * for 8 November because the record from 0 to 715 is <40% of that day, but reports 350.8? for 10 November because the record from 1215 to 2400 is >40% of that day.

EXPORT TO EXCEL

To enter the text from LIST.ED into an EXCEL spreadsheet:

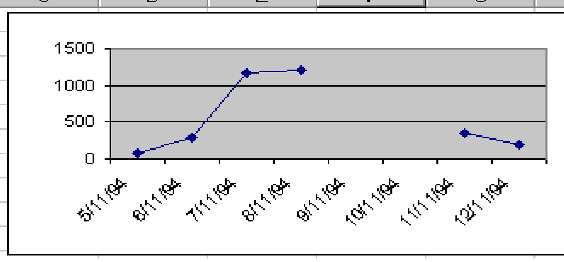
- Start the EXCEL program and click **FILE**, click **Open**
- Then in the dialog select Files of type: "All files (*.*)", select "List.ed",
- Click **Open**, then select the following options in the Text Import Wizard,
- in Step 1 Delimited, in Step 2 Delimiters: Comma, in Step 3 Date: DMY
- Click Finish to close the wizard
- Use the Plot Wizard to make a plot

	A	B
1	Site 89301 Whataroa R	
2	Date	flow m3/s
3	34643	82.2
4	34644	287.8
5	34645	1169.2
6	34646	1216.4
7	34647	*
8	34648	*
9	34649	350.7?
10	34650	188



- Delete the * and ? gap indicators to improve the appearance of the plot
- Highlight the Date column numbers,
- Click Format, click Cells, click Custom, type dd-mmm-yy hh <enter>

	A	B
1	Site 89301 Whataroa R	
2	Date	flow m3/s
3	05-Nov-94 00	82.2
4	06-Nov-94 00	287.8
5	07-Nov-94 00	1169.2
6	08-Nov-94 00	1216.4
7	09-Nov-94 00	
8	10-Nov-94 00	
9	11-Nov-94 00	350.7
10	12-Nov-94 00	188

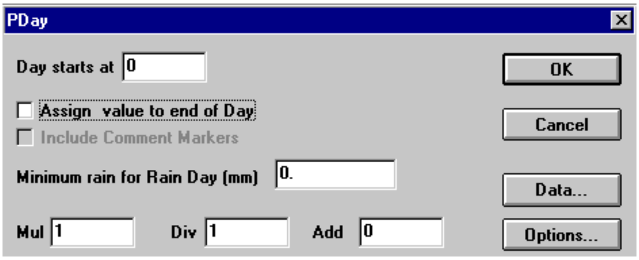


The values are averages up to a time of day. Recall that in the Export Data dialog we selected **Show midnight as 24:00:00** so that the TIDEDA dates are the day actually measured. However EXCEL always shows midnight as 00 hours so that its dates are all a day later. Watch out!

Calendar tabulations

There are long standing conventions for calendar tabulations of time series, and the program offers several, including the following.

Click **TABLE**,
click **CALENDAR**
click **DAILY**
set this dialog >



Click **DATA**
to get the Data dialog,
set **Site Title**: Whataroa R
Measurement: flow (Rated); **From**: 94 0, **To**: 95 0,
click **OK** to exit the **DATA** dialog, click **OK** again to print:

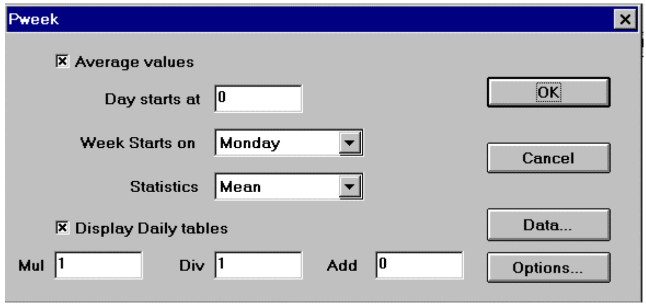
(Note that 94 is a synonym for 940101, etc.)

```
Source is C:\TD\DEMO.MTD
24 hour periods beginning at midnight each day.
Daily means   Year 1994   site 89301 Whataroa R  flow m3/s
```

Day	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	
1	187	150	202	37	50	90	68	40	?	38	87	160	
2	183	141	142	36	88	78	57	81	55	36	78	148	
3	169	130	93	101	244	70	50	150	45	35	74	126	
4	260	126	83	55	118	64	47	99	42	35	82	121	
5	582	120	77	43	76	59	219	69	40	37	288	118	
6	222	118	272	39	63	55	96	54	64	37	1170	111	
7	293	116	219	37	56	52	82	48	386	36	1216	113	
8	1201	117	107	61	52	54	116	96	109	38	?	122	
9	2283	161	80	193	51	50	62	64	72	71	?	179	
10	566	149	69	99	48	267	82	48	57	75	?	157	
11	275	132	64	83	46	221	101	43	52	80	188	120	
12	204	122	59	143	44	108	61	40	50	106	356	107	
13	161	119	57	444	41	84	52	83	45	62	238	99	
14	142	122	145	194	39	70	48	361	41	54	197	105	
15	218	118	497	118	40	62	44	110	46	45	356	112	
16	165	115	686	92	46	63	82	69	72	40	256	115	
17	126	112	274	79	38	56	227	56	136	38	168	107	
18	244	108	187	72	39	69	81	62	88	64	384	105	
19	336	190	224	408	349	138	62	46	165	55	168	141	
20	206	255	145	352	221	78	53	41	364	46	127	158	
21	692	140	98	186	116	67	61	38	160	41	611	117	
22	1758	186	74	116	82	59	?	36	105	39	459	104	
23	836	146	62	89	119	54	?	36	78	39	238	89	
24	585	110	54	74	132	97	?	?	65	41	166	76	
25	329	102	47	71	101	603	?	?	57	44	131	75	
26	268	93	45	64	110	209	?	?	52	48	115	68	
27	239	91	46	58	236	118	?	?	48	52	108	69	
28	220	90	47	55	278	86	46	?	44	53	108	78	
29	208		45	52	186	71	52	?	41	109	168	167	
30	185		40	50	220	62	44	?	39	148	147	520	
31	165		38		113		41	?		101		214	
Min	126	90	38	36	38	50	41	36	39	35	74	68	35
Mean	436	131	138	117	111	107	77	77	90	56	285	132	148
Max	2283	255	686	444	349	603	227	361	386	148	1216	520	2283

Click **TABLE**,
click **CALENDAR**
click **WEEKLY**
set this dialog >

Click **DATA**
to get the Data dialog,
set **Site Title**: Whataroa R
Measurement: flow (Rated)
From: 94 0, **To**: 98 0,
click **OK** to exit the **DATA**
dialog, click **OK** again to print.



A daily table is laid out in 13 week seasons. Every 5 or 6 years there is a 14 week season.

Autumn 1994

		Mar.		(138)			Apr.		(117)			May.		(111)		
Mon.	90		219	145	98	47		55	83	72	71		88	51	46	119
Tue.		202	107	497	74	45		43	143	408	64		244	48	38	132
Wed.		142	80	686	62	40		39	444	352	58		118	46	39	101
Thu.		93	69	274	54	38		37	194	186	55		76	44	349	110
Fri.		83	64	187	47		37	61	118	116	52		63	41	221	236
Sat.		77	59	224	45		36	193	92	89	50		56	39	116	278
Sun.		272	57	145	46		101	99	79	74		50	52	40	82	186
Week		137	94	308	61		49	75	165	185		57	100	44	128	166

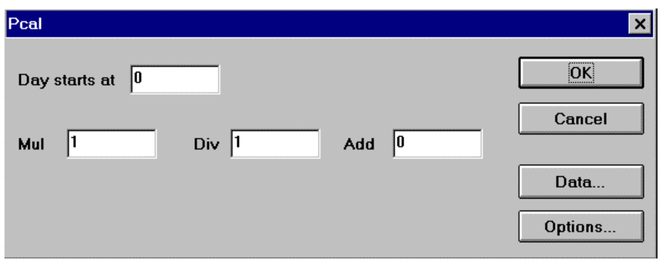
A seasonal table is laid out a year to a line and the seasons are ranked.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	295?	5	?	?	?	?	?	?	295?	5
1994/5	129?	3	121	1	91?	4	138?	2	164	2
1995/6	158	4	185	4	48	2	159	3	241	4
1996/7	123	2	162	3	41	1	164	4	125	1
1997/8	103?	1	128	2	69	3	99	1	225?	3
Period	162?		149?		62?		140?		210?	

Click **TABLE**,
click **CALENDAR**
click **MONTHLY**
set this dialog >

Use the same data as above.

Click **OK** to print:

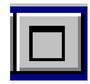


Source is C:\TD\DEMO.MTD
Monthly means 1994 to 1997 site 89301 Whataroa R flow m3/s

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Mean
1994	436	131	138	117	111	107	?	?	?	56	?	132	154?
1995	204	150	237	201	114	45	37	63	228	134	120	400	161
1996	160	147	132	244	110	56	28	33	95	284	135	132	130
1997	83	180	89	169	103	51	54	97	47	104	183	?	105?
Min.	83	131	89	117	103	45	28	33	47	56	120	132	130
Mean	221	152	149	183	109	65	40	64	123	145	146	222	145
Max.	436	180	237	244	114	107	54	97	228	284	183	400	161

GRAPH AND ZOOM

We now select data from a graph. Tabular formats impose an arbitrary time partition on series that really represent continuous measurements that are better presented as graphed lines. There are several ways to graph series. When graphing we need a sufficiently large File window to resolve graph details and have wide enough margins for labels. To get the maximum window click this button at top right in both program banner and File window >.

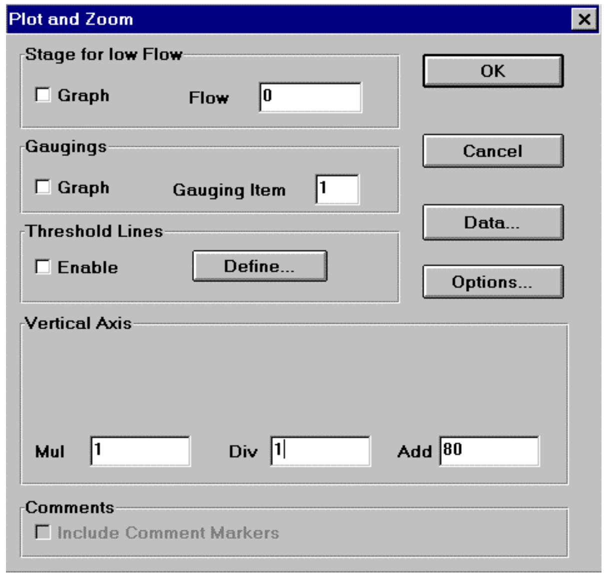


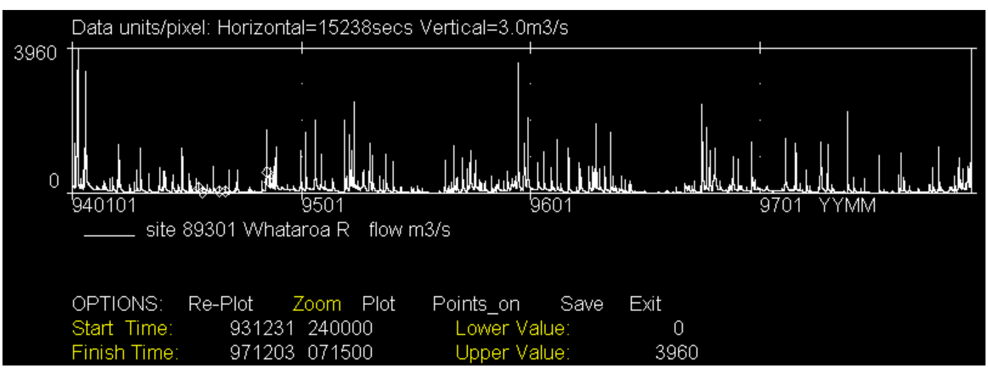
ZOOMING

Click **GRAPH**,
click **GRAPH AND ZOOM**,
to get this dialog >

Click **DATA**, and in the **DATA** dialog set
Site Title: Whataroa R.,
Measurement: flow (Rated);
click the grey boxes
"From 940101 0"
"To 971203 071500"
which sets **From**: and **To**:
click **OK** to exit the **DATA** dialog.

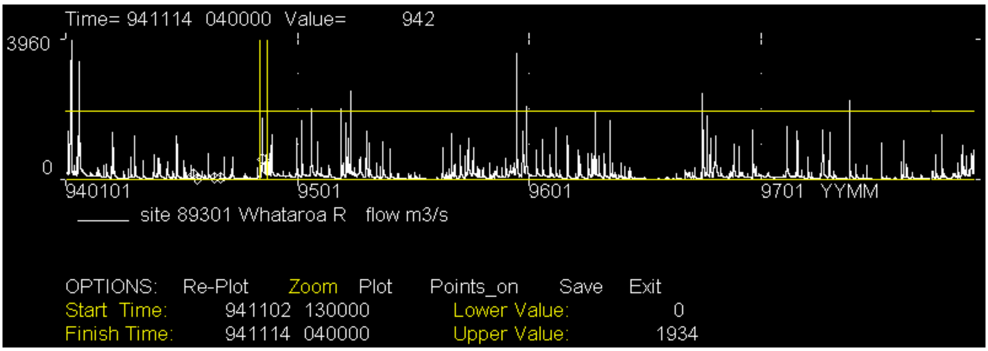
Click **OK** to get this graph of 3 years at a very small scale in the file graph window.



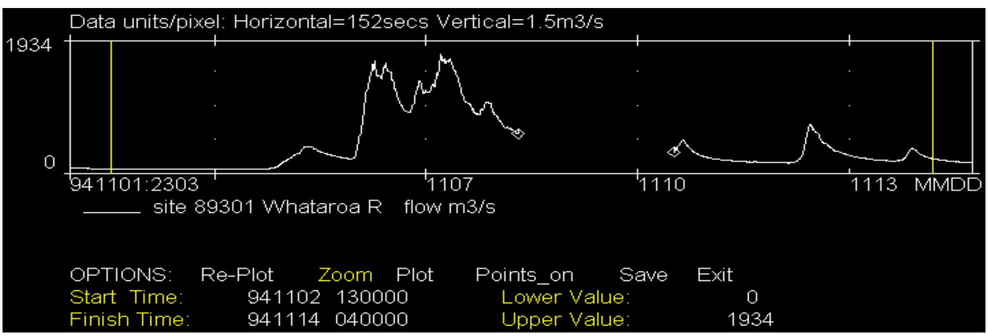


We now zoom in to select the same period as we have already plotted in an EXCEL spreadsheet. At this small scale, the zoom steps by relatively large increments. We select a range as small as possible that includes all the range we wish to select.

- Click **Zoom** which makes the box outline change colour,
- click next to the left edge in the middle of the highlighted box, to make it red,
- move the cursor across until the "Time" above the box is just before 941104,
- click and the left of the box moves to that time;
- click next to the top edge in the middle of the new highlighted box, which makes it red,
- move the cursor down to around the 2000 level
- click and the top of the box moves to that value;
- click next to the right edge in the middle of the highlighted box, to make it red,
- move the cursor across until the "Time" above the box is just after 941112
- click and the right of the box moves there; and the graph looks like this:



- RIGHT click to display the zoom menu and click the **Plot Zoom** option to get:



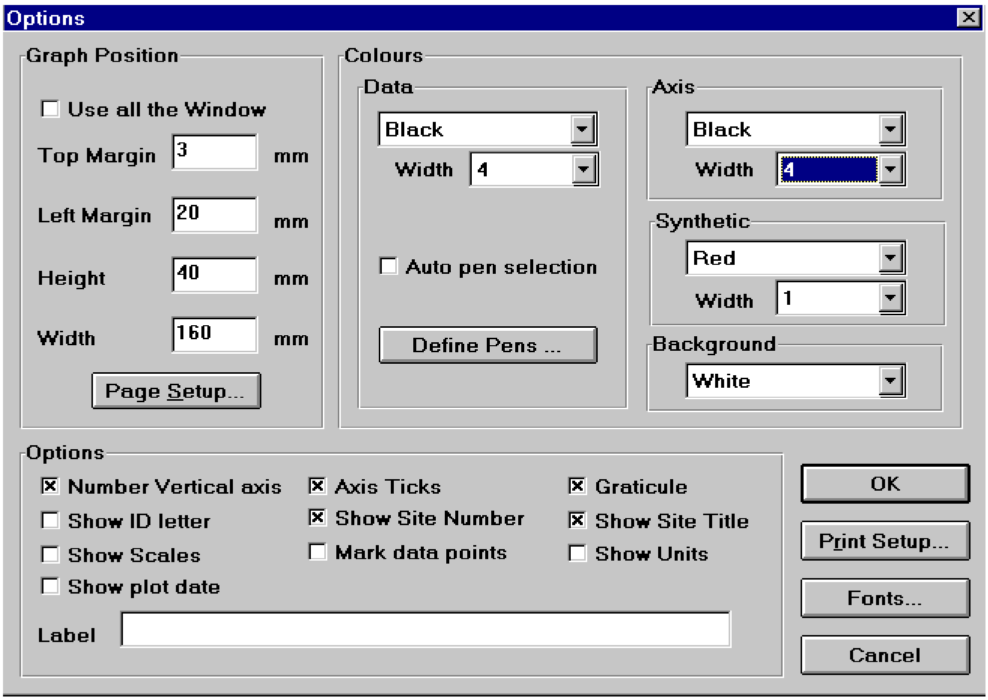
Click **Save zoom**, then click **Exit** to exit the **GRAPH AND ZOOM** process. You must explicitly **Exit** from this process before continuing with another process.

Graph options

Process **GRAPH AND ZOOM** that we illustrated above will always **Use all the Window**, and there is no choice. However when using other Graph processes this is an option that we can deselect to specify a required size or to avoid any risk that an inadvertent resizing of the window would spoil an **Over Plot**.

We now customise this and several other graph options.

Click **GRAPH**, click **OPTIONS**, to get this dialog >



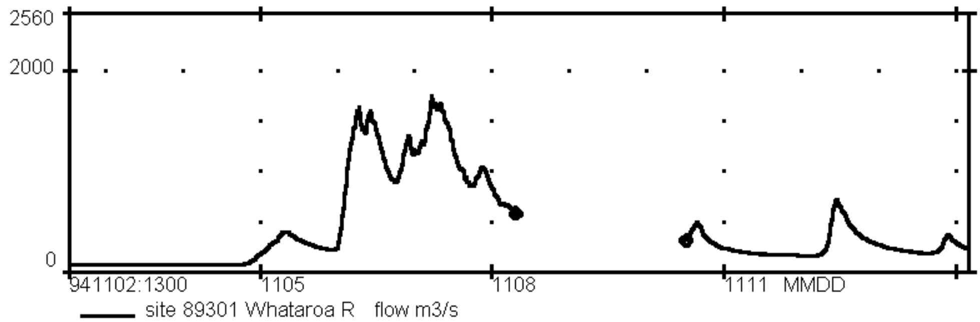
We have just used process **GRAPH AND ZOOM** which calculated **Graph Position** values to use all of our File window. However for the following examples we deselect **Use all the Window**, increase the **Height** and decrease the **Width** slightly, to round numbers, to get the values shown above, set the other options as shown, and click **OK**.

GRAPH OVER TIME

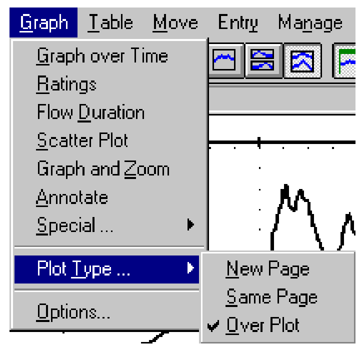
We now compare time series by plotting them together. Recall that we clicked **Save** before exiting from the **GRAPH AND ZOOM** process which has saved the parameters defining the plot so that it can be reproduced. To use these saved parameters:

1. click the menu **GRAPH** in the program banner and a list appears,

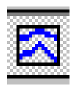
2. click the process **GRAPH OVER TIME** in the list and a dialog appears,
3. click **OK** in the dialog and the following graph appears.



4. Click the menu button **GRAPH** in the program banner again and the same list appears,
5. click the sub-menu button **PLOT TYPE** and a sub list appears like this,
6. click **Over Plot** in the sub list.



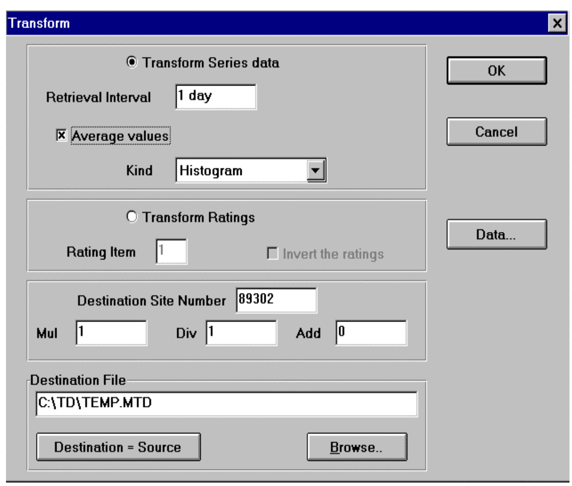
The above steps 4-6 can be short cut by clicking the **Over Plot** button in the program banner which looks like this >.



New Page and **Same Page** have shortcut buttons immediately left of the **Over Plot** button.

We are now ready to plot a second series on our graph but we must calculate it first. The calculation will be described in a similar way to steps 1-6 immediately above, but with fewer words.

Click **MOVE**,
click **TRANSFORM**,
to get this dialog >,
but don't set values in it yet.

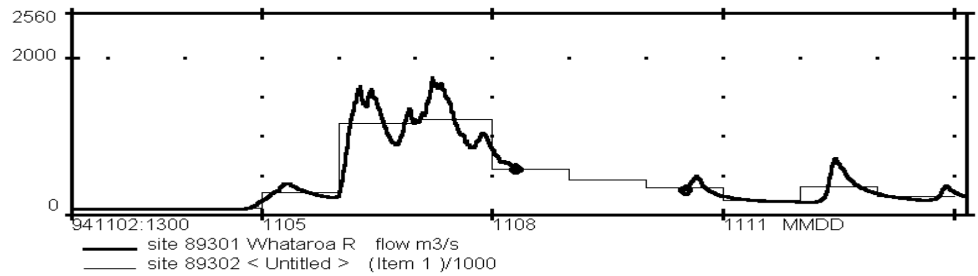


Click **DATA**, set
Site Title Whataroa R
Measurement flow (Rated)
From 941102 0 **To** 941115 0
click **OK** to exit the **DATA** dialog.

Note that in this dialog >
the **Kind** and **Destination Site Number** have changed to match those just set using the **DATA** dialog, and we now reset them to

Histogram and 89302 then click **OK** to do it.

Click **DATA**, set **Site** 89302, click **OK.**
Click **GRAPH**, click **GRAPH OVER TIME**, set **Vertical axis Div** =1000,



click **OPTIONS**, set **Colours Data Width** = 1, click **OK**, click **OK** again, to graph:

Note that our graph remained in the File graph window while we used process **TRANSFORM**. We have also interpolated data across the gap in a way that seems to be a reasonable assumption in this particular case. The fine line Histogram of daily means is a more accurate graph of daily means than the sloping lines that joined the data points in the EXCEL plot, because the area under this line is accurate. The bold line plot of the continuous record is even better, because it portrays more information and has no arbitrary time partition.

Graphing increment data

Series, such as rain series, which are filed as Increments are plotted as Histograms and a **Retrieval Interval** is required. Thus the advantages of a continuous graph are not available for these series. This last statement is not strictly true as explained on page 13.10.

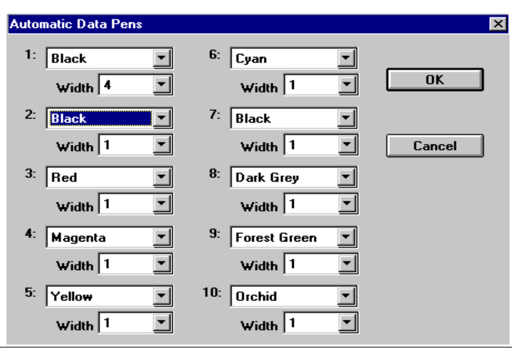
We will plot together: (a) part of the Hermitage rain record, and (b) the corresponding part of the Whataroa River flow record; to check how many of the floods in the Whataroa River are associated with storms that reach across the alpine divide as far as the Hermitage. We start by selecting the rain series: click **DATA**, and in the Data dialog: set **Site Title**: Hermitage, **Measurement** rain, **From**: 960120 0 **To**: 960209 0, and click **OK**.

During this example we illustrate several widely applicable features of the graph processes.

AUTOMATIC PEN SELECTION

In the previous graph we made the lines have different **Width**s so they could be distinguished in a black and white plot. On a colour screen we use different colours. To facilitate this

Click **GRAPH**, click **OPTIONS**,
click **Define Pens** to get this dialog >
set the pens as shown here, or at least first two for the following illustrations.

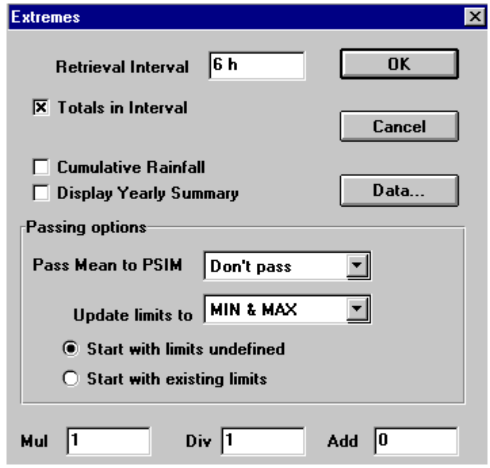


Select **Auto pen selection**

Click **OK** to exit this dialog.
Click **OK** to exit the Options dialog

Y-AXIS MIN AND MAX FROM THE DATA

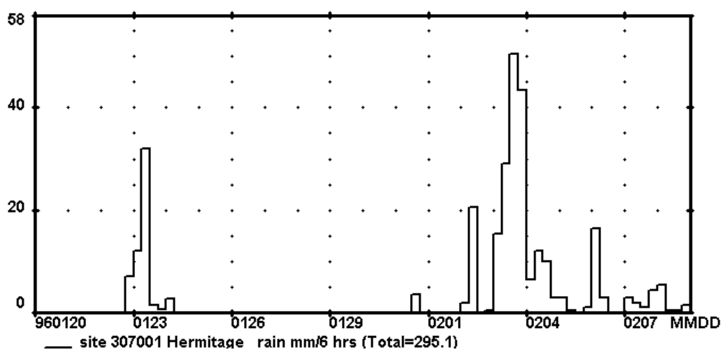
Click **TABLE**,
click **QUICK EXTREMES**,
set the dialog thus>,
and in particular with
Retrieval Interval = 6 h, click **OK**.



The following appears in the File text window.

```
Source is C:\TD\DEMO.MTD
Site 307001 Hermitage
From 960119 240000 to 960208 240000
Interval =       21600
Totals for interval   rain mm
Minimum is .000000 at 960120 60000
Maximum is 50.5000 at 960203 180000
Mean is        3.68863
```

Click **GRAPH**, click **GRAPH OVER TIME**,
click **OK** to get:

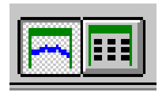


AUTOMATIC Y-AXIS SCALING

The y-axis limits were set automatically to include the range 0 to 50.5 determined by process **EXTREMES**, and to make the scale a power of 10 times one of the numbers 1, 1.25, 1.6, 2, 2.5, 3.2, 4, 5, 6.4 or 8 units/mm. On page 2.14 we set the y-axis **Height** to be 40 mm in the **GRAPH** menu's **OPTIONS** dialog. The labels are rounded to the nearest integer. These scales enable numbers to be read from the graphs using an engineers scale rule. The ratios of successive scales are either 1.25 or 1.28 which is enough to be obviously different, but never so great as to be inconvenient.

BUTTONS THAT SWITCH FILE WINDOWS

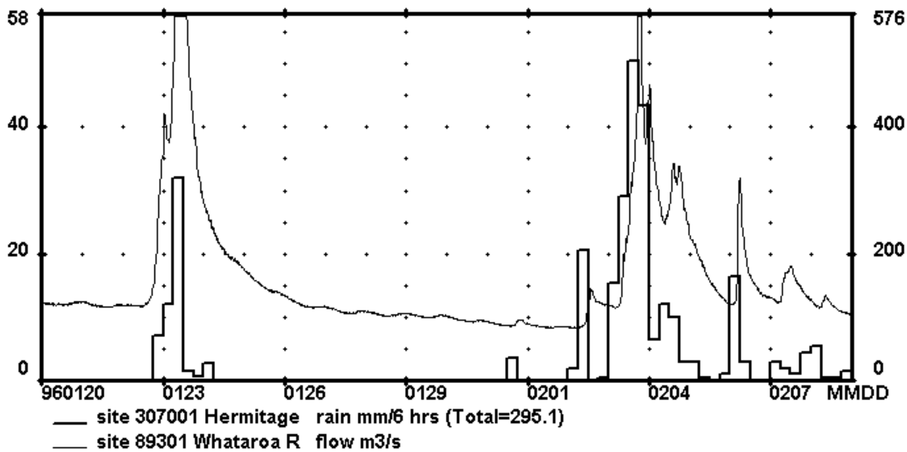
Our graph is in the File graph window, and to change back to the File text window, click **VIEW**, click **TEXT**. Alternatively use these buttons in the program banner to switch between the File graph and text windows.



Y-AXIS SCALE FOR OVER PLOT

We are now ready to overplot the river flow: click **DATA**, and in the data dialog box: set **Site Title**: Whataroa R., select **Measurement**: flow (Rated) and click **OK**.

Click **GRAPH**, click **GRAPH OVER TIME**, set **Vertical Axis Max** = 570 in the Plgraph dialog. River flow data is skewed with large values (in this case up to 1155) for a small fraction of the time. When comparing rain data it is best to emphasise the variations of flow somewhat less than the maximum, and to cut off the maxima. Thus we have set the **Vertical Axis Max** to a lesser value, and in particular a value that makes the graticule provided for the rain series equally applicable to the flow series. We set a number smaller than 580, and let the automatic scaling choose the maximum that corresponds precisely to the standard scale of 6.4 units/mm. Click **OK** to get the graph we require



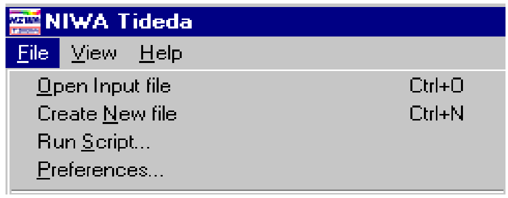
Clearly every flood in the Whataroa River is caused by a storm that rains at Hermitage, at least according to this small sample of the two records.

CHAPTER 3 FILE MENU

When we start TIDEDA the first menu we use is always the **FILE** menu.

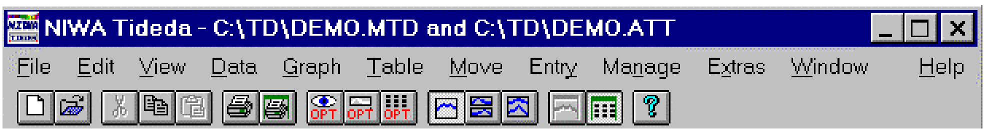
The examples in the following description assume that we have created a folder c:\td\ and put there the two files Demo.mtd and Demo.att, which are supplied with the program.

Start the program by double clicking its icon >, to get a program banner.

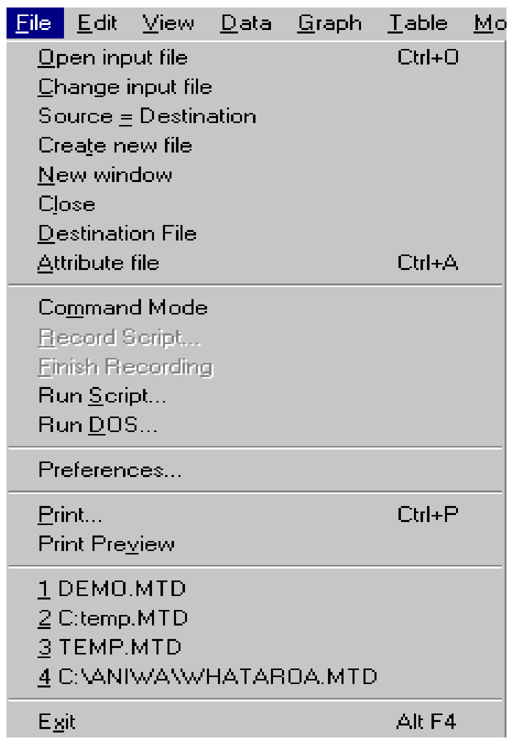


Click **FILE** in the program banner to get a list of processes >.

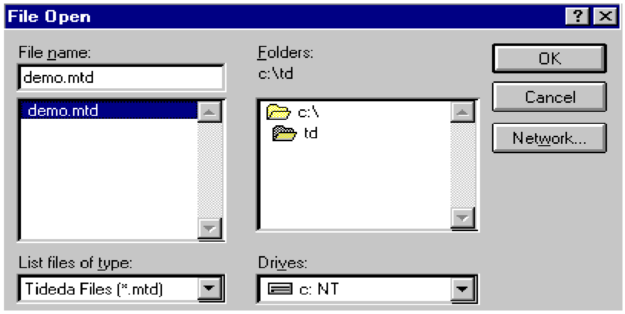
Either of the first two of the listed processes will open a File window, and when we do this the banner and list expand as follows:



The expanded list includes the 4 items in the shorter list. This chapter describes the items in the order they are in the expanded list, except that the numbered file names near the end of the list are described on page 3.3.

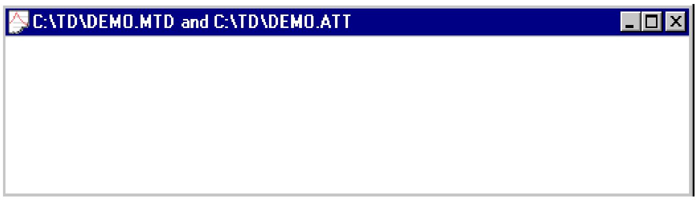


OPEN INPUT FILE



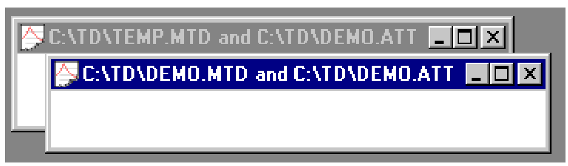
Click **FILE**,
click **OPEN INPUT FILE** to get this dialog >
in which we select
Folder c:\td
File name demo.mtd

FILE WINDOW



Click **OK** and this File window opens >.

This window has the name of a file in its banner. There will also be an Attribute file name in the banner if one is available, but it is not necessary for processing to continue. File windows are where the program prints and graphs information from files.



We can open several File windows >, but only one can have "the focus" which makes its banner blue. Two windows can be opened for the same file so that we can view together information obtained at different times during a session.

Each File window, sometimes called a File text window, has an associated graph window which cannot be seen until a **GRAPH** menu process has drawn a graph in it.

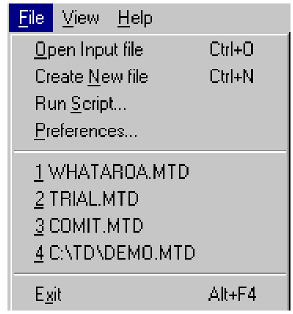
CURRENT FOLDER

When we click **OK** in the **OPEN INPUT FILE** dialog, we set the current folder, in this case to c:\td. Until we reset it again, which requires that we use this dialog again, this folder is assumed except when explicitly set otherwise.

The most recently used files are named in the **FILE** menu, and names with a path attached are not in the current folder. Click on any one of these file names to open a window with that file. This will not change the current folder.

The "Start in" property of the program shortcut sets the initial "current folder".

LIST OF RECENTLY OPENED FILES

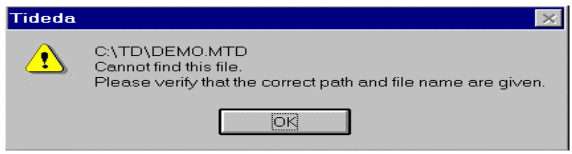


When we close all the File windows the program banner reverts from 12 back to 3 menus. After we have opened several files, the **FILE** menu includes the names of the most recently opened. For example the menu it might look like this >

File names in the current folder are listed without the directory path, but others are not. For example C:\TD\ has been prefixed to DEMO.MTD in the above list for this reason.

If we click on one of these listed files, a window opens connected to that file. The effect is different when the file is not in the current folder from when we use the **OPEN INPUT FILE** dialog, because opening a window this way does not change the current folder.

If we click on one of the listed files, but it no longer exists we get this message >.



CHANGE INPUT FILE

Click **FILE**, click **CHANGE INPUT FILE** to get the same dialog as **OPEN INPUT FILE**, and it changes the file attached to the File window that has the focus. It does not open a new File window.

SOURCE=DESTINATION

Click **FILE**, click **Source=Destination** as a shortcut. It has the same effect as using **CHANGE INPUT FILE** to select the current **Destination file**.

CREATE NEW FILE

Click **FILE**,
click **CREATE NEW FILE**
to get this dialog >
and set the file properties
for the new Tideda data file.

We can change these properties of an existing file using the **MOVE** menu's process **RELEASE**.

The **Number of batches** is rounded up to a multiple of 16 and cannot be larger than 32,000. Specify less than this maximum to reduce the storage space occupied by the directory at the start of the file. This space is 512 bytes for every 16 batches and is 1 Mb for 32,000 batches. See the chapter "Organisation of Data" for more about Batches.

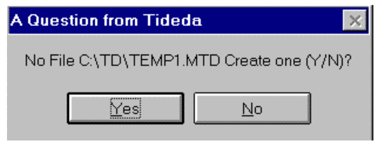
An **Extended Directory** enables the file to hold directory information copied into it from other files without copying the contents of those files. Thus:

1. we can create a single name for a collection of files we wish to use together; and
2. we can incorporate edited information with a file stored on CDROM, thereby editing the contents of the "read only" file without copying it; which can be convenient when a read only file is very large.

See page 15.9 for more about Extended Directories.

The **Destination File** name is the name of the new file. To reduce the amount of typing we can use the **DESTINATION=SOURCE** or **BROWSE** button to find a file with a similar name, then specify the new name by editing that name.

Processes that write to a Tideda file that does not already exist, seek our response to this question >
If we respond **YES** it is created with the same number of batches as in the current input file.



NEW WINDOW

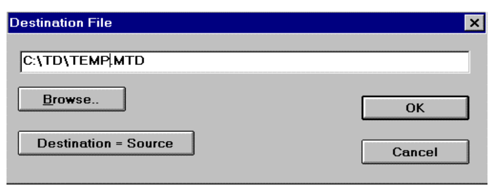
Click **FILE**, click **NEW WINDOW** to open an empty File window, connected to the same files as the window that has the focus, and then transfer focus to the empty window. A number is added at the end of its label >

CLOSE

Click **FILE**, click **CLOSE** to close the File window with focus, and shift focus to the window that previously had the focus.

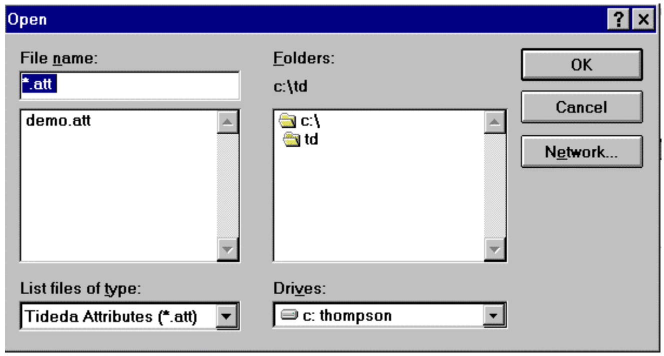
DESTINATION FILE

Click **FILE**, click **DESTINATION FILE** to get this dialog >
and type in a **Destination File** name.



This simply presets the name that appears in the process dialog of Tideda processes that write data to the destination file.

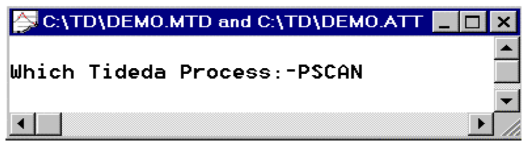
ATTRIBUTE FILE

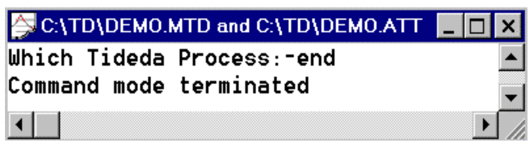


Click **FILE**, click **ATTRIBUTE FILE** to get this dialog > and set the **Attribute File** connected to the File window which has the focus.

To create a new Attribute file, set the new file name then use the **MOVE** menu's **ATTBUILD** command or the **EDIT** menu's **ATTRIBUTES** command to add Tideda attribute records. Ignore any “Attribute file not found” or “cannot open attribute file” messages when setting the new file name.

COMMAND MODE





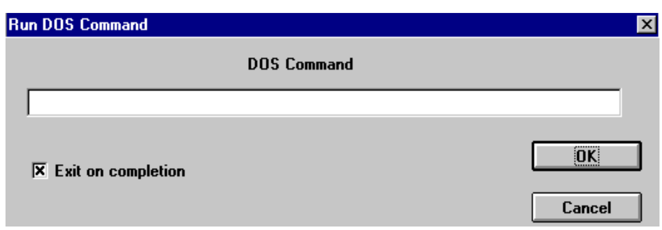
Click **FILE**, click **COMMAND MODE** to get the Tideda command prompt in the File window. This emulates the command line interface of the earlier versions of Tideda. The commands that can be typed here are the same as those used by Tideda script files (see chapter 11- TSF Language).

Type **end** to exit from command mode.

RECORD SCRIPT, FINISH RECORDING & RUN SCRIPT

These three processes use the TSF language and are described in chapter 11.

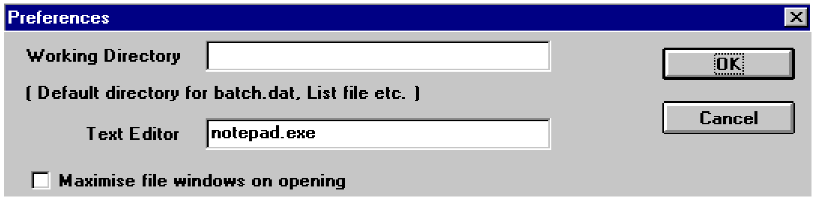
RUN DOS



Click **FILE**, click **RUN DOS** to get this dialog > in which we type a DOS command.
Click **OK** to run it.

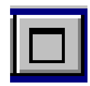
PREFERENCES

Click **FILE**, click **Preferences** to get this dialog >



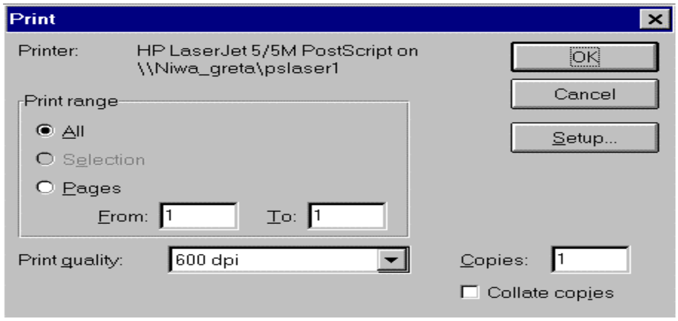
The **Working Directory** is used for the following files when a full path name is not specified: Timouse.dat (see page 2.1), the Batch file Batch.dat (see page 8.2), List files and intermediate files created by the **GRAPH** menu's processes **CUSUM** and **DOUBLE MASS**. When left blank in this dialog, the current folder is used, but this is not recommended (see page 3.2).

We specify a **Text Editor** we are accustomed to using. The program will assume Notepad unless we specify otherwise.

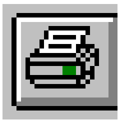


When we select **Maximise file windows on opening**, this button > at the top right of the window is automatically clicked when a new File window is opened. This is advisable when we are using a small display, with desktop area 800 by 600 pixels or less, and then it is also adviseable to maximise the program window.

PRINT



Click **FILE**, click **PRINT** to get an operating system dialog like this > click **OK** to print the contents of the File window that has the focus.

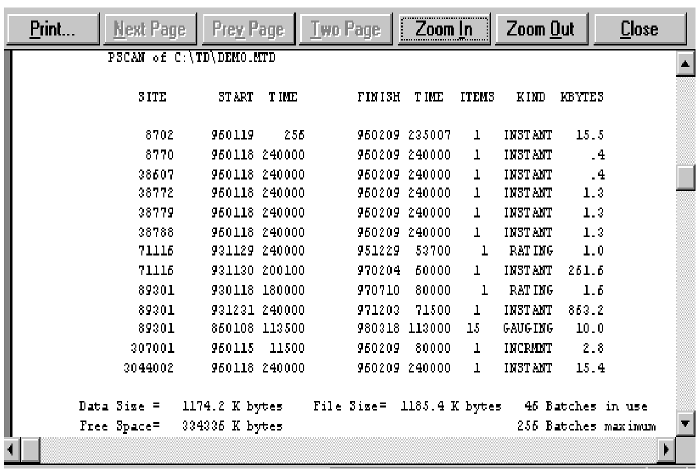


< This button is a short cut equivalent to this sequence.

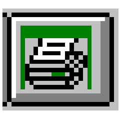
In this manual an explanation about setting **PRINT SETUP**, **PAGE SETUP** and **FONT** parameters for the Graph window is in the **GRAPH** menu **OPTIONS** dialog, and on pages 5.3 & 5.4.

The Text window has a different set of printer parameters which are set in the **TABLE** menu **OPTIONS** dialog, see page 6.1.

PRINT PREVIEW



Click **FILE**,
click **PRINT PREVIEW**
to get this dialog >



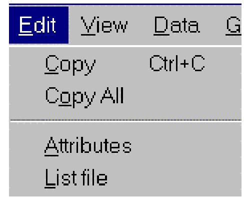
< This button is a short cut equivalent to this sequence.

This dialog enables us to examine all parts of a page that will not fit in the File window.

The following steps prepared this illustration.

1. Click **FILE**, click **OPEN INPUT FILE**, select c:\td\demo.mtd.
2. Click **FILE**, click **COMMAND MODE**, type PSCAN <return>.
3. Click **TABLE**, click **OPTIONS**, click **FONT**, select Courier New 8
4. Clicked **FILE**, click **PRINT PREVIEW**

CHAPTER 4 EDIT, VIEW & DATA MENUS



Click the **EDIT** menu to get this list of processes for editing text. >

They can be used to transfer graphs as well as text to reports prepared by word processors such as the Microsoft WORD program.

COPY

Highlight any part of the contents in the File text window, by dragging the cursor.

Click **EDIT**, click **COPY** or click the **COPY** button on the toolbar  to copy the highlighted text to the Clipboard. Then put the cursor in another program's window and paste the text there. When the Graph window is displayed the entire graph is always copied. Pressing theCtrl-C keys on the keyboard is a shortcut to copy to the clipboard.

COPY ALL

Click **EDIT**, click **COPY ALL** to copy all the contents of the Text window to the Clipboard without any highlighting.

ATTRIBUTES

Site Attribute information is kept in an Attribute file separate from the Tideda file(s), which contain the data described. We should prepare attributes for all our data to reduce the likelihood of mistakes about what the filed numbers represent, although the program does not require attributes.

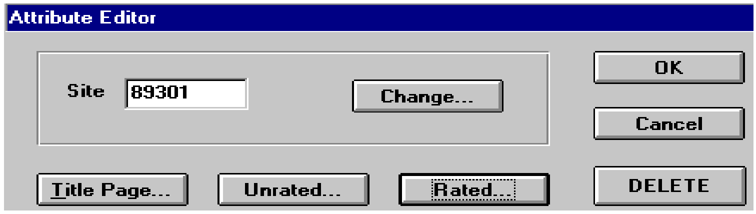
The **MOVE** menus's process **ATTBUILD** adds to the attached Attribute file the subset of Attributes that apply to the file connected to File window. The subset is taken from a larger Attribute file called Sites.att. Those who manage a database should maintain a Sites.att file so that they can reliably prepare attribute files to accompany any data they supply from their database.

The attributes specify a **File Unit** name, a **Presentation Unit** name and a **Divisor**:

Presentation Unit = File Unit / Divisor

We select which units for a particular report on the **DATA** dialog. Note that the file quantities, and the Divisor are represented as integers, that is with no digits after the decimal point.

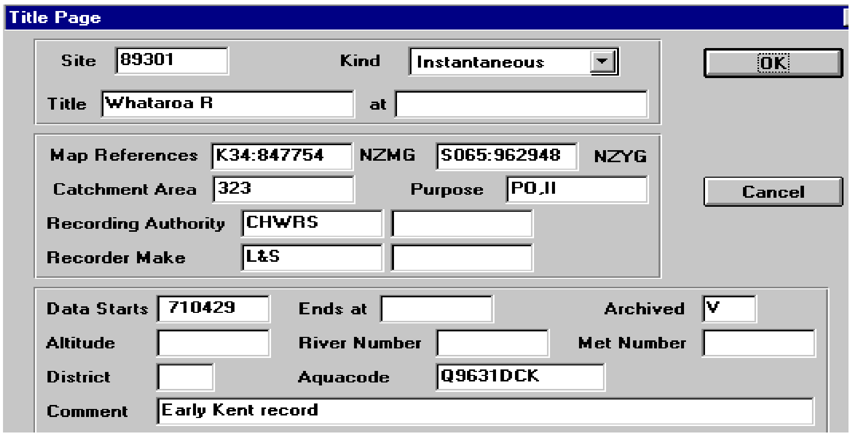
Click **EDIT**, click **ATTRIBUTES** to get this dialog > and to enter new or edit existing Attributes.



Click **Change** to get the **DATA** dialog where we can select any **Site** in the Tideda file connected to the File window. We can also type any other **Site** number directly into the Attributes dialog.

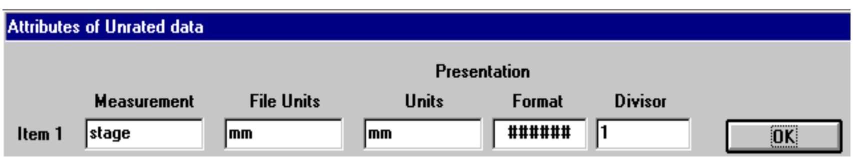
Click **Title Page** to get this dialog >

Kind is a key, so there can be three different sets of attributes for each **Site**.



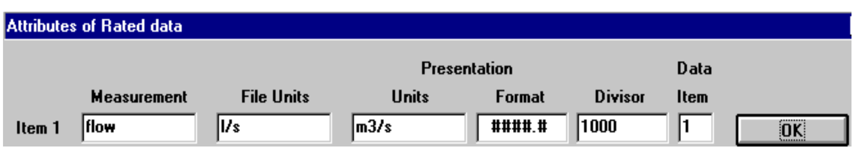
The program uses **Title** as a label, but does not use the other information in this dialog.

Click **Unrated** to get this dialog >



Only 1 line of 15 lines is shown here, and there is a separate line for each **Data Item** in the unrated series.

Click **Rated** to get this dialog >

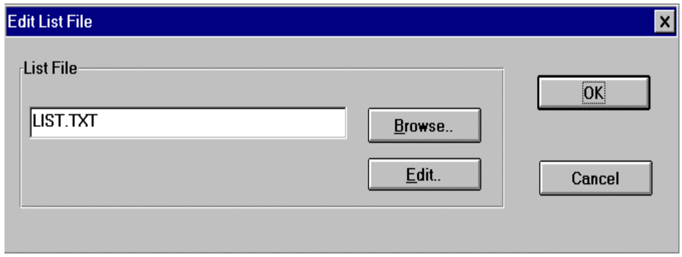


Only 1 line of 4 lines is shown here. Thus there is space for 4 different series of Ratings, each of which operates on the unrated series **Data Item** specified at the right of the line.

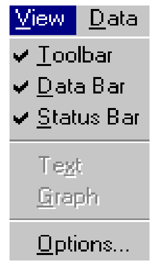
LIST FILE

Click **EDIT**, click **LIST FILE** to get this dialog >.

The List file is a general-purpose file we examine and edit using a text editor.



VIEW MENU



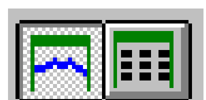
Click **VIEW** on the program banner to get this menu list >

We can simplify the screen by removing:

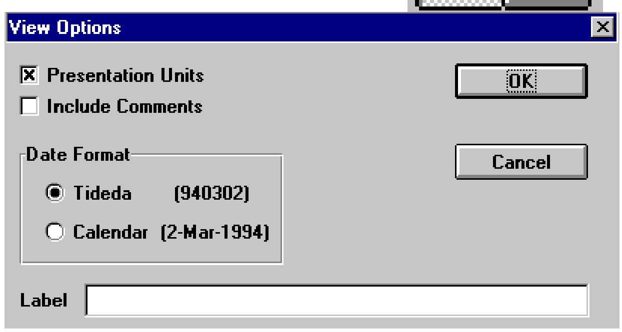
- the bar of buttons (click **VIEW**, click **Toolbar** to deselect)
- the bar below the buttons (click **VIEW**, click **Data bar** to deselect)
- another bar at the bottom of the screen (click **VIEW**, click **Status bar** to deselect).

These three bars provide shortcuts and advice. They are not be mentioned again in this manual apart from the effects of the toolbar buttons.

The **Text** and **Graph** buttons in the **VIEW** menu become live when we opened a pair of File windows, one for text and the other for graphs, and only one can be selected for viewing. Alternatively we can use these buttons in the program banner to switch windows >.



Click **VIEW** click **OPTIONS** to get this dialog >



Presentation units has effect if an Attribute file is attached and contains the scaling factors to convert data from File units. Selection here presets the same option in the **DATA** dialog.

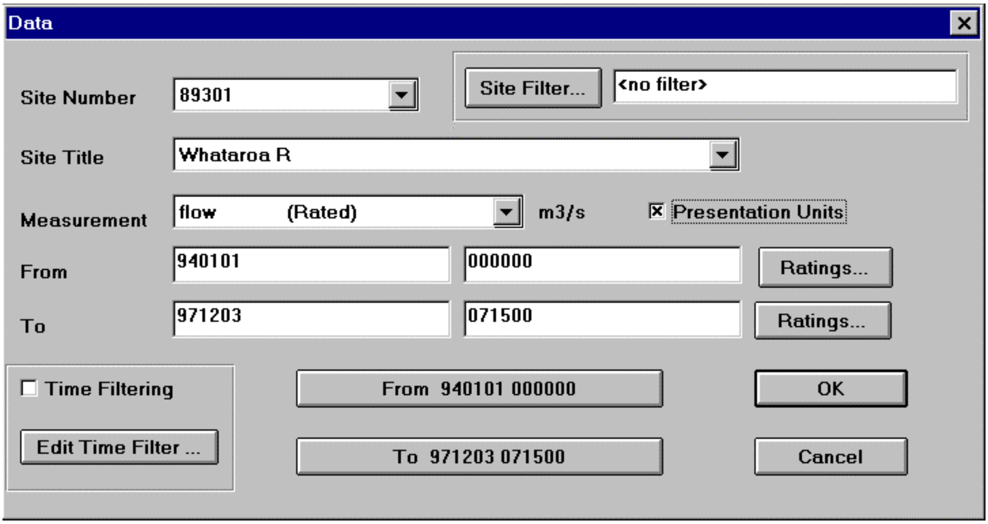
Include Comments has effect if a Comment file is attached and we use the **COMMENTS** process or some **GRAPH** processes.

The **Date Format** applies to both data entry and presentations.

The **Label** text specified here presets the Label in process dialogs.

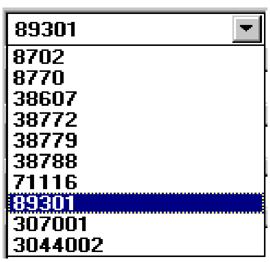
DATA DIALOG

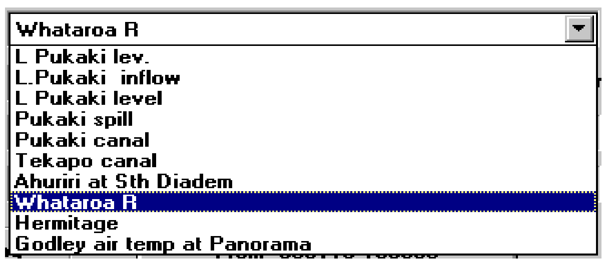
Click **DATA** in the program banner to get this dialog >



SITE SELECTION

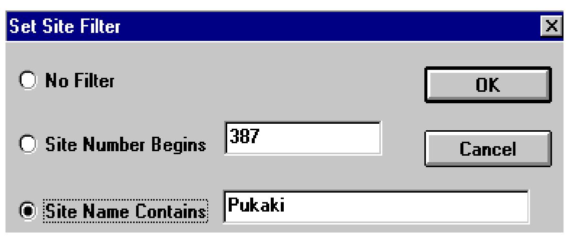
We select either a **SITE NUMBER** or a **SITE TITLE** using one of the following the drop down menus. The titles are listed as <untitled> for sites without a Title Attribute.





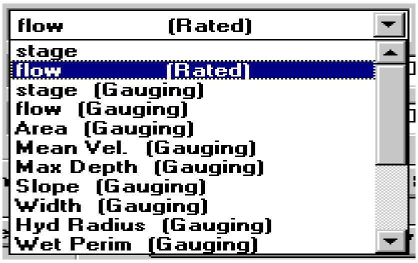
If the file has too many sites for convenient selection from the drop down lists we should use the **SITE FILTER** dialog >

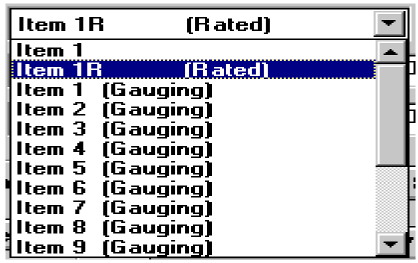
This filter setting would restrict the drop down lists to the first six sites.



MEASUREMENT SELECTION

To select a particular **MEASUREMENT**, we use another drop down list. Like the preceding list, the contents of this list depend on what Attributes are specified. The following illustrates two lists of the same data, with and without Attributes.





We will now use the following summary to explain this long list of different **Measurements**. This summary was written by the **MANAGEMENT** menu's process **SCAN**.

SITE	START	TIME	FINISH	TIME	ITEMS	KIND	KBYTES
89301	930118	180000	970710	80000	1	RATING	1.6
89301	931231	240000	971203	71500	1	INSTANT	863.2
89301	860108	113500	980318	113000	15	GAUGING	10.0

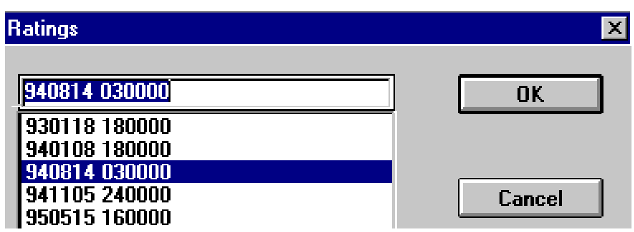
The measurements are all from Site 89301, and cover three different time periods. We will deal with times next. This summary shows that most of the data (863.2 kbytes) are of Kind=Instant (that is a continuous time series) from 1 January 1994 to 3 December 1997 with one item. Continuous series are always listed first in the **Measurement** list, and so in this case we can see that the one item is called "stage". Next ratings are listed, and in this case there is one series and it converts stage to flow. Finally gaugings are listed, and in this case the series has 15 items of which the first two are stage and flow respectively.

TIME SELECTION

The bottom half of the **DATA** dialog is where we set the time period we wish to examine. WE can type new numbers directly into the white selection boxes, or edit numbers already there.

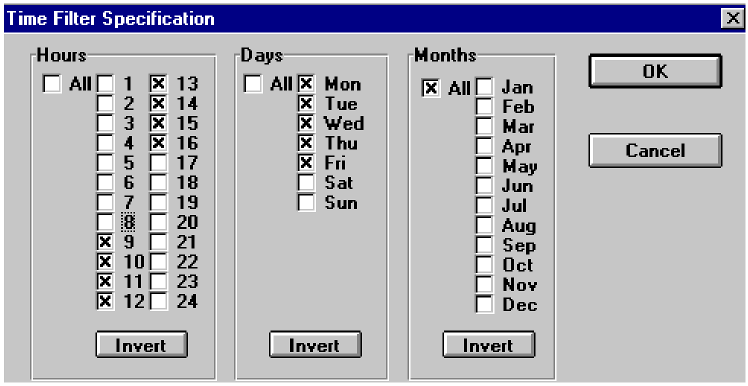
The time **limits of available data** are the labels on the long buttons, bottom centre. If we click one of these buttons, that limit is transferred to the relevant selection box. In this case the selected **Measurement** is "flow (Rated)" and the time limits are the period for which there are both stage and ratings: `940101 0  971203 71500`. If the selected **Measurement** was one of the Gauging items then the limits would be `860108 113500  980318 113000`.

To see the **times of the ratings**, click one of the **RATINGS** buttons to get this list >.
If we select a time then click **OK**, the time of that rating is transferred to the adjacent selection box.



TIME FILTER

Click
EDIT TIME FILTER
to get this dialog >
which can restrict
our examination of
a time series to
certain
hours of the day,
and/or
days of the week
and/or
months of the year.



We specify which hours, days and months in this dialog. Select the **Time Filtering** button to cause it to happen. It will slow down the rate of processing.

The filter setting illustrated restricts analysis to the period 8.00 am – 4.00 pm on working days of the week, and might be used on a series of electricity price to determine the average price at this time.

WATCH OUT ! If we inadvertently leave **Time Filtering** selected in the **DATA** dialog, subsequent processes will find inexplicable blanks in the data.

CHAPTER 5 GRAPH MENU

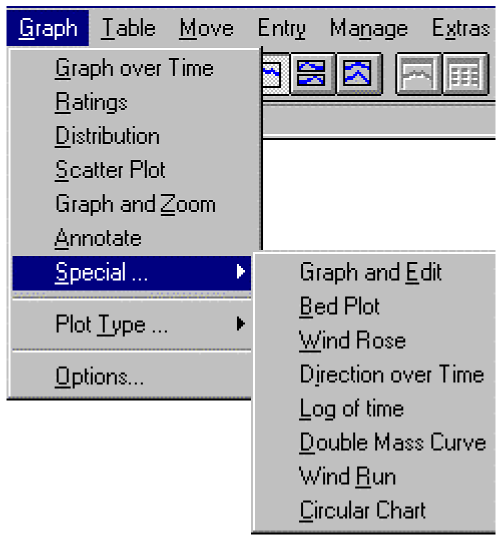
The basic graphing process is **GRAPH OVER TIME**. Other processes are available to deal with specific types of data or to process data in different ways.

The following list on the left shows the order in which the menu items are described. The first three are used in conjunction with all the others. The next five are together because their descriptions are linked. The rest are in the order in the menu

ANNOTATE adds text to a graph.

PLOT TYPE selects new, same page or over plot.

OPTIONS opens the "graph options" dialog.



GRAPH OVER TIME, item vs. time.

SCATTER PLOT, item vs. item.

RATINGS, unrated vs. rated.

BED PLOT, difference from rating vs. time.

LOG OF TIME, item vs. log-time.

DISTRIBUTION, item vs. % of time.

GRAPH AND ZOOM, item vs. time.

GRAPH AND EDIT, item vs. time.

WIND ROSE item vs. % of time and direction.

DIRECTION OVER TIME item vs. time.

DOUBLE MASS CURVE as Cusums items vs. time

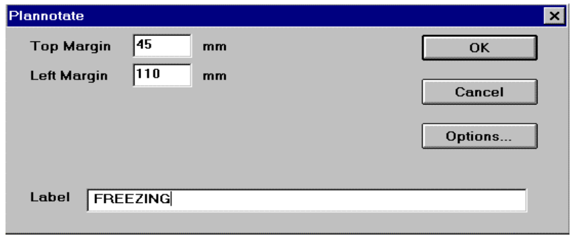
WIND RUN item vs. time.

CIRCULAR CHART item vs. time.

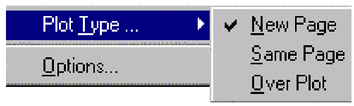
ANNOTATE, PLOT TYPE, OPTIONS

Click **GRAPH**, click **ANNOTATE** to get this dialog > which we use to add text to a graph.

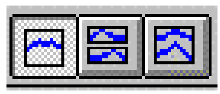
These values were used to add the label FREEZING to the graph on page 5.17, which illustrates process **LOG OF TIME**.



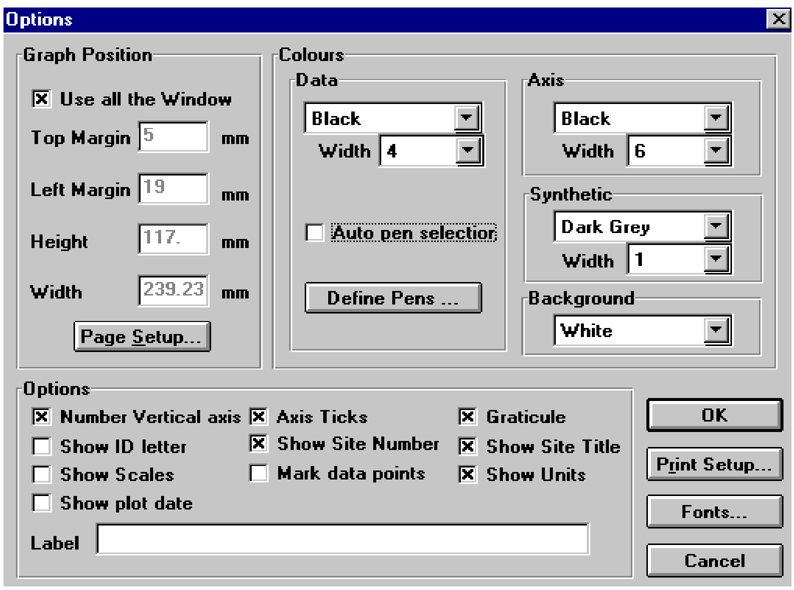
Click **GRAPH**, click **PLOT TYPE** to get this sub-menu > which we use to set where the next graph will appear.



Alternatively click one of these buttons in the program banner >



Click **GRAPH**, click **OPTIONS** to get this dialog >

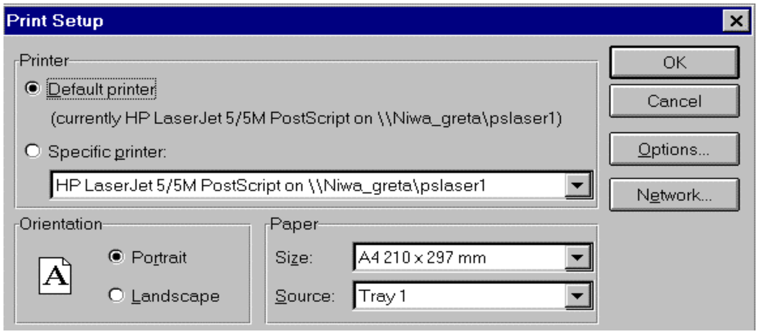


Some option labels are self-explanatory. The less obvious are explained below.

The **Graph Position** dimensions we set are for a <u>page printed on paper</u>, and these are scaled to corresponding dimensions in a <u>window on a computer screen</u> so that the paper page will fit in the window. This fitting of graph to paper is illustrated on page 5.4

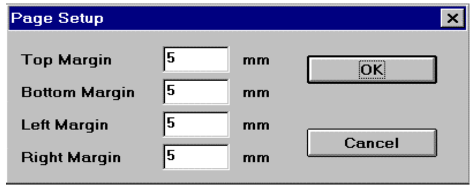
Thus we start with the paper.

Click **GRAPH**, click **OPTIONS**, click **Print setup** to get a dialog like this > where we set the paper size and orientation.

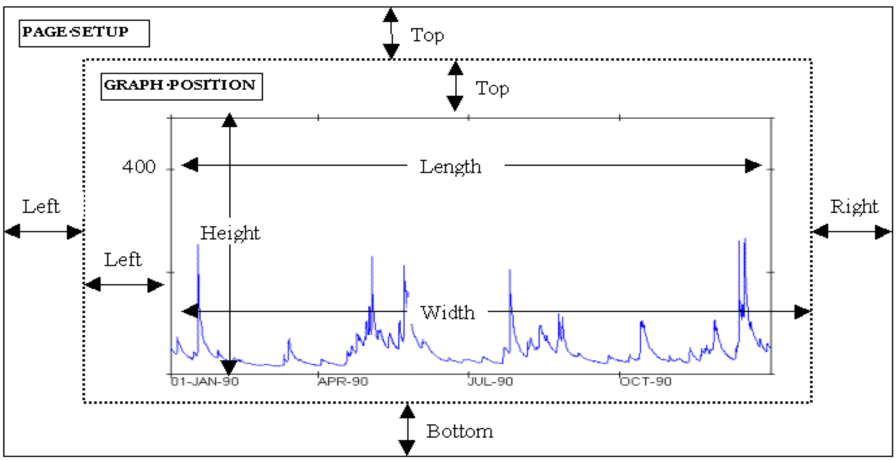


However it is not the whole piece of paper, it is the space on it inside the paper margins that is scaled to fit the window

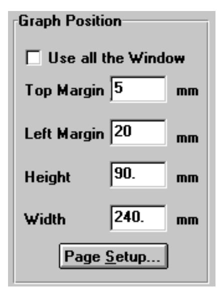
Click **GRAPH**, click **OPTIONS**, click **Page Setup** to get this dialog > where we set the paper margins.



The graph is positioned in the space inside the paper margins as shown in this diagram.



In a **Graph Position** box on the **OPTIONS** dialog > we set the inner margins, height and width of the space for a graph.

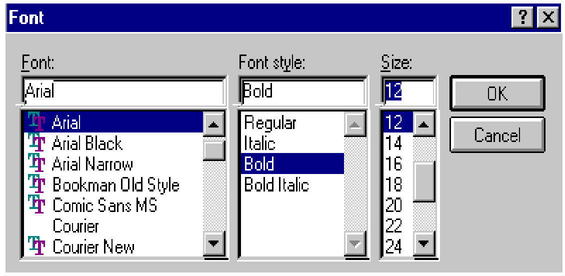


Immediately after starting, it is a good idea to temporarily select **Use all the Window** and let the program assign values to the 4 numbers. Then deselect **Use all the Window** and guided by these values, set these or smaller values.

We deselect **Use all the Window** so that our values do not change if our window is inadvertently resized, which will spoil the layout when we plot two or more times on the same page.

We also set the size of the labels, which must fit in the inner margins, and outside the graph axes.

Click **GRAPH**, click **OPTIONS**, click **FONT** to get this dialog > where we set the font and its size.

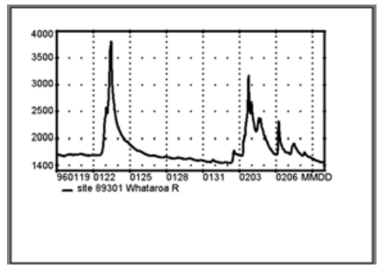


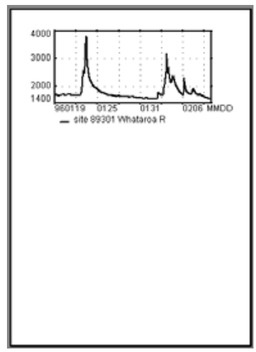
We can experiment as follows to show how the graph dimensions are determined.

Experiment 1

In **GRAPH** menu's **OPTIONS PRINT SETUP** set **Orientation Landscape,** maximise the size of the File window, in the **GRAPH** menu's **OPTIONS** set **Use all the window**, use **GRAPH OVER TIME** to draw a graph, then use the **FILE** menu's

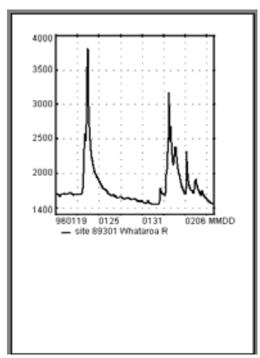
PRINT PREVIEW to see the page >





Experiment 2

In **GRAPH** menu's **OPTIONS PRINT SETUP** set **Orientation Portrait,** use **GRAPH OVER TIME** to draw the same graph, then use the **FILE** menu's **PRINT PREVIEW** to see the page >



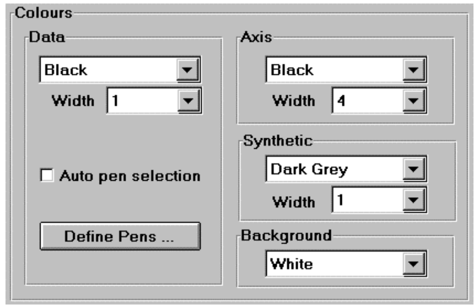
Experiment 3

Halve the width of the File window**,** use **GRAPH OVER TIME** to draw the same graph, then use the **FILE** menu's **PRINT PREVIEW** to see the page >

Notice how the program calculates different values for the **Height** and **Width** of the graph in each case.

PEN SELECTION

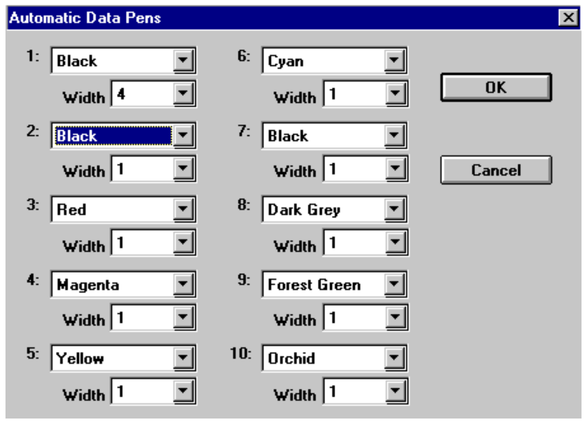
In the **Colours** box in the **GRAPH** menu's **OPTIONS** dialog >
we set properties so that different "pens" plot different kinds of line on the graph.



Pen widths are in arbitrary units that depend on pixel size and we will have to try different values to see what we get on our equipment.

The **Synthetic** pen is used for data read from any Tideda file with a name which ends .SYN.

Select **Auto pen selection** to automatically assign a new pen to each new piece of data, so it can be clearly distinguished on the graph. For this purpose we define a set of pens.

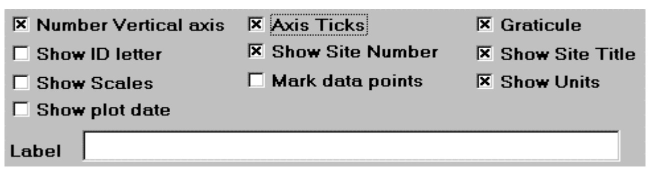


Click **GRAPH**,
click **OPTIONS**,
click **DEFINE PENS**
to get this dialog >

Pen 1 is used for the first line of data on our graph, pen 2 when we **Over Plot** a second line, pen 3 for a third line, and so on.

LABELS ON GRAPHS

Most of these labeling options >
are illustrated on page 5.10



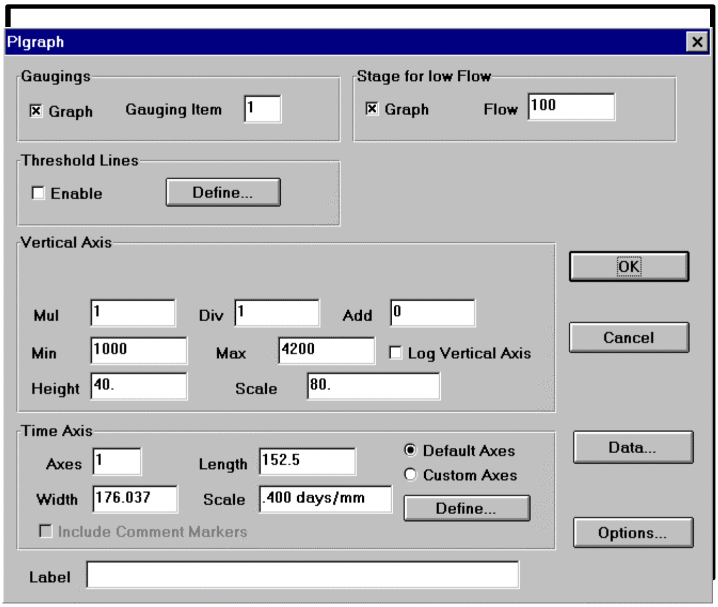
GRAPH OVER TIME

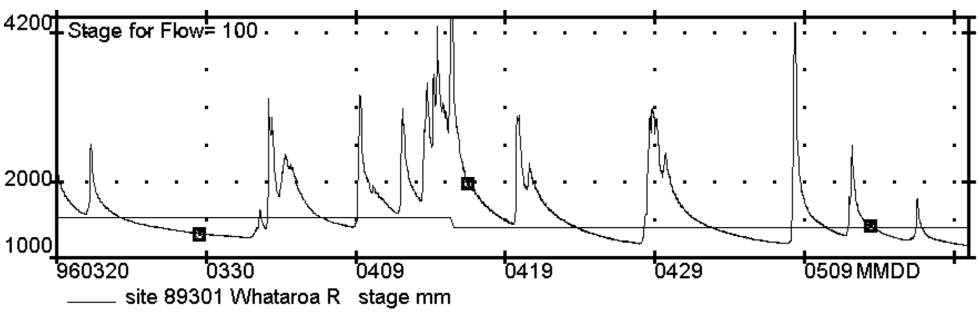
This process plots values (*y*) versus time (*x*). We illustrate it using data from Demo.mtd.

Click **GRAPH**, click **GRAPH OVER TIME.**

Click **DATA**, set **Site Title** Whataroa R **Measurement** stage **From** 960320 0 **To** 960520 0 click **OK**

Set this dialog thus >, click **OK** to do it.





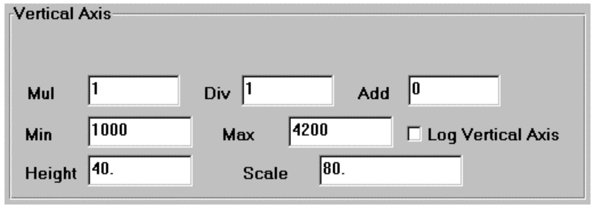
 These symbols are plotted because, in the **Gaugings** box, we set **Graph** and **Gauging Item** 1. Gaugings are measurements made so infrequently that temporal interpolation between them is not valid. In this case they are manual observations of stage, and the symbols confirm the continuous line on three occasions.

The horizontal line stepped at 960416 is plotted because, in the **Stage for low Flow** box, we set **Graph** and **Flow** 100. This line shows the stage that corresponds to a rated flow of 100 m^3/s. The step represents a rating change, and the slope of its riser shows how gradually the change is introduced.

The **Length** 152.5 mm is less than the available **Width** 176.037 and has been adjusted down so that the **Time Axis Scale** can be one of a set that includes 0.4 days/mm. This is explained next.

AXIS SCALES

In the **Vertical Axis** box on the **GRAPH OVER TIME** dialog > we set the y-axis scale. Typically we set **Min** and **Max**, the **Height** is copied from the **OPTIONS** dialog, and the program calculates the **Scale**.



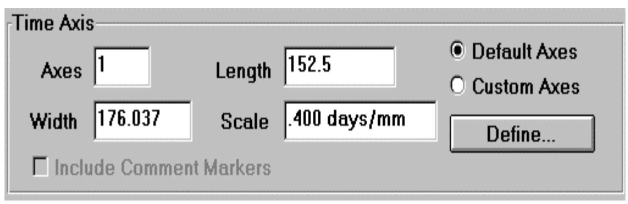
With these values of Min and Height, if we set Max to be anywhere in the range 2561 to 4200 the program will automatically increase the Max to 4200. The same happens if we set Scale in the range 64.1 to 80. This is very frustrating if we do not know why.

The program changes **Max** so that the **Scale** is the next largest number in the sequence: **10, 12.5, 16, 20, 25, 32, 40, 50, 64 or 80** units/mm times a power of 10. In this case (4200 – 1000) / 40 = 80.

The reason this feature was provided is historical, and it is retained because it is still useful. Before computer plotting, numbers were easily read off graphs manually drawn on gridded graph paper. Computer graphs are drawn by equipment that cannot be set up to draw the data with a specific relationship to a pre-printed grid, and pen plotters are too slow to draw a grid. Thus the program used a restricted set of scales that are available on **engineers scale rulers**. The ratios of adjacent scales in this sequence are either 1.25 or 1.28 which is large enough to be obviously different so we know which rule to use, but never so great as to be inconvenient.

Tideda offers an accurately located **Graticule** to assist reading.

In the **Time Axis** box on the **GRAPH OVER TIME** dialog > we set the x-axis scale. Typically we set the number of **Axes** and the **Width** available is copied from the **OPTIONS** dialog, and the program calculates the **Scale**.



The program changes the **Length** so that the **Scale** is the next largest number in the sequence: **10, 12.5, 16, 20, 25, 32, 40, 50, 64 or 80** units/mm times a power of 10. The unit is a minute up to 10 min/mm, then an hour up to 2 hrs/mm then a day.

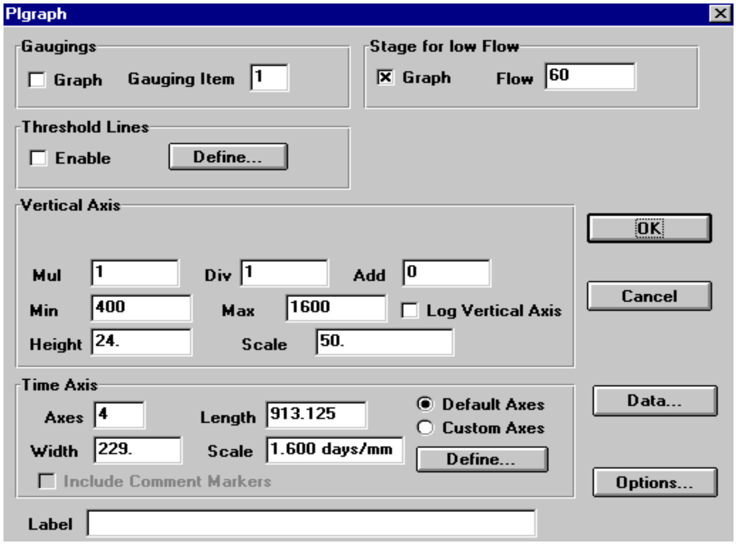
TIME AXIS FOLDED

In the next example we plot four years of data with a folded time axis.

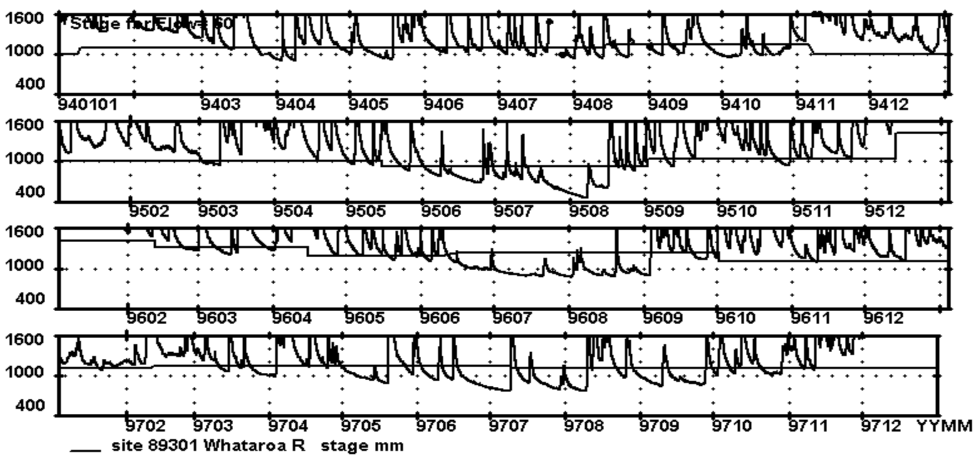
Click **GRAPH**, click **GRAPH OVER TIME.**

Click **DATA**,
set **Site Title** Whataroa R
Measurement stage
From 94 0
To 98 0
click **OK**

Set this dialog thus >, click **OK** to do it.



We often choose values for graph dimensions and data ranges interactively. For example in this case we started with **Axes** 1, **Min** 0 and **Max** 3000, and read off the resulting graph more refined values for a second trial, and so on. By setting the available **Width** 229 mm which is the **Length** divided by the number of **Axes** (rounded up to an integer), we get four graphs of equal length with time of year appearing as a column down the page.

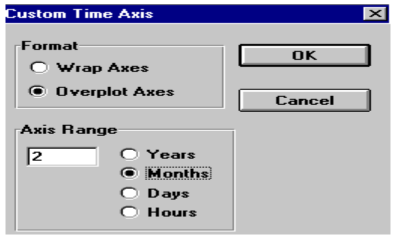


The **Stage for low Flow** line shows that the rating changes are less than the variations of dry weather stage. A graph we will plot with process **BEDPLOT** will show the evidence for the size of the rating changes; see page 5.15.

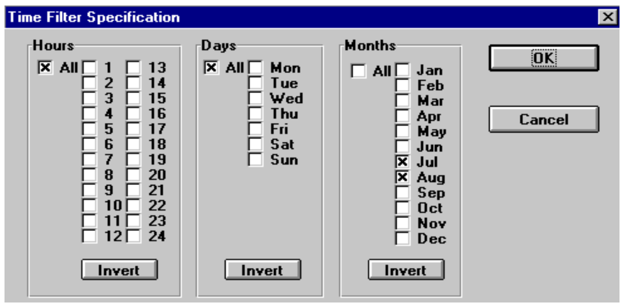
DIFFERENT DATES OVER PLOTTED

The previous graph shows a seasonal pattern with low stages in three of the four winters. We now plot three winters together. We also illustrate time filtering and some of the labeling options.

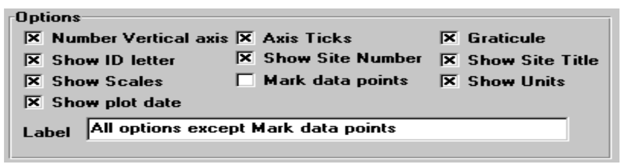
Click **GRAPH OVER TIME** then in the **Time Axis** box > select **Custom Axes**
Click **DEFINE**, set this dialog thus>
click **OK** to exit **DEFINE**.



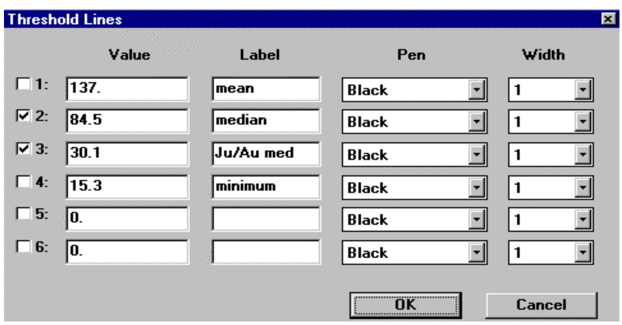
Click **DATA**, set
Site Title Whataroa R
Measurement Flow (Rated)
From 950701 0 **To** 970901 0
select **Time filtering**
click **EDIT TIME FILTER**
set this dialog thus >
click **OK** to exit **TIME FILTER**
click **OK** to exit **DATA** dialog.



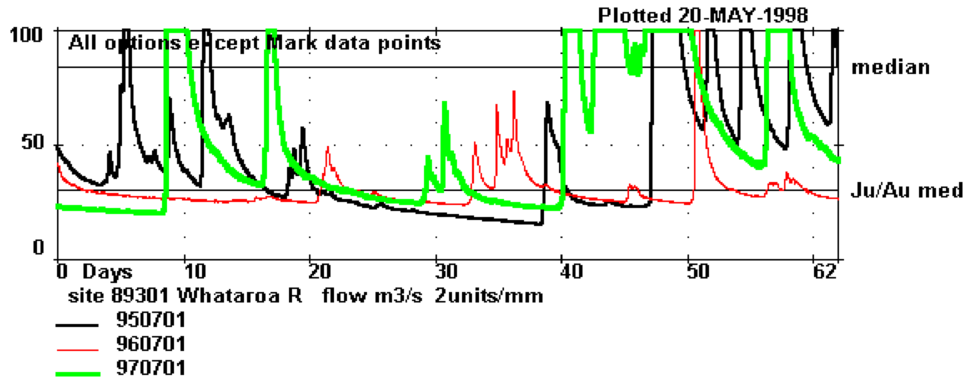
Click **OPTIONS**,
select **Auto pen selection**
click **DEFINE PENS**,
set pens as on page 5.5
click **OK** to exit **DEFINE PENS**.
Set these labels >
click **OK** to exit **OPTIONS**.



In the **Threshold Lines** box in the **GRAPH OVER TIME** dialog select **Enable**
click **DEFINE**,
set this dialog thus >
click **OK** to exit **THRESHOLD**



Click **OK** to exit the **GRAPH OVER TIME** dialog and draw the graph.



This graph compares low flows for July and August in three years. The previous graph of the unrated (i.e. stage) record includes the period of these rated (i.e. flow) records.

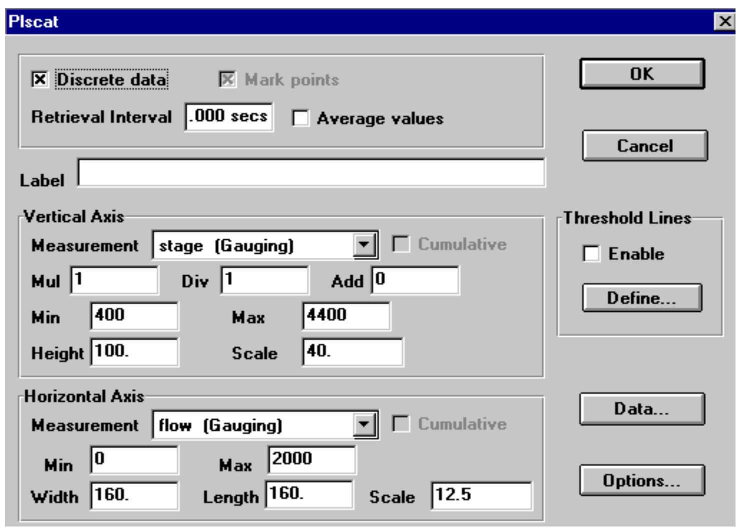
SCATTER PLOT

Scatter plots show the cross-correlation between any two series. The program requires that the input is two items in a multi-item series and this often requires prior work with processes **MERGE 1** and **MERGE 2**. For our example we use the Whataroa River stage-flow gaugings which are already a multi-item series in Demo.mtd.

Click **GRAPH**, click **SCATTER PLOT.**

Click **DATA**, set **Site Title** Whataroa R **Measurement**: stage (Gauging); click the grey boxes **From** 860108 113500 **To** 980318 113000 click **OK**.

Select **Discrete data** because these are gaugings that should be plotted that way.

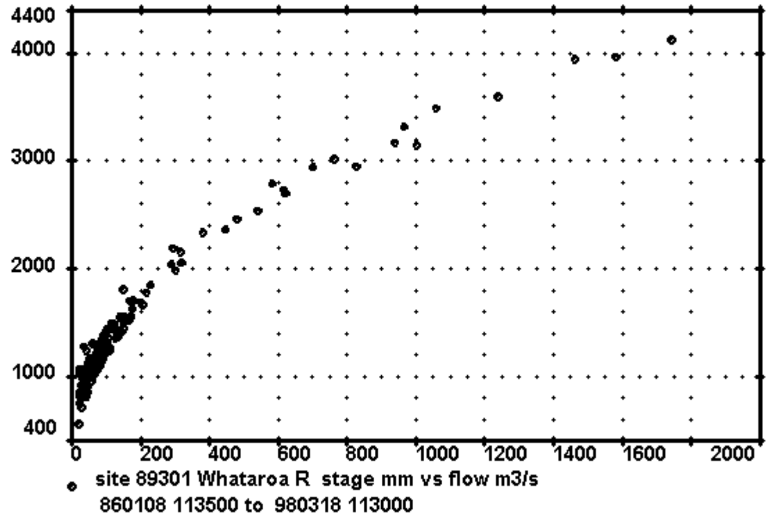


Select **Horizontal Axis Measurement** “flow (gauging)”.

If we do not accurately know the range of each quantity, we set large ranges for stage and flow (or use the Quick Extremes process on page 6-2), then click **OK** to get a graph.

We read accurate ranges off the graph and set them as new Mins and Maxs.

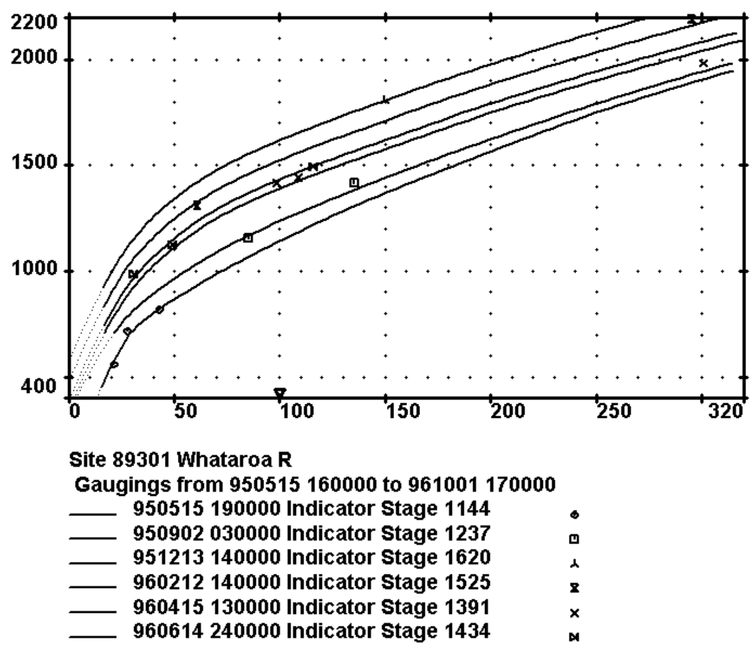
Then we run the process again to get this scatter plot >



RATINGS

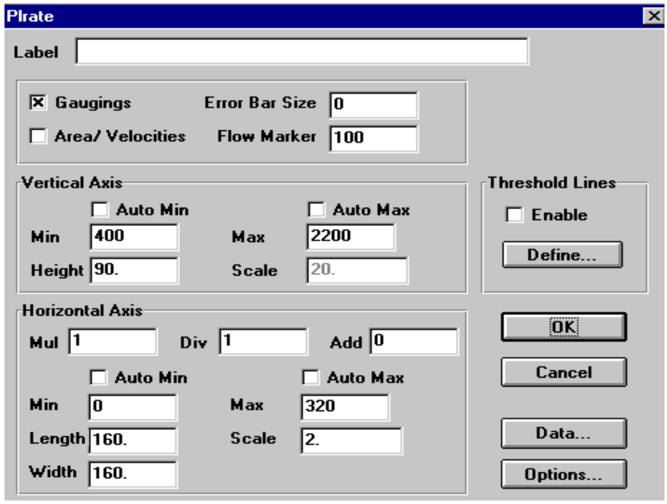
We use rating curves to make use of a relationship that is evident on a scatter plot. The relationship on this scatter plot immediately above would seem to be adequately represented by a single curve when we first see it. However in practice there are many curves, and the 6 curves plotted below are some of them. These particular curves represent the relationship during the 16 months from 950515 to 961001, and are a time series of ratings.

When we look carefully at the scales of the two graphs we see that the rating changes are minor and lie within the scatter of the gaugings. Only 14 of the gaugings were during the period of these ratings, and are plotted on both graphs.



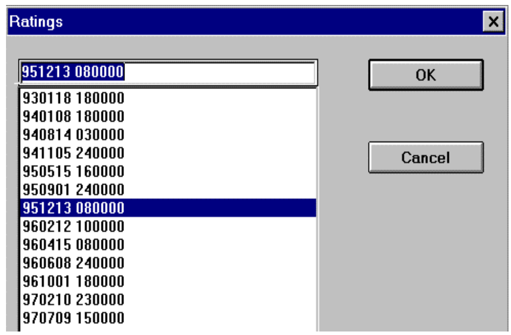
To prepare the above graph of ratings, from the data filed in Demo.mtd, we do the following.

Click **GRAPH**
click **RATINGS**
set this dialog thus >



Click **DATA**, set
Site Title: Whataroa R
Measurement flow (Rated)

Click the **RATINGS** button in the **DATA** dialog on the **From** line to get this dialog > select "950515 160000", click **OK**.



Click the **RATINGS** button in the **DATA** dialog on the **To** line to get this dialog > select "961001 180000", click **OK**.

Change the **To** time in the **DATA** dialog from 180000 back one hour to 170000 to exclude the October rating while retaining all the gaugings since the June rating.

Click **OK** to exit the **DATA** dialog.

Click **OK** in the **RATINGS** dialog to draw the graph.

When we do not know appropriate values for the Min and Max click the **Auto Min** and **Auto Max** boxes then use the resulting plot as a guide to choosing more accurate parameters for a second plot.

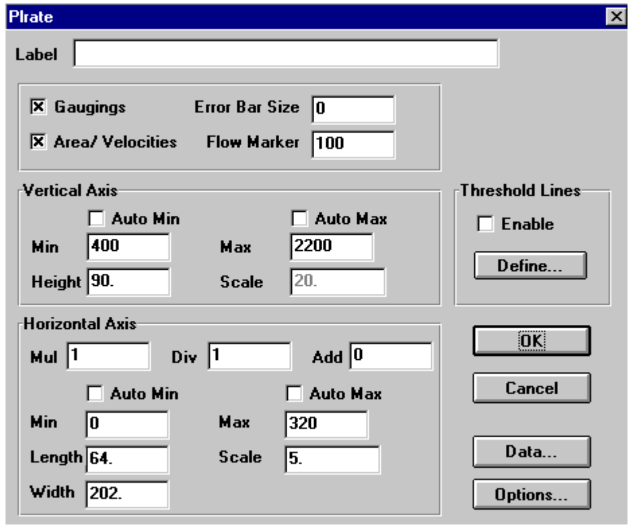
Different symbols are associated with each rating, and are labeled in a legend where each date represents a new rating.

The rating curves are identified on the plot by the Indicator Stage in the legend which corresponds to **Flow Marker** = 100 set in the **RATINGS** dialog, and a V symbol on the x-axis.

We use ratings to transform a measured quantity into a related quantity. For example, in the Whataroa R, water stage is measured and river flow is required. Direct measurement of flow (i.e. gauging) is time consuming and therefore costly, and done only when necessary to update the changing relationship between stage and flow.

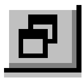
Rating curves constructed for other relationships than "stage to flow", for example "flow to sediment load", can be used in the same way.

We can graph more details about ratings as follows.



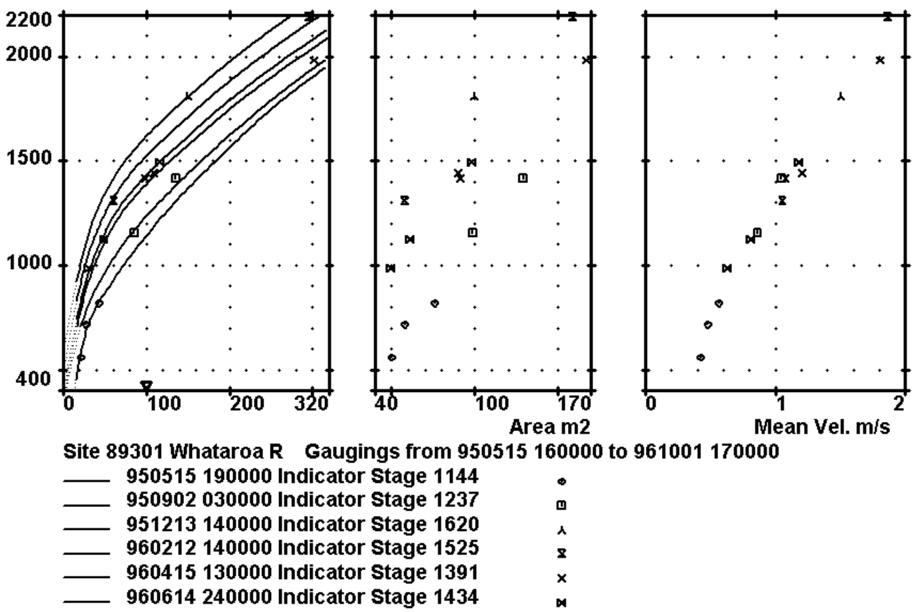
Click **GRAPH**,
click **RATINGS**
to get this dialog again >

This time we:
1. select **Area/Velocities**;
2. maximise the File window size by clicking this button >



if it is present at the top right of the program and/or File windows;
3. in the **Horizontal Axis** box set a small **Length** and set **Width** to be (3 x **Length**) + 10.

Click **OK** to graph
(1) stage vs. flow, (2) stage vs. water cross-sectional area, and (3) stage vs. mean velocity.



This illustration requires that the first 4 items in the Gauging series are 1. Stage, 2. Flow, 3. Cross-section area, 4. Mean velocity. A Tideda standard arrangement for filing river flow gauging data is described on page 8.4.

BED PLOT

Given a series of gaugings and the related ratings, this process plots as a series the difference:

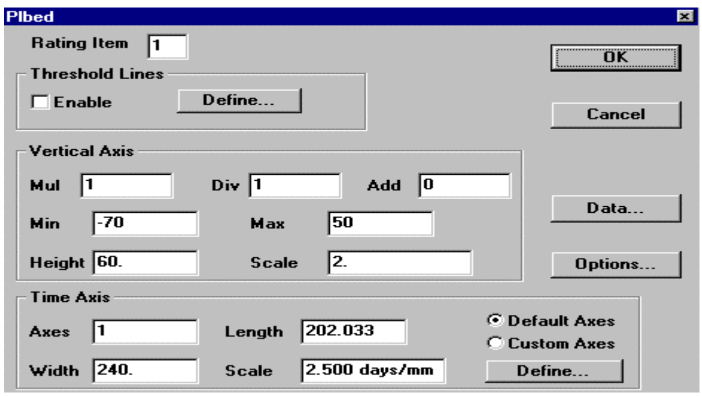
(unrated value corresponding to a gauged value of a rated quantity) - (gauged value of the unrated quantity)

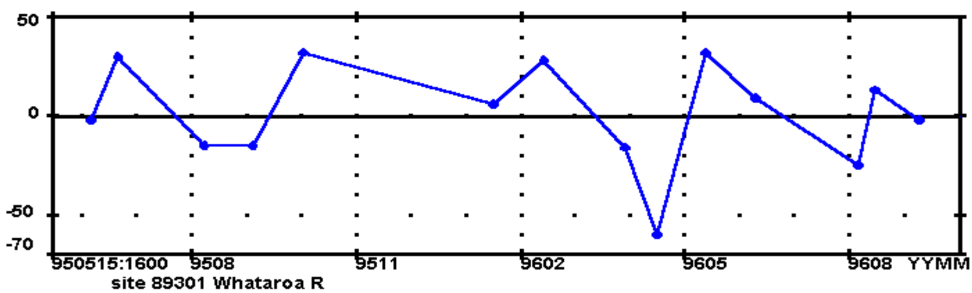
E.g. (stage on the rating corresponding to a gauged flow) - (gauged stage)
The stage-to-flow rating curve usually used to calculate flow from stage, is used the other way around with measured flow as input to calculate stage. This stage ideally would match the gauged stage but differences occur due to errors and to changes in the rating curve. In the absence of other errors, the differences correspond to changes in the level of the riverbed. Thus the graph gives a pictorial view of the level of the riverbed.

Interpretation of what the differences imply can be achieved from an examination of the resulting plots. If the plot seems to fluctuate randomly around a difference of zero, like the first example below, it suggests that the rating curve(s) are probably accurate. Differences that have a similar temporal pattern to the recorded stage may indicate that part of the rating curve is in error, e.g. the high stage part of the rating. Runs of positive or negative differences indicate that the low stage part rating may have changed, like the second example below.

In the following example we continue to use the Whataroa River ratings and gaugings used in the previous example.

Click **GRAPH**, click **SPECIAL**, click **BED PLOT** set this dialog thus > click **OK** to do it.

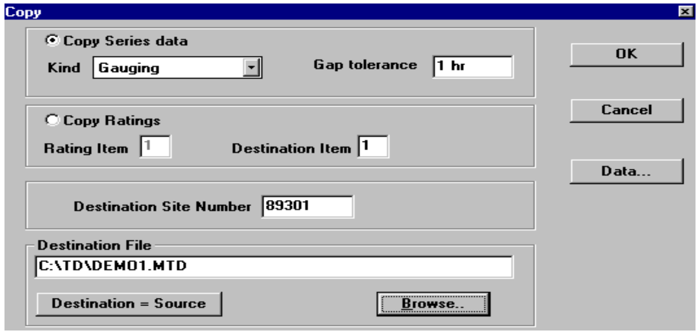




These Bed Plot deviations with one exception are within 35 mm of zero and scatter in a random way, indicating that this is the size of random errors at this site.

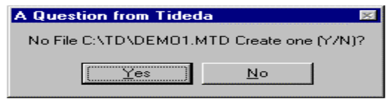
We now examine the evidence for the magnitudes of the rating changes by making a different kind of bed plot. We use the same gaugings as before and just one rating, which we **COPY** to a temporary file.

Click **MOVE**,
click **COPY.**
set this dialog thus >

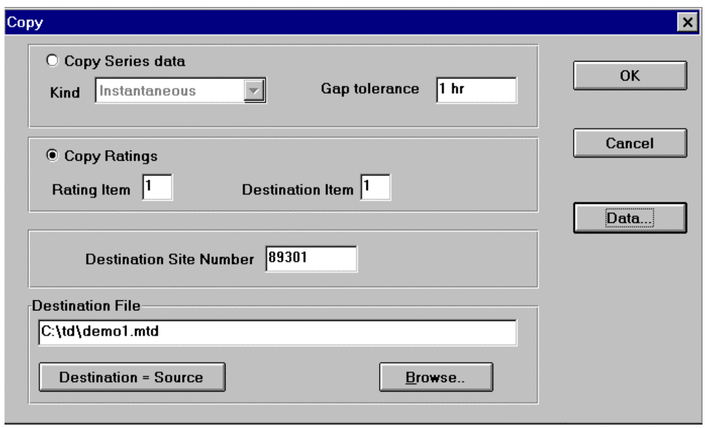


Click **DATA**,
set **Measurement**
stage (Gauging)
Click **OK**
to exit the **DATA** dialog
Click **OK** to **COPY**.

This dialog will appear >
click **YES** to create the Demo1.mtd file >



Click **MOVE**,
click **COPY.**
set this dialog thus >



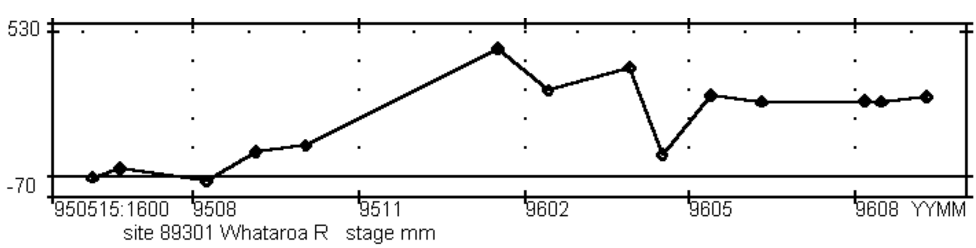
Click **DATA**, set **Measurement** flow (Rated)

Click the **RATINGS** button in the **DATA** dialog on the **From** line, select "950515 160000", click **OK**.

Click the **RATINGS** button in the **DATA** dialog on the **To** line select "950515 160000" again, click **OK**, click **OK** to exit the **DATA** dialog, click **OK** to **COPY**.

Click **FILE**, click **Destination = Source**, click **OK**

Click **GRAPH**, click **SPECIAL**, click **BED PLOT**, set **Min** -70 and **Max** 530, click **OK** to **GRAPH**.

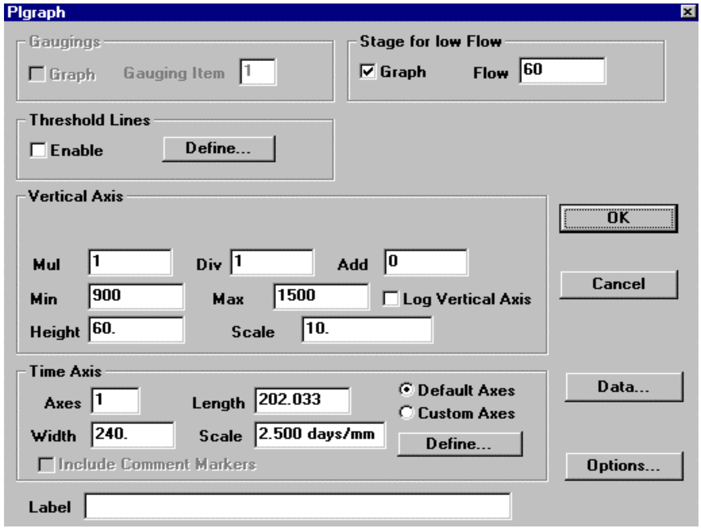


Now we over plot the **Stage for low flow** line that indicates the time and size of rating changes.

Click **FILE**, click **CHANGE INPUT FILE**, select Demo.mtd, click **OK**.

Click **GRAPH**, click **PLOT TYPE**, and click **OVER PLOT**.

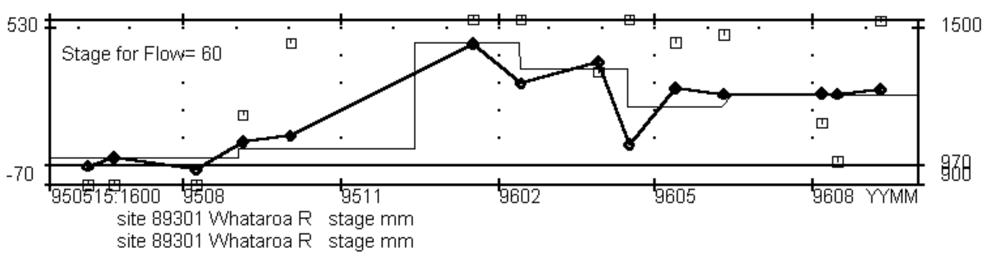
Click **GRAPH**, click **GRAPH OVER TIME** but do not set this dialog yet >



Click **DATA**, set **Measurement** stage (Gauging) click **OK**.

Click **OPTIONS**, set **Number vertical axis** click **OK**.

Set **Stage for low Flow Graph** and **Flow** 60 **Min** 900, **Max** 1500 click **OK** to **GRAPH**.



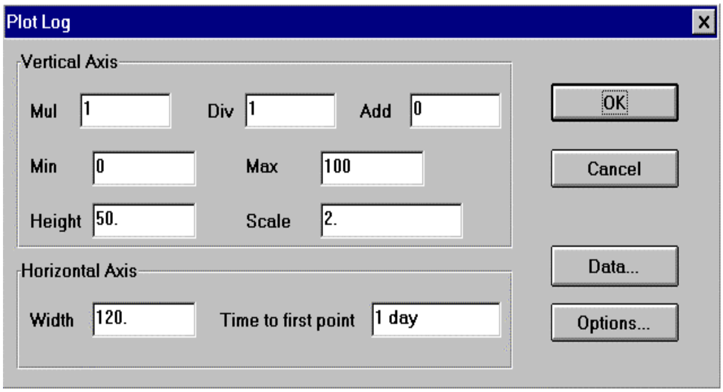
The bold line is the **BED PLOT** and its scale is on the left. The fine line is the **Stage for low Flow**, its scale is on the right, and it is part of the fine line plotted on page 5.6. The box symbols are values of "stage (Gauged)" and their scale is on the right. The rough correspondence between the bed plot and the vertical shifts of the rating changes is the evidence for the rating changes. The writer would replace the 5 rating changes with only 3 changes given the data plotted here.

The last example checked some existing ratings. The process can also be used in the same way when creating a new set of ratings.

LOG OF TIME

This process graphs a time series versus the log of time. With this scaling a recession curve decaying exponentially plots as a straight line. It can be useful for detecting circumstances, (including errors) that affect recessions in time series.

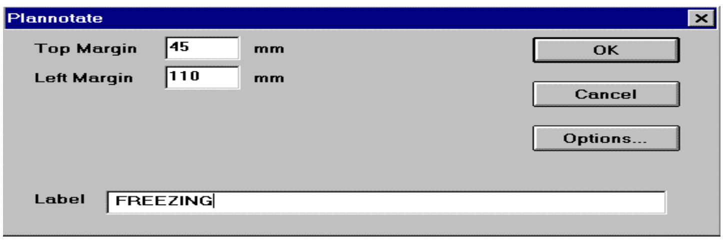
Click **GRAPH**,
click **SPECIAL**,
click **LOG OF TIME**
set this dialog thus >

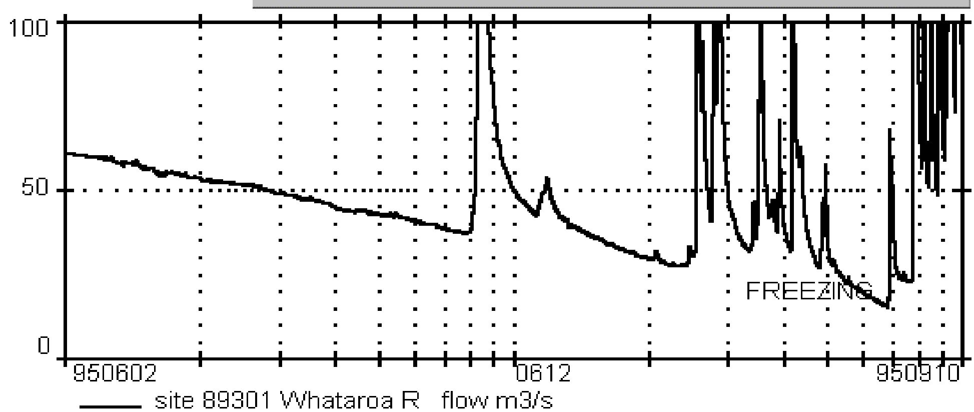


Click **DATA**, set
Site Title
Whataroa R
Measurement
flow (Rated)
From 950601 0
To 950909 0
click **OK** to exit **DATA**

Click **OK** to **GRAPH**

Click **GRAPH**,
click **ANNOTATE**
set this dialog thus >
click **OK** to do it.



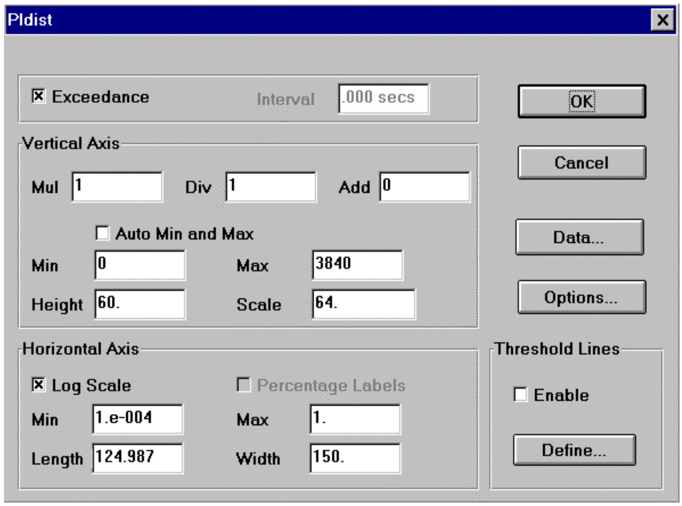


This graph shows a flow recession that becomes much steeper than exponential, probably due to freezing temperatures in the catchment.

DISTRIBUTION

This process graphs the distribution of a series, as the fraction of time that the value is less than (or greater than) the value on the *y*-axis.

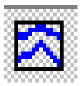
Click **GRAPH**,
click **DISTRIBUTION**
set this dialog thus >



Click **DATA**, set
Site Title Whataroa R
Measurement
flow (Rated)
From 94 1 **To** 95 1
click **OK** to exit **DATA.**

Click **OK** to **GRAPH**

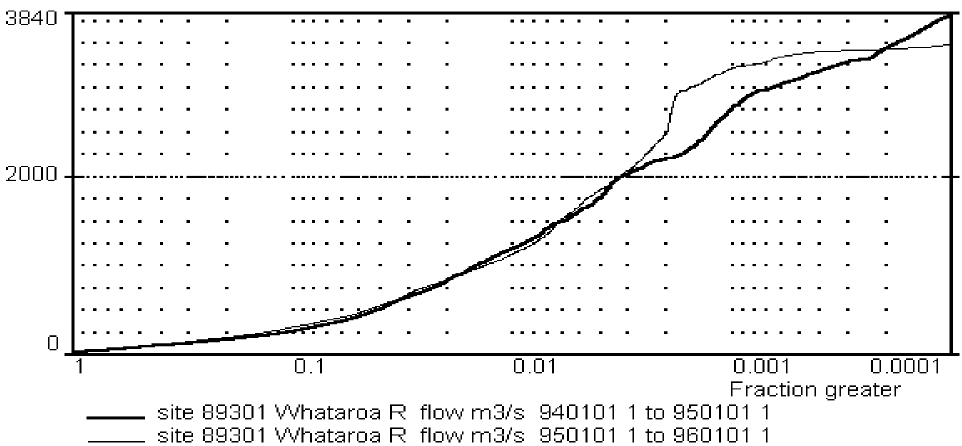
Click his button
to **Over Plot**
>



Click **GRAPH**,
click **DISTRIBUTION**

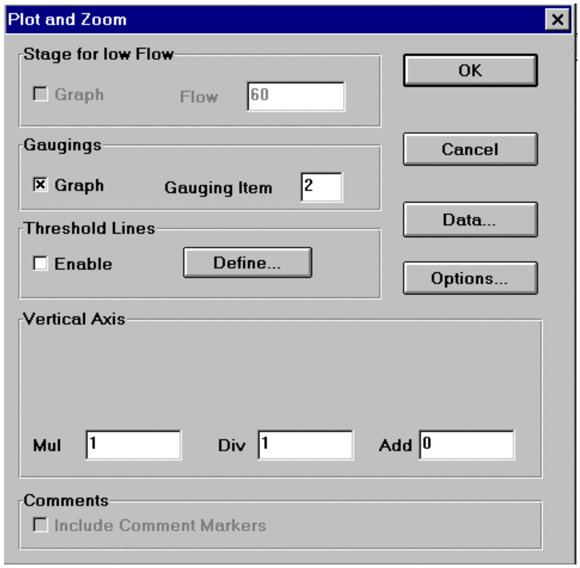
Click **DATA**, set
From 94 1 **To** 95 1
click **OK** to exit **DATA.**

Click **OK** to **GRAPH**



- **Exceedance** makes the graph "fraction greater"; otherwise it is "fraction less".
- **Log Scale** changes the x-axis to be like this example; excellent for highly skewed data.
- Set the **Horizontal Axis Min** and **Max** to larger than 0 and less than 1 respectively to focus on a part of the distribution. In the case of a Log Scale we must set the Min > 0 and usually to a small value like 0.01 or smaller.

GRAPH AND ZOOM

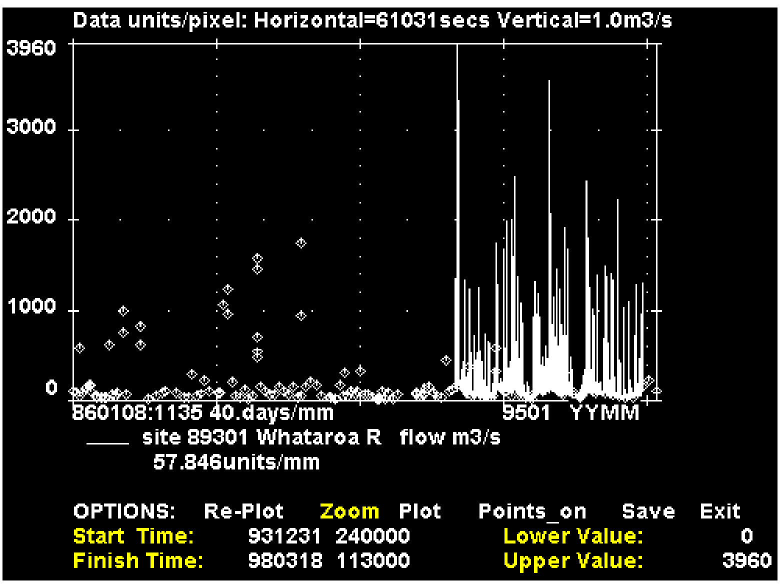


This process is explained in tutorial style on page 2.8.

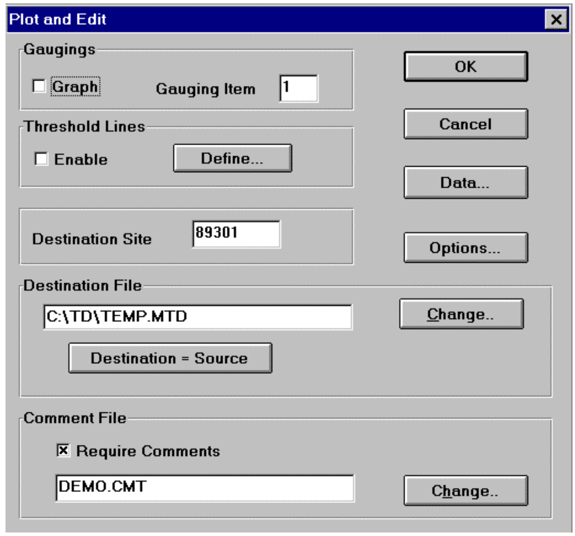
It provides a simple, convenient way to examine a complete time series. We can zoom in on any selected part and examine it more closely. Unlike most other graph processes, there are few options. Every graph is on a New Page, the colours, plot position, data limits, etc., are all preset and the scales automatically adjust to include all the data.

It is often the first process used to view new data.

In the example below we look at data for the Whataroa over the whole time range available. It is immediately evident that the gaugings cover a much longer period than the available continuous series.



GRAPH AND EDIT



This process is described in tutorial style on page 13.15.

It allows us to edit data on our graphics screen and then file the changes back into a Tideda file. It begins exactly the same way as the **GRAPH AND ZOOM** process in that unlike most other graph processes, there are few options, every graph is a New Page, the colours etc. are all preset and scales automatically adjust to include all the data.

It cannot edit a rated series.

After the graph is drawn we have a menu with all of the **GRAPH AND ZOOM** options, plus one additional option, **Edit Data**, which opens other options.

- **Edit Data** invokes a second menu that allows modification of the data points within the current zoom area. Choosing this option automatically turns the data points on. All the data in the plot are then held in program memory and are limited to 1000 data values. However, there is no limit to the number of data points while in the zoom menu. Hence, we can use the zoom option to narrow down the range of data before starting to Edit Data. Revised data are displayed in a different colour from the original plot.
- **Select** enables us to select a data point prior to modifying it. Activate this option, and then move the mouse over the data point of interest. Good office practice dictates that we should modify points based on independent field observations. In this case we will already know the time of the reading we wish to alter. An option is to press <spacebar> and enter the date and time using the keyboard.
- **Delete** removes the selected point from the time series. This option is useful for removing spikes, and faulty readings introduced by electronic dataloggers. Tideda will insert a gap marker around the deleted point, but this can be removed using the Gap option on the menu under Next_Menu (see below).
- **Move** lets us change the time and/or value of the selected data point. We are asked to provide a new date and time; these must keep the point between its current neighbours.
- **Add** adds a new data point at the time and value defined by the mouse cursor. Activate the option, position the mouse cursor where we require a new point then press the left mouse button. To define the point more accurately, press <spacebar>, and we are prompted for the date, time and value.

A point that is added to the data using Add becomes the selected point for the immediately subsequent editing operation.

- **Ramp** corrects systematic errors caused by the recorder clock being fast or slow, or errors caused by siltation in the stilling well. There are dual- or single-ended ramp corrections. In the dual-ended method, the points at the boundary lines are held in their positions while we move a selected point between them. We are asked to provide the new position of the selected point. The remaining data are then linearly adjusted to pass through the new point.

  The single ended-ramp correction is more common. Our field observations will show the date, time and value when we first visited the site, but the recorder was in error at our next visit. Use the Zoom option to set the boundary lines at the times which correspond to our field visits. Press <spacebar> then enter these dates and times from the keyboard, instead of using the mouse. Note that in this case the dates and times must be recorder times, not real times, if it is the recorder times that are filed. Real times will be appropriate after we perform a ramp correction.

  After zooming to the desired boundary lines, choose the Select option and select the point at the right-hand boundary. Now choose the ramp option and enter the new date, time and value. The preceding data will be linearly adjusted across the range, and we have now matched the data to our site visits.

- **Undo** removes the alterations from our screen to correct any mistakes we think we may have made. Choosing undo a second time will reactivate our edits and display them again. Modifications we make to the data are displayed in a different colour and are displayed along with the original data.

- **Exit** finishes our edit session. If we have altered our data we are asked if we wish to Save changes back into the Tideda file. Choosing this option causes a description of the changes we have made to be written into the current Comment file. It then asks for our name and our reasons for changing the data. We must provide this information, and we may include as much detail as we wish. Alternatively we Abandon changes completely and start again, or Resume editing without exiting.

- **Next Menu** invokes a third menu with the following options.

- **Delete all** deletes all data points between the boundary lines.

- **Draw** sets up a freehand draw. As we move the mouse, the data value associated with an existing reading is moved to where the mouse cursor is. The draw facility does not add extra points and the times of the data points cannot be altered.

  When we select the draw option, we will be prompted to specify a minimum move, or tolerance, in the units of the data plotted on the vertical axis. A draw that is within this tolerance will not alter the data point. The tolerance specified must be at least equivalent to the value represented by 1 pixel (which is the default).

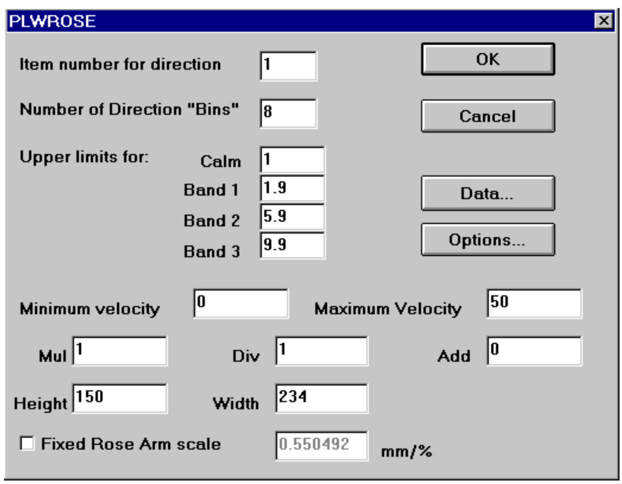
We then position the mouse cursor at the time where we want to start drawing. Press the left mouse button to initiate a draw. After this, as we move the mouse, data points will be "pulled" onto the value defined by the position of the mouse cursor as the cursor reaches the time of each data point. We will be able to see the new position of each data point as we trace the mouse cursor across the plot. Pressing the left mouse button again completes a draw. The original line is drawn in a different colour and a new data line is added through the revised points. We can then move the cursor to a new position and repeat this process as required. When all drawing is complete, press the right mouse button or <esc> to return to the **Edit** menu.
(Note: The **Draw** option is not supported in Tideda for Windows 16-bit versions.)

- **Undo** is the same here as in the **Edit** menu.
- **Exit** is the same here as in the **Edit** menu..
- **Previous Menu** takes us back to the **Edit** menu.
- **Gap** invokes a fourth menu which lets us remove or add gap markers. A gap appears when we delete a point, but we can allow interpolation, and will do so if the deleted point was a spike in the data. If we write our data back to the same Tideda file with the same Site number, deletions are effected by writing a deleting batch in the same way as process **DELETE**. Thus if we subsequently wish to restore the deleted data we can do this with process **BATCH**.

WIND ROSE

This process plots a two-way distribution of two items from a multi-item series, one of which is a polar coordinate. It is implemented specifically for wind series with compass degrees as the polar coordinate and specified using the **DATA** dialog, and the other item is velocity.

Click **GRAPH**, click **SPECIAL**, click **WIND ROSE** to get this dialog.



Item number for direction specifies the item number of the wind direction. The direction must be stored in Tideda as degrees from north (in the range 0–360 degrees).

Number of Direction Bins specifies the number of equal sectors in the wind rose compass, that is, the number of “arms” on the wind rose. The top arm is centred on north and the rest are equally spaced from there. The direction in which the rose arm is pointed is the compass direction that the wind is blowing from, so the top arm represents wind blowing from the north.

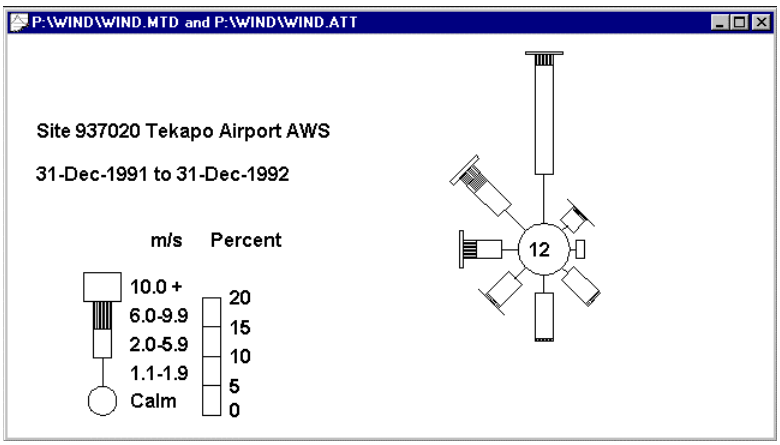
Upper Limits specify the maximum value for each of the first four velocity bins with the top velocity bin being open ended (see the "Maximum Velocity" option below). The fraction of time that the velocity is within a bins range is accumulated in that bin.

Minimum Velocity and **Maximum Velocity** specify the range of valid velocities. The time velocity is outside this range is treated as a gap in the record.

Mul, Div and **Add** are applied to the velocity values.

Height and Width specify the dimensions of the plot frame that the wind rose will be plotted within.

Fixed Rose Arm scale enables specification of a particular scale for the rose arms in units of millimetres per percent of rose arm length, otherwise the scale is automatically set, after the arm percentages are known, to fit the longest arm inside the plot area. For a given plot frame, the centre of the rose is always in the same place.



The number in the "calm circle" at the centre of the wind rose is the percentage of time that the velocity doesn't exceed the calm velocity limit.

The length of each segment in the rose arm represents the percentage of time that the wind velocity was within the velocity range specified on the velocity key and from the direction indicated by the rose arm. The percentage represented by the length of each segment is shown on the "Percent" key.

A tabulation is printed in the File text window. It is the same as printed by the Table menu's process Wind.

```
Source is WIND.MTD Site 937020 Tekapo Airport
From  911231 240000 to  921231 240000

Values are percentages.
Directions in degrees.
Velocity units are                       : m/s
Number of Direction bins used            :       8
Number of Velocity bins used             :       4
Number of Data points read               :   25812
Number of Velocities outside limits      :       0
Number of Directions <0.0 or >360.0 deg. :       1
Number of Data points used               :   25806
Percentage of "calms" recorded           :    11.6
Limits of valid Velocities:    .0 to 50.0 m/s

               1.1-1.9      2.0-5.9      6.0-9.9      10.0 +    Total

337.5- 22.4      8.3         18.6          1.8          .4       29.1
 22.5- 67.4      1.2          2.7           .5          .1        4.5
 67.5-112.4      1.0          1.5           .0          .0        2.5
112.5-157.4      1.3          5.9           .4          .0        7.6
157.5-202.4      3.1          7.7           .4          .0       11.3
202.5-247.4      2.6          5.1           .5          .0        8.3
247.5-292.4      2.9          4.2          2.2          .8       10.0
292.5-337.4      5.0          5.7          3.6          .8       15.1

      Total     25.4         51.5          9.4         2.1       88.4
```

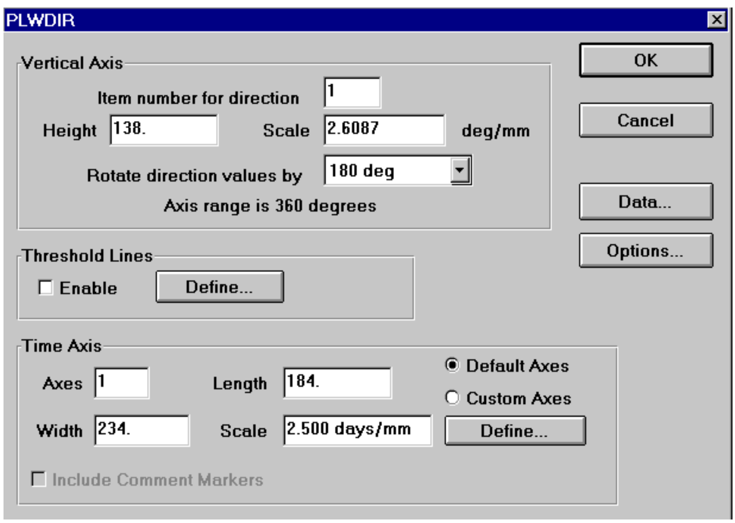
DIRECTION OVER TIME

This process graphs a polar coordinate as a time series. It is implemented specifically for wind series with compass degrees as the polar coordinate

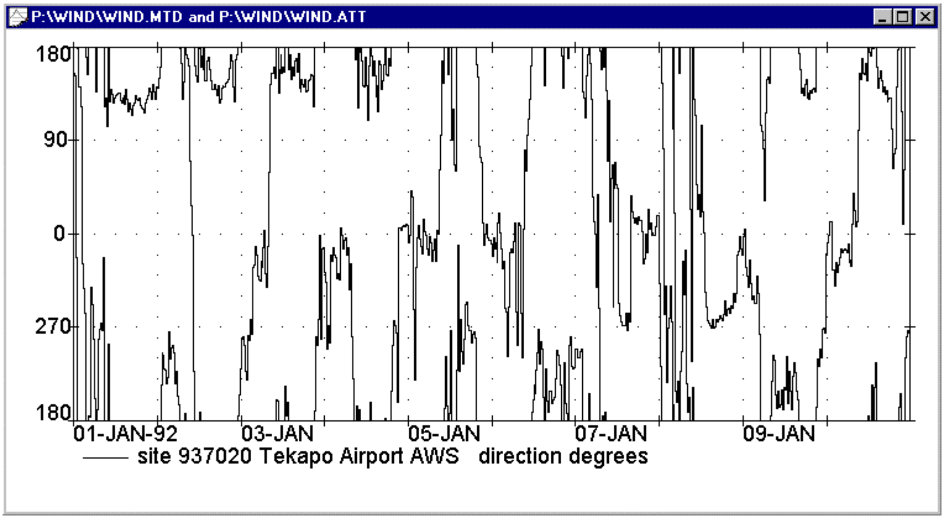
Click **GRAPH**, click **SPECIAL**, click **DIRECTION OVER TIME** to get this dialog.

Many options are the same as in the **GRAPH OVER TIME** dialog; see page 5.2.

The y-axis range is always 360 degrees. When the plotted trace scrolls off one edge of the graph it reappears at the other edge.



Rotate direction values by allows the origin of the vertical axis to be set to any of the four main compass points. Setting this to 180 degrees makes the axis start and finish at 180 degrees with the zero in the middle of the axis.



DOUBLE MASS CURVE

This process graphs two series in ways that compare their cumulative sums. If their fluctuations are nearly coincident in time and the ratios of the magnitudes of the fluctuations are nearly constant, the graph is nearly straight. Thus when two records measure fluctuations caused by the same weather, their double mass curve tends to be straight. In this situation, the time of failure of one of the two sets of measuring equipment is indicated where an otherwise straight line suddenly bends.

The process offers three alternative styles for the graph. Only the Cusum style is described below. A cusum is

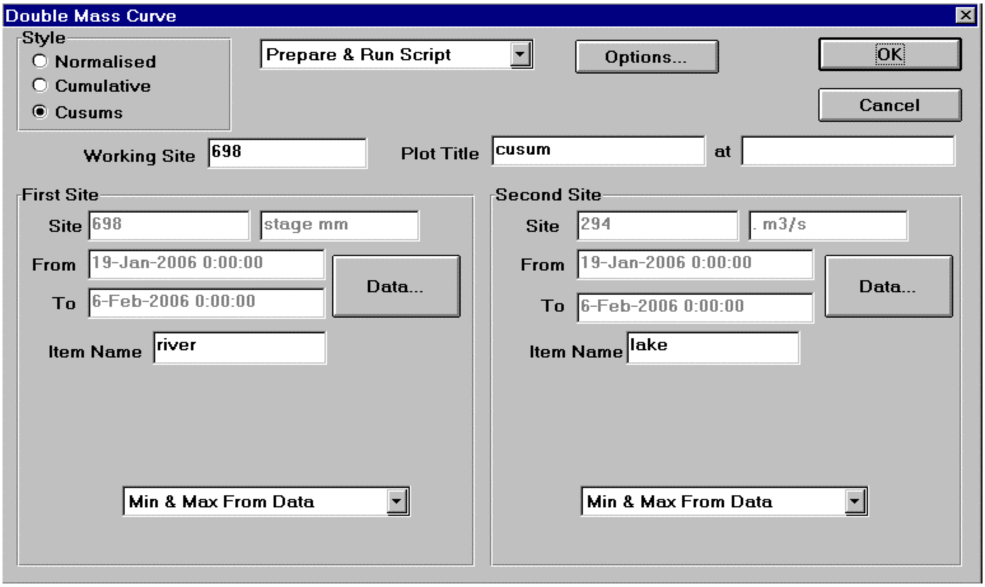
the cumulative sum of the deviation from the mean.

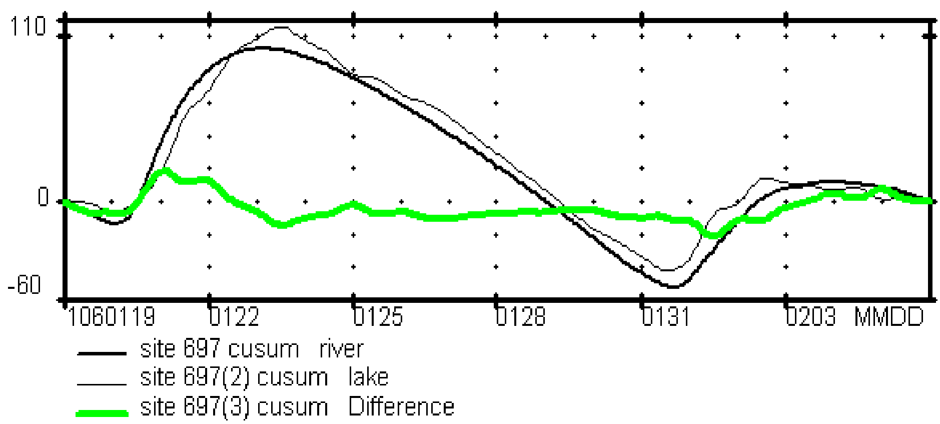
The Cusum style differs from the other two styles, which graph a sloping line, and has the following advantages:

1. the deviations are graphed relative to the x-axis and so the deviation scale can be varied;
2. the times of deviations are along the x-axis and can be labeled like any other time series.

Click **GRAPH**, click **SPECIAL**, click **DOUBLE MASS CURVE** to get the following dialog in which we select **Style Cusum**.

There are two Data buttons, one for each input series. Different series are set in this dialog to produce the following example.





This example does not expose any errors but is used to construct a forecast formula as explained after page 13-18.

One line for each series is 1000 times the cumulative sum of the difference from the mean divided by the standard deviation.

The third line on the graph is just the difference of the other two and represents the same quantity as the deviations of the other styles of double mass curve from a straight sloping line.

CUSUM METHOD

Process **CUSUM** writes five files into the **Working Directory** specified in the **FILE** menu's **PREFERENCES** dialog. Check that this is a suitable location before proceeding. These files are called Cusum1!.sim, Cusum2!.sim, Dblmass!.att, Dblmass!.mtd, Dblmass!.tsf and we can examine them and see they are what we might have written for ourselves. We can also amend them and rerun them for other purposes.

The two "sim" files contain the follow ing scripts for the **PSIM** process. Four **Variables** output from the first "sim" are input to the second "sim".

```
$$$ CUSUM1!.SIM
   init n 0
   init atot 0
   init btot 0
   init ass 0
   init bss 0
xlock
   get a /gap1
   xget b /gap2
   g=gap1*gap2
   if g eq 0 goto next
   n=n+1
   if n eq 1 goto next
   atot=atot+a
   btot=btot+b
   ass=ass+a*a
   bss=bss+b*b
next:
endloop
   n=n-1
   amean=atot/n
   bmean=btot/n
   armsd=.001*sqr((ass-n*amean*amean)/(n-1))
   brmsd=.001*sqr((bss-n*bmean*bmean)/(n-1))
   outvar 1 amean
   outvar 2 bmean
   outvar 3 armsd
   outvar 4 brmsd
   endprog
```

```
$$$ CUSUM2!.SIM
   init n 0
   init ac 0
   init bc 0
   var 1 amean
   var 2 bmean
   var 3 armsd
   var 4 brmsd
xlock
   get  a /gap1
   xget b /gap2
   g=gap1*gap2
   if g ne 0 goto nogap
   gap
   goto next
nogap:
   n=n+1
   if n eq 1 goto L1
   ac=ac+(a-amean)/armsd
   bc=bc+(b-bmean)/brmsd
L1:
   diff=ac-bc
   put ac bc diff
next:
endloop
```

CUSUM DIMENSION

The dimension of the deviation of a Cusum is the product of dimension of the original series and time. For example if

the original series is: flow m^3/s, sedi.load kg/s, wind vel. m/s or air temp. ^oC,
the deviation is: volume m^3, sedi.mass kg, wind run m and heat input ^oC days

CUSUM UNIT

The unit of the deviation is the product of the standard deviation of the original series in file units times 0.001 and times the **Retrieval Interval**. Thus the y-axis unit is the product:

0.001 x standard_deviation x **Retrieval_interval**.

For example if the 0.001 x standard deviation of the wind velocity is 3.3 m/s and we specify **Retrieval Interval** = 300 s, then the deviation is wind run in km.

We can find out the means and standard deviations as follows.
Click **MOVE**, click **PSIM** and look in the dialog to see the following 4 **Variables** calculated by CUSUM1!.SIM:

The first two are the means in file units and the last two are 0.001 x the standard deviations in file units. The **Retrieval Interval** is set in the TSF file (and should be on the process dialog as well) and in this case was 1 hour.

CIRCULAR CHART

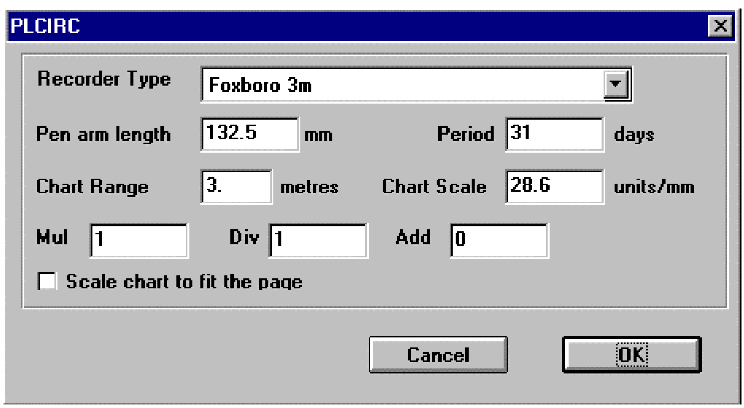
This process plots a series that was originally recorded on a circular chart as it appeared on that chart. It is used for quality control of data digitised from such charts.

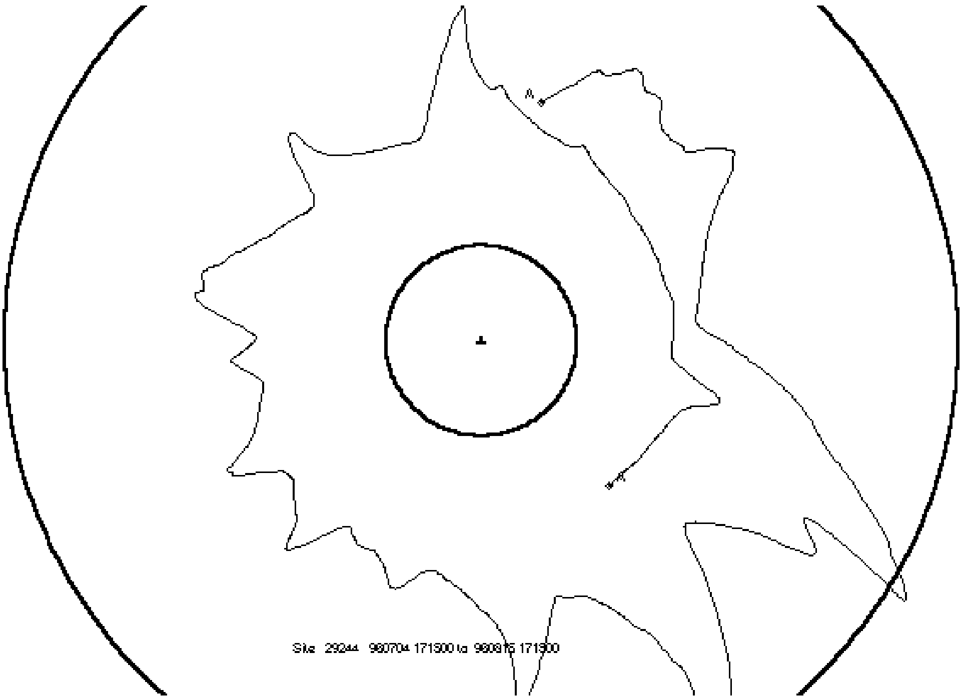
We must have already specified the mechanism of the particular **Recorder Type** in the Chart.def file using the **ENTRY** menu's **CIRCULAR CHART DEFINITIONS** dialog.

We must also specify the time series before running this process. The following graph is a plot of data from the Demo.mtd file selected thus.
Click **DATA**, set **Site** 29244, **From** 960704 171500 **To** 960815 171500, click **OK**.
This is the stage record digitised from the circular chart illustrated on page 8.19

Click **GRAPH**, click **SPECIAL**, click **CIRCULAR CHART** to get this dialog >
Select a **Recorder Type** and its dimensions appear in the dialog, copied from the Chart.def file.

Click **OK** to get the following graph.





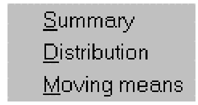
If this graph is full size it can be over laid on the original chart to check how well the digitising was done. To get a full size graph it may be necessary to print it on an A4 page and then enlarge it using a photocopier.

This plot was reduced so it could be printed on an A4 page by specifying a 3-m range (the chart range was actually 1.5 m), and then reduced still further fit this page.

CHAPTER 6 TABLE MENU

This menu's processes extract statistics from data and print them in the File text window in a range of formats. Click **TABLE** in the program banner to get a list of processes, sub-menus and a options. We describe the processes in this order.

STATISTICS >

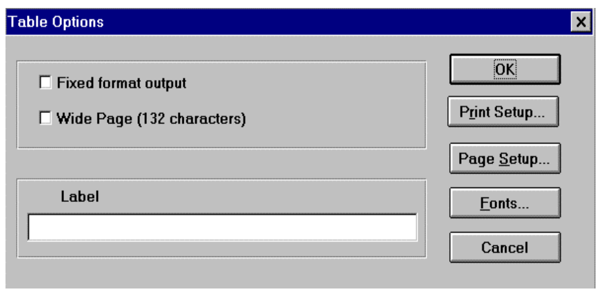


CALENDAR >

OTHER >

OPTIONS

Click **TABLE**, click **OPTIONS** to get this dialog >.



Fixed format output

suppresses automatic scaling and uses the presentation format from the Attribute file. Values that exceed the field width appear as ****. Normally automatic scaling makes the largest value fit the field width. If applied, the scaling mask is shown in the heading, e.g., Read table XXXX00.

Wide Page lengthens lines from 80 characters to 132 characters, and the layout has more characters per value and/or more values per line.

The **PRINT SETUP**, **PAGE SETUP** and **FONTS** buttons open dialogs where we set properties of the File text window.

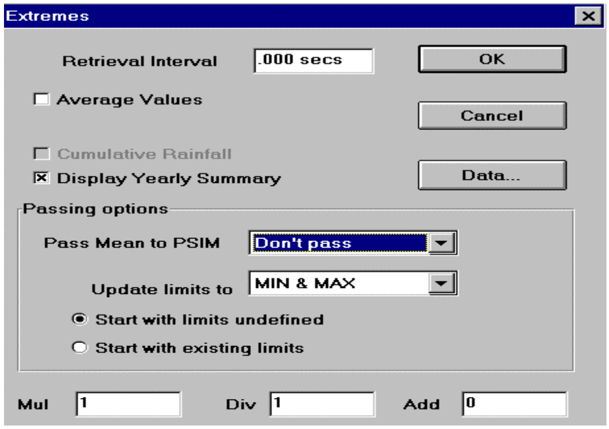
Other options that apply to all the **TABLE** processes, and are set in the process dialogs include:

- **Mul, Div** and **Add** scale values before any automatic scaling, and after **Rating**, as follows:

  $$\textit{Tabulated value} = \textit{Presentation value} \times \textit{Mul} / \textit{Div} + \textit{Add}$$

- Gaps in the filed record are indicated in tables by a question mark "?".

QUICK EXTREMES



This process finds the series maximum and minimum values for use by another process, such as a graph or a **PSIM** calculation. It also calculates the mean and standard deviation.

It is quick compared to process **MOVING MEANS** which also finds the maximum and minimum values, because it only looks at filed values or the time partition that starts at the **From** time. When **Retrieval Interval** > 0 and accuracy is important we must use the slower process.

Display Yearly Summary prints the extreme values in each calendar year.

Update limits to **Min** & **Max** or **Hmin & Hmax** is used to automatically set the axis range for processes such as **GRAPH OVER TIME** and **SCATTER PLOT**.

Pass Mean to PSIM does this to either Var 1, Var 2, Var 3 or Var 4 for use in process **PSIM**.

When applied to the Whataroa R flow record in Demo.mtd the report is as follows:

```
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R.
From  930118 180000 to  971203  71500
Interval =          0        flow m3/s
                 Coeff.
 Year    Mean    of Var. Minimum       Date        Maximum       Date
 1993   No values for this year
*1994   148.42     1.54   33.272   940823 183000   3952.3   940109 161500
 1995   162.95     1.49   15.344   950808  70000   3557.3   951213  93000
 1996   129.64     1.25   23.849   960802  61500   2442.0   961001 191500
*1997   105.79     1.40   20.195   970709  14500   2231.4   970520 194500
Average Annual Minimum    19.597         Maximum    2999.6  (complete yrs)
'*' denotes years with gaps in the data or incomplete years
Coeff. of Var. = sd/mean
Minimum is          15.3440     at  950808  70000
Maximum is          3952.35     at  940109 161500
Mean is             137.192
Std. Dev. is        201.073
Coeff. of Var. is      1.47
```

Gaps are excluded from the calculation of means and standard deviations.

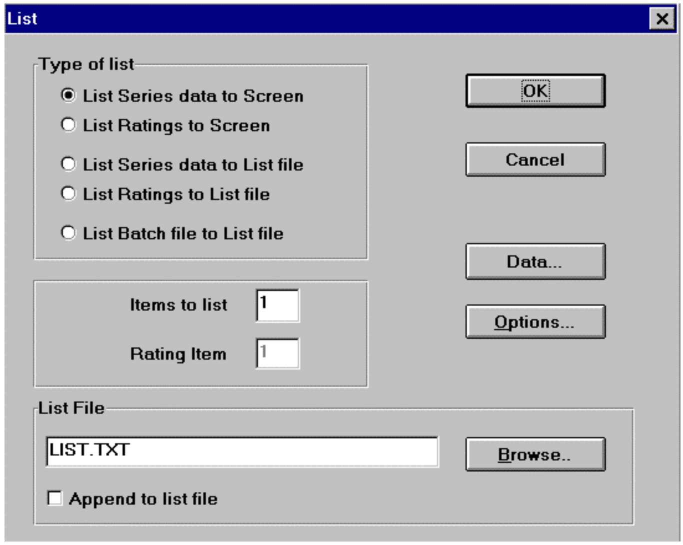
<u>With Increment data</u>:

option **Average Values** is replaced by **Total in Interval** and made compulsory;

option **Cumulative Rainfall** becomes available.

- If we select **Cumulative Rainfall**, **Min** is zero and **Max** is the sum of all the increments
- otherwise if Retrieval Interval > 0, **Min** and **Max** are selected totals interpolated on an equal step time time partition.
- or if Retrieval Interval = 0, **Min** and **Max** are selected from the filed increments

LIST



This process tabulates the values in a Tideda file that represent series or ratings. It does not interpolate.

The table layout is designed for entry back into a Tideda file using the **ENTRY** menu's process **FULLY SPECIFIED**. The contents of the List file may be edited in any way that preserves the essential features of its layout before re-entry.

- **List Series data to Screen**
  This layout has more labels than the next.

```
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R.
1 Item INSTANT From 961001 233000 to 961002 3000
stage        Date         Time
 mm
3620      961001       233000
3578      961001       234500
3546      961001       240000
3484      961002         1500
3452      961002         3000
```

- **List Series to List file**

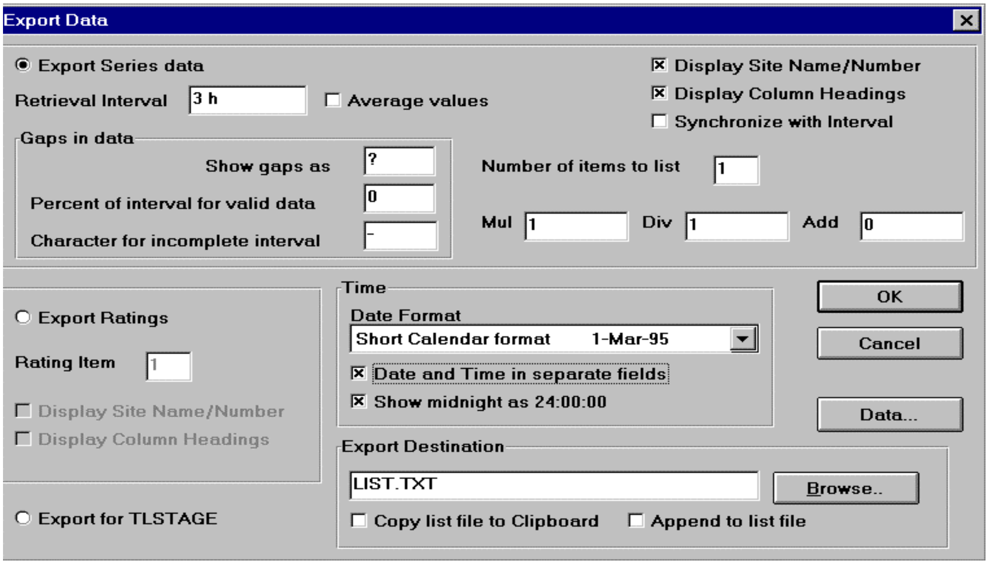
```
89301           1 INSTANT
3620      961001       233000
3578      961001       234500
3546      961001       240000
3484      961002         1500
3452      961002         3000
```

- **List Ratings to List file**

```
89301 RATING 1 961001 180000 961001 210000
 695        18000
 795        24100
 895        32200
 995        43600
1095        58000
1295        98800
1495       149500
1695       207500
1995       309000
2495       517000
2995       778000
3495      1135000
3995      1600000
4495      2175000
```

EXPORT

This process is described in tutorial style starting on page 2.3. It tabulates time series , with comma delimiters, for transfer to another program such as EXCEL.

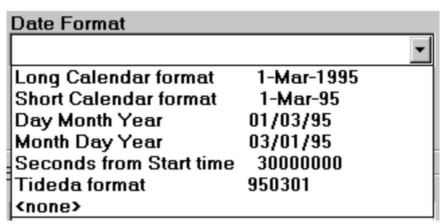


A typical output looks like this >
This file can now be imported into another application for further processing.

```
Site 89301 Whataroa R.
Date,Time,flow m3/s
1-Oct-96,6:00:00,50.41
1-Oct-96,9:00:00,51.42
1-Oct-96,12:00:00,325.2
1-Oct-96,15:00:00,645.1
1-Oct-96,18:00:00,2134.
1-Oct-96,21:00:00,1939.
1-Oct-96,24:00:00,1177.
2-Oct-96,3:00:00,796.7
2-Oct-96,6:00:00,557.1
2-Oct-96,9:00:00,483.6
2-Oct-96,12:00:00,542.2
2-Oct-96,15:00:00,456.9
2-Oct-96,18:00:00,438.7
2-Oct-96,21:00:00,425.8
2-Oct-96,24:00:00,397.0
3-Oct-96,3:00:00,360.0
3-Oct-96,6:00:00,320.1
```

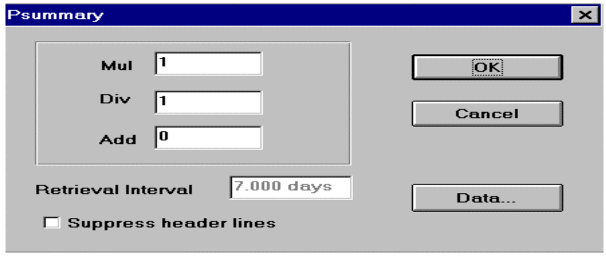
Synchronize with Interval has an effect when the **Retrieval Interval** matches the time step of the filed data, and the specified **From** time is not a file time, and causes the **From** time to be delayed to the next file time.

The **Date Format** list includes <none> which will export values without times >



SUMMARY

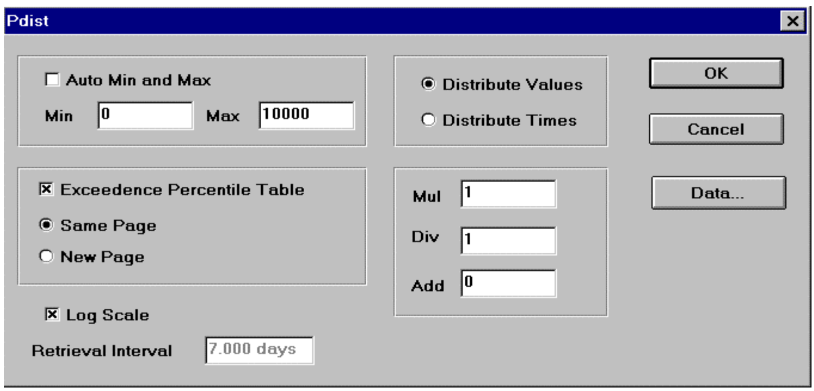
This process prints summary statistics of a selected series.




```
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R.
From   931231 240000 to  971203  71500 flow m3/s
   Site    Minimum    Maximum       Mean  Std.Dev.    Median  Lower Q. Upper Q.
  89301         15       3952        137     201.1        85        50      142
```

DISTRIBUTION

This process tabulates the distribution of a series as the fraction of time values are greater than a specified set of equally spaced values.



The data are grouped into up to 50 classes for a print-plot, 1 class per line. The time in each class is at the left, and represents the probability density.

The class boundaries can be set to round numbers by setting **Max** - **Min** to be a multiple of 50.

The mean and standard deviation are calculated directly from the data in the file, not from the distribution. This avoids errors caused by large class intervals. Values are interpolated at the start and finish times when these do not coincide with filed data. Gaps are excluded from the distribution, as are single values, which have a gap before and after them.

Log scale only affects the print-plot and more effectively displays highly skewed series.

Retrieval Interval must be > 0 with Incremental data.

Distribute Times prints the distribution of time intervals in the file's time partition. It is used for checking records for unmarked gaps, and checking Incremental data where equal value increments are recorded at unequal time intervals.

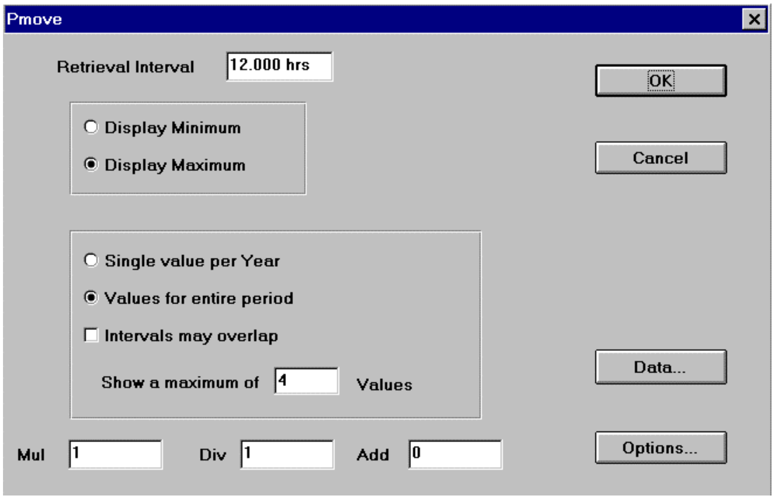
When **Auto Min and Max** is selected, class boundaries are determined as the calculation proceeds, and the number of classes varies according to the data, with a maximum of 50. This feature allows once through processing of sequential files, because Min and Max need not be determined first.

```
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R.
From  931231 240000 to  971203  71500
flow m3/s Distribution computed in the range        0 to     10000

DISTRIBUTION                PERCENTAGE OF TIME "VALUE" IS EQUALLED OR EXCEEDED
 DAYS  HHMMSS  "VALUE"   %  .032     .16       .80       4.0       20        100
                  3952      *MAXIMUM |         |         |         |         |
    0   5235      3800 .003 *        |         |         |         |         |
    0   2141      3600 .004 *        |         |         |         |         |
    0  41757      3400 .016 *        |         |         |         |         |
    0  65837      3200 .037 |*       |         |         |         |         |
    0  64753      3000 .057 |    *   |         |         |         |         |
    0  52130      2800 .073 |     *  |         |         |         |         |
    0  22601      2600 .080 |      * |         |         |         |         |
    0  42316      2400 .093 |       *|         |         |         |         |
    0 101719      2200  .12 |        *         |         |         |         |
    0 184613      2000  .18 |        |*        |         |         |         |
    0 183739      1800  .23 |        | *       |         |         |         |
    1  33030      1600  .31 |        |   *     |         |         |         |
    1 160232      1400  .43 |        |     *   |         |         |         |
    3   4330      1200  .65 |        |        *|         |         |         |
    5 103130      1000  1.0 |        |         | *       |         |         |
    9 155819       800  1.7 |        |         |    *    |         |         |
   16 222330       600  2.9 |        |         |       * |         |         |
   33  22526       400  5.2 |        |         |         |   *     |         |
  135  93952       200  15. |        |         |         |       * |         |
 1207  43410        15 100. |MINIMUM |         |         |         |         *
   15 101500=GAPS
 1432  71500=TOTAL          MEAN=   137.192            STD.DEV.=   201.073
This table uses   20 classes.
Percentages in the printplot table are accurate only to the
precision shown (e.g. 5.0% is 5.0% +/- 0.05%)

Exceedance percentiles
         0      1      2      3      4      5      6      7      8      9
  0   3952   1013    739    589    474    412    365    330    302    278
 10    258    242    229    218    208    198    190    183    176    171
 20    165    160    155    150    146    142    138    135    132    129
 30    126    124    121    119    117    115    113    110    108    106
 40    104    102    100     98     96     94     92     90     88     86
 50     85     83     81     79     78     76     75     73     72     71
 60     69     68     66     65     63     62     61     60     58     57
 70     56     55     54     52     51     50     49     48     47     45
 80     44     43     42     40     39     38     37     36     35     34
 90     32     31     30     29     28     27     26     25     23     21
100     15
Values in the exceedance table are not exact. They are good
approximations based on linear interpolation of 2000 classes.
```

MOVING MEANS

This process prints the maxima or minima in a time series calculated by using a moving average, (or moving total for Incremental data).

With this dialog setting and the data tabulated on pages 2.7 & 2.8 the report is:



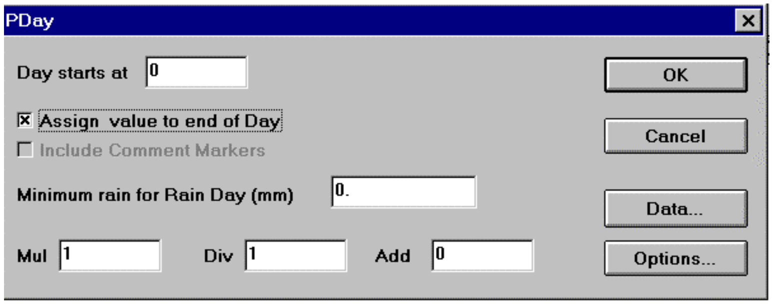

```
Source is C:\TD\DEMO.MTD Site 89301 Whataroa R
From  931231 240000 to  971203  71500

Maximum moving averages over     12.000hrs  interval (Non-Overlapping)

 Gap from  940722  21500 to 940727 124500 of    5.44 Days
 Gap from  940824 190000 to 940901 134500 of    7.78 Days
 Gap from  941108  71500 to 941110 121500 of    2.21 Days

The   4 largest values are:         (flow m3/s)
    1:      3263.4 at interval beginning  951213  20000
    2:      2673.4 at interval beginning  940109  80000
    3:      2369.7 at interval beginning  940122  34500
    4:      1680.8 at interval beginning  951213 140000
```

This process checks all possible time partitions which process **QUICK EXTREMES** does not.

Single value per Year prints the maxima or minima values from a 12-month time partition, starting at the **From** time we set in the **DATA** dialog. The years will only match a calendar year if we set **From** to time zero on the first of January.

DAILY

This process tabulates daily values in columns of calendar months, a year to each table. Monthly and annual minima, maxima and means are also tabulated.



The values are 24 hour means of Instant and Histogram series, and totals of Increment series.

- **Day starts at** sets a time of day as "hhmmss" e.g., 90000 = 9 am.
- **Assign value to end of Day** changes where values appear in the table except when **Day starts at** 0 or 240000.
- **Minimum rain for Rain Day (mm)** with Increment data, causes a note to be added about the count of days exceeding the value set**.**

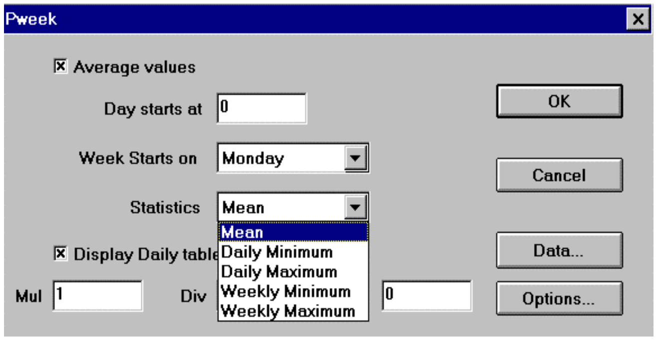
```
24 hour periods ending at midnight each day.
Daily means    Year 1996    site 89301 Whataroa R.
flow m3/s
```

Day	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	
1	148	86	67	53	140	112	35	24	26	723	112	165	
2	127	107	662	76	109	62	32	26	26	548	92	167	
3	110	291	266	323	90	55	30	40	33	280	81	128	
4	127	305	133	307	78	51	29	38	220	174	74	105	
5	128	166	94	176	70	58	28	48	159	120	70	92	
6	118	166	75	109	64	128	28	53	133	130	78	81	
7	111	147	66	87	61	65	27	36	87	666	93	75	
8	110	117	61	77	470	96	27	32	127	284	74	75	
9	106	91	56	320	149	166	27	30	149	152	65	75	
10	100	77	55	185	103	120	27	28	140	112	60	75	
11	105	135	51	132	86	71	26	27	84	92	90	138	
12	114	860	48	328	222	58	26	27	68	179	113	94	
13	581	245	47	382	105	51	26	26	107	591	162	77	
14	275	155	46	743	81	45	25	26	139	196	96	73	
15	190	113	105	822	69	43	26	30	139	130	172	71	
16	152	92	62	300	106	50	26	28	78	97	147	64	
- -	- -	- -	- -	- -	- -	- -	- -	- -	- -	- -	- -	- -	
26	121	56	75	67	120	32	28	31	49	123	121	121	
27	110	56	69	62	82	33	27	33	106	109	331	144	
28	105	55	64	345	68	33	26	34	115	229	139	154	
29	102	56	60	430	63	32	26	30	70	264	167	136	
30	96		57	217	66	48	25	28	55	164	126	128	
31	90		55		101		25	27		149		132	
Min	90	55	46	53	54	32	25	24	26	84	60	64	24
Mean	158	146	131	236	111	58	28	33	91	287	139	137	130
Max	581	860	662	822	470	166	41	69	220	858	622	423	860

WEEKLY

This process tabulates daily values in a calendar format, in which weeks are arranged in columns.

A summary of weekly, seasonal and annual statistics in a one-year-per-line format, follows the main data tables



- **Day starts at** sets a time of day as "hhmmss" e.g., 90000 = 9 am. When this time is noon or later, values are tabulated in the following day.
- **Compute Average values** gets 24 hour means (or totals when data is Increment). Otherwise we get instant values interpolated at the same time each day (except with Increment data when we get nonsense).

The following examples illustrate all the options. Select **Display Daily table** and **Compute Average values** to get the following layout of the numbers in the process **DAILY** table above.

```
Site 89301 Whataroa R. flow m3/s
Means from 0 hours. Min daily mean 26. Min weekly mean 29
Mean of 13 weeks 165. Max daily mean 858. Max weekly mean 454
Spring 1996
         Sep.    ( 91)          Oct.(287)           Nov.(139)
Mon. 31      26 149  78  49  55     666 196 858 229      74  90  92
Tue. 33      33 140  66  49     723 284 130 758 264      70 113 622
Wed. 34     220  84 151  48     548 152  97 284 164      78 162 239
Thu. 30     159  68  93  49     280 112  84 189 149      93  96 192
Fri. 28     133 107  67 106     174  92 121 147     112  74 172 132
Sat. 27      87 139  58 115     120 179 361 123      92  65 147 106
Sun.     26 127 139  53  70     130 591 581 109      81  60 109 109
Week     30 112 118  81  69     290 297 225 353     156  74 127 213
```

If we deselect **Compute Average values** we get values at the time set in the box **Day starts at** (ie. midnight in the morning when this time = 0).

```
Site 89301 Whataroa R. flow m3/s                Read Table: XXX0
Values at 0 hours. Min value 3. Min weekly mean of values 3
Mean of 13 weeks 16. Max value 118. Max weekly mean of values 45
Spring 1996
         Sep.    (  9)          Oct.( 28)           Nov.( 15)
Mon.  3       3  13   9   5   6      17  26  81  11       8   6  10
Tue.  3       3  17   7   5       5  52  16  88  25       7  12  30
Wed.  4      11  12   6   5     118  19  11  38  20       7  10  38
Thu.  3      16   7  13   5      40  13   9  21  14      12  12  18
Fri.  3      15   7   7   5      21  10  10  17      13   8  12  16
Sat.  3      10  15   6  13      14  11  28  13      10   7  13  12
Sun.      3   9  13   6   9      11  59  41  11       9   6  14  10
Week      3   9  12   8   7      31  26  20  39      15   8  11  19
```

These tables illustrate significantly different values due to changes in the interpolation method. Note that the scaling has been changed in the second table and is stated at the top right:
`Read Table: XXX0`. To get actual values we multiply values in the table by 10.

Two of the values underlined in October (5 x 10 and 14 x 10) correspond as closely as the scaling will allow to the first hour of the day values (51 and 140) in the table of hourly means on page 6.11. However the value in between (118 x 10) differs significantly from the hourly means either side of midnight (1250 and 1091) because at that point the graph is much steeper.

- **Statistic** determines what is summarised in the one-year-per-line table at the end. These options are illustrated below. Note that the seasonal statistic in each case is just one of the numbers reported in the headings of the tables above which detail all the days in a season.

Table of seasonal and annual min. values at 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	93?	5	?	?	?	?	?	?	93?	4
1994/5	35?	4	35	2	35?	4	35?	3	67	3
1995/6	15	1	50	4	15	1	46	4	58	2
1996/7	24	3	47	3	24	3	26	1	57	1
1997/8	20?	2	29	1	20	2	29	2	109?	5

Table of seasonal and annual min. daily means from 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	91?	5	?	?	?	?	?	?	91?	4
1994/5	35?	4	35	2	35?	4	35?	3	66	3
1995/6	16	1	50	4	16	1	45	4	57	1
1996/7	24	3	46	3	24	3	26	1	59	2
1997/8	20?	2	28	1	20	2	28	2	117?	5

Table of seasonal and annual min. weekly means from 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	117?	5	?	?	?	?	?	?	117?	4
1994/5	36?	4	41	2	44?	4	36?	3	78	2
1995/6	17	1	55	4	17	1	56	4	94	3
1996/7	26	3	52	3	26	3	29	1	70	1
1997/8	21?	2	34	1	21	2	32	2	137?	5

Table of seasonal and annual means from 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	295?	5	?	?	?	?	?	?	295?	5
1994/5	129?	3	120	1	90?	4	138?	2	163	2
1995/6	160	4	185	4	51	2	161	3	241	4
1996/7	124	2	159	3	42	1	165	4	129	1
1997/8	102?	1	124	2	68	3	102	1	229?	3
Period	162?		147?		63?		141?		211?	

Table of seasonal and annual max. weekly means from 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	774?	2	?	?	?	?	?	?	774?	2
1994/5	428?	4	308	4	180?	2	428?	2	376	3
1995/6	814	1	479	1	141	3	402	3	814	1
1996/7	454	3	417	2	130	4	454	1	328	4
1997/8	353?	5	353	3	267	1	265	4	253?	5

Table of seasonal and annual max. daily means from 0 hours.

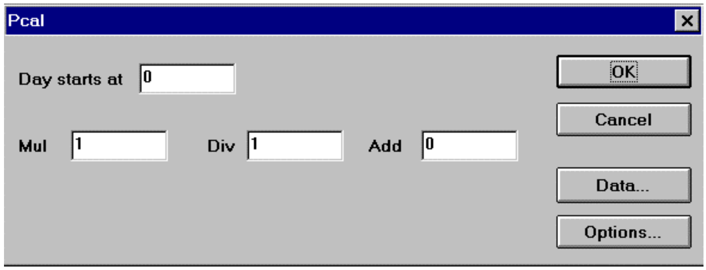
Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	2283?	2	?	?	?	?	?	?	2283?	2
1994/5	1216?	3	686	4	603?	1	1216?	1	941	3
1995/6	2672	1	990	2	430	3	858	2	2672	1
1996/7	858	5	822	3	166	4	858	3	789	4
1997/8	1024?	4	1024	1	550	2	561	4	515?	5

Table of seasonal and annual max. values at 0 hours.

Year	Annual	rank	Autumn	rank	Winter	rank	Spring	rank	Summer	rank
1993/4	2007?	2	?	?	?	?	?	?	2007?	2
1994/5	1213?	5	689	4	403?	3	1213?	1	1030	4
1995/6	2886	1	1206	2	423	2	1097	4	2886	1
1996/7	1458	3	753	3	308	4	1177	2	1458	3
1997/8	1343?	4	1343	1	1004	1	1163	3	441?	5

MONTHLY

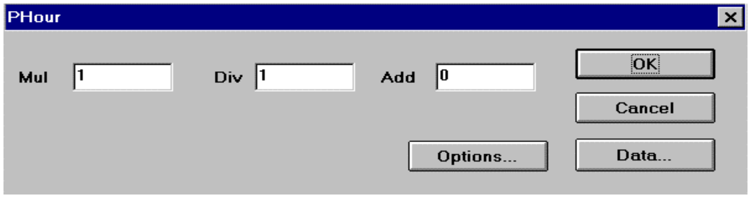
This process tabulates monthly values, a year to a line, with monthly and annual statistics.



For Instantaneous and Histogram data the values are monthly means. For Incremental data they are monthly totals. Where there is gap in the data during a month the monthly value is replaced by a question mark "?".

```
Source is C:\TD\DEMO.MTD
Monthly means   1994 to 1997    site 89301 Whataroa R.

flow m3/s

     Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec   Mean

     1994  436  131  138  116  111  107    ?    ?    ?   56    ?  132    154?

     1995  204  150  236  200  118   48   39   67  224  140  125  401    163
     1996  158  146  131  236  111   58   28   33   91  287  139  137    130
     1997   87  178   85  165   99   48   54   99   48  106  187    ?    104?

     Min.  87  131   85  116   99   48   28   33   48   56  125  132    130
     Mean  221  151  148  179  110   65   40   67  121  147  150  223    146
     Max. 436  178  236  236  118  107   54   99  224  287  187  401    163
           The Min Mean and Max of Annual means are for complete years only.
```

HOURLY

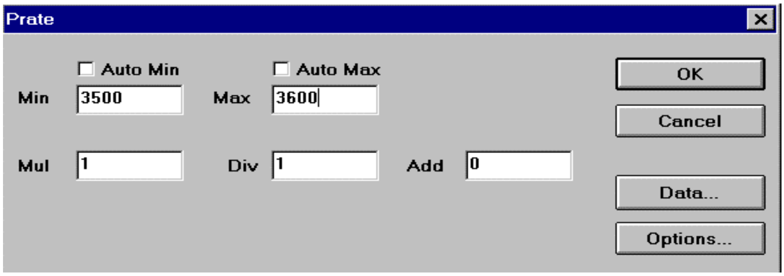


This process makes tables of hourly values for each day, a calendar month to each table.

Daily and time-of-day statistics (minima, maxima and means) are also tabulated. The values are hourly means, or totals for Increment data. With 80 character lines there are 2 lines per day, and with 132 characters 1 line per day.

```
Hourly Means for    October   1996    site 89301 Whataroa R.     flow m3/s

    Hour   1    2    3    4    5    6    7    8    9   10   11   12    Mean
Day
 1   am   51   51   51   51   51   50   50   50   51   54   69  213
     pm  417  549  615  753 1141 1864 2300 2354 2113 1756 1453 1250     723
 2   am 1091  944  851  743  653  587  540  502  498  477  491  511
     pm  508  476  454  452  442  437  433  421  419  416  411  395     548

-  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

31   am  140  140  141  143  144  145  148  160  181  185  173  163
     pm  159  155  149  147  144  143  142  139  137  135  131  129     149

Mean am  281  280  284  283  280  279  273  269  264  260  256  257
Mean pm  269  283  294  300  310  326  331  325  312  298  290  285     287
```

RATING

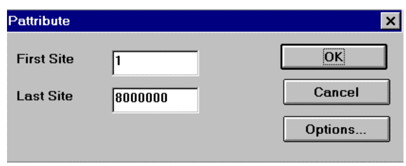


This process tabulates a rating with closely spaced interpolated values for manual use.

Min and **Max**
set the range of unrated values covered by the rating table.

```
Site 89301 Whataroa R.
Rating at   961001 180000  Overlap until   961001 210000
             0   1   2   3   4   5   6   7   8   9
                  Multiply following by          10
  3500    114 114 114 114 114 114 114 114 115 115
  3510    115 115 115 115 115 115 115 115 115 115
  3520    116 116 116 116 116 116 116 116 116 116
  3530    116 116 117 117 117 117 117 117 117 117
  3540    117 117 117 117 118 118 118 118 118 118
  3550    118 118 118 118 118 119 119 119 119 119
  3560    119 119 119 119 119 119 119 120 120 120
  3570    120 120 120 120 120 120 120 120 120 121
  3580    121 121 121 121 121 121 121 121 121 121
  3590    122 122 122 122 122 122 122 122 122 122
  3600    122
ARGUMENT
3600                                                                 *
3590                                                           *
3580                                                        *
3570                                                   *
3560                                            *
3550                                       *
3540                                *
3530                           *
3520                     *
3510               *
3500   *
  1139                                                        1224
                                         ENTRY
```

ATTRIBUTES



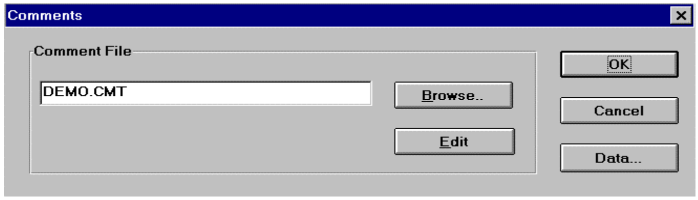
This process lists the **Site Number**, **Site Title, Kind** and Rating unit of the attributes in the attached Attribute file in the specified range of sites.

```
Attribute file is C:\TD\DEMO.ATT
    10 Sites in this attribute file

                                                            Rating
Site No.    Site Name                                  Kind    Units
     211 canal in                                    INSTANT
     226 canal out                                  HISTOGRM
     227 spillway                                    INSTANT
     294 lake inflow                                 INSTANT
     295 lake vector                                HISTOGRM
     296 lake gauge - smoothed                      HISTOGRM
     297 lake gauge                                  INSTANT
     697 river - smoothed                            INSTANT
     698 river gauge                                 INSTANT  l/s
     998 rain gauge                                  INCRMNT
```

If we set **Wide Page** in the **PREFERENCES** dialog, the list includes the "Recording Authority", the "Recorder Make", the "Data Begins" and the "Data Ends" fields from each record.

Refer to the index for pages with more information on attributes.

COMMENTS



This process prints comments from the attached Comment file for the **Site, From** and **To** times set in the **DATA** dialog.

Refer to the index for pages with more information on comments. The following are two comments.

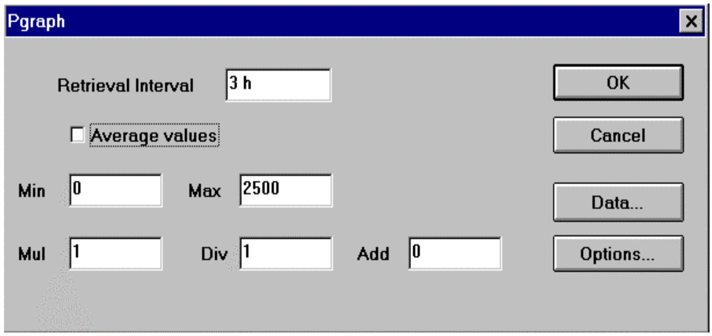
```
Contents of DEMO.CMT:

@@ 65104 870403 140000
New rating curve because of large flood

@@ 68526 880503 100000
Recorder housing flushed Little sediment
```

The comment file is a text file. The start of a comment is indicated by two @ signs in columns 1 and 2, followed by the site, date and time of the comment. These parameters may appear anywhere on the initial line.

The lines of text that follow are the actual comment, and **COMMENTS** will read all the lines until another two @ signs appear, or it reaches the end of the file. **COMMENTS** prints the comments in the order it finds them in the file, and this need not be in time order.

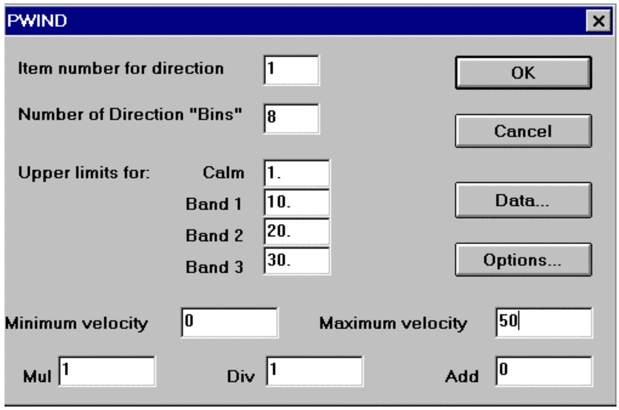
PRINTPLOT

This process prints a time series as a plot on a non-graphic printer or screen. On graphic devices it is usually better to graph the data. Values are normally plotted as asterisks, but when interpolated across gaps are plotted as dots.




```
Site 89301 Whataroa R.
From  961001  60000  to   961003  60000    INTERVAL   10800
flow m3/s
     50. 961001  60000 |*        |         |         |         |         |
     51. 961001  90000 |*        |         |         |         |         |
    325. 961001 120000 |      *  |         |         |         |         |
    645. 961001 150000 |         |  *      |         |         |         |
   2134. 961001 180000 |         |         |         |         |  *      |
   1939. 961001 210000 |         |         |         |        *|         |
   1177. 961001 240000 |         |         |   *     |         |         |
    797. 961002  30000 |         |     *   |         |         |         |
    557. 961002  60000 |         |*        |         |         |         |
    484. 961002  90000 |         *         |         |         |         |
    542. 961002 120000 |         |*        |         |         |         |
    457. 961002 150000 |        *|         |         |         |         |
    439. 961002 180000 |        *|         |         |         |         |
    426. 961002 210000 |        *|         |         |         |         |
    397. 961002 240000 |       * |         |         |         |         |
    360. 961003  30000 |      *  |         |         |         |         |
    320. 961003  60000 |     *   |         |         |         |         |

SUMMARY OF PLOTTED DATA
MEAN    VALUE IS  652.9650
MAXIMUM VALUE IS  2133.785     AT 961001 180000
MINIMUM VALUE IS  50.40600     AT 961001  60000
```

WIND



This process tabulates a two-way distribution of two items from a vector series. At present this is implemented specifically for wind series.

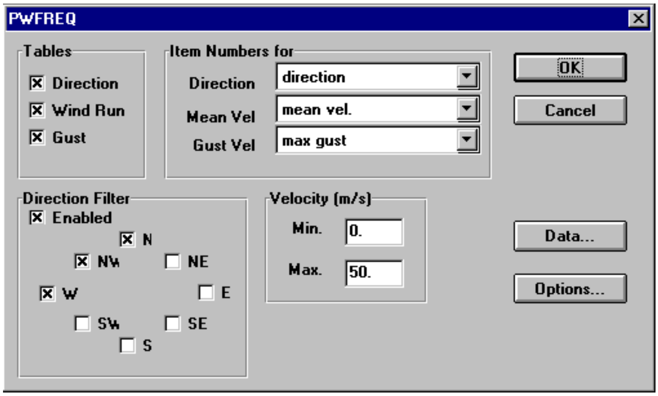
Item number for direction sets the second item in the two-way distribution, the first item having been specified already in the Data dialog (and called velocity). The unit for direction is degree and:

Tabulated direction = Filed direction Modulo 360

```
Source is WIND.MTD Site 937020 Tekapo Airport
From  911231 240000 to  921231 240000

Values are percentages.
Directions in degrees.
Velocity units are                       : m/s
Number of Direction bins used            :       8
Number of Velocity bins used             :       4
Number of Data points read               :   25812
Number of Velocities outside limits      :       0
Number of Directions <0.0 or >360.0 deg., :       1
Number of Data points used               :   25806
Percentage of "calms" recorded           :    11.6
Limits of valid Velocities:    .0 to 50.0 m/s

               1.1-1.9      2.0-5.9      6.0-9.9      10.0 +     Total
337.5- 22.4      8.3         18.6          1.8          .4        29.1
 22.5- 67.4      1.2          2.7           .5          .1         4.5
 67.5-112.4      1.0          1.5           .0          .0         2.5
112.5-157.4      1.3          5.9           .4          .0         7.6
157.5-202.4      3.1          7.7           .4          .0        11.3
202.5-247.4      2.6          5.1           .5          .0         8.3
247.5-292.4      2.9          4.2          2.2          .8        10.0
292.5-337.4      5.0          5.7          3.6          .8        15.1

      Total     25.4         51.5          9.4         2.1        88.4
```

DAILY WIND

This process produces daily tables of Wind direction, Wind run and Gust velocities. Time, Direction and Velocity filters can be applied to produce statistics for specified wind directions or veleocity ranges.




```
Source is WIND.MTD Site 937020 Tekapo Airport
From  911231 240000 to  921231 240000
Direction item is 1 direction
Mean Vel. item is 2 mean vel.
Gust Vel. item is 4 max gust
No Time filter applied
Direction filter includes:  N, W,NW
Velocities in the range    .0 to  50.0 m/s
```

Frequency of wind (percent)

Day	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1	19	21?	54	51	67	76	94	88	94	36	63	25
2	6	100	88	68	67	88	81	96	86	3	93	44
3	40	100	35	60	63	71	93	100	79	29	61	57
4	51	88	51	47	67	68	86	72	53	65	68	31
5	63	25	50	65	39	74	63	81	31	57	50	1
6	44	26	79	67	17	72	85	68	32	68	44	0
7	60	40	46	71	35	92	63	88	63	94	49	7
8	65	43	32	97	100	65	0	83	56	40	42	26
9	28	79	86	56	72	75	10	33	47	81	74	39
10	43	43	47	68	100	86	96	86	13	54	68	49
11	8	57	83	79	85	63	99	74	58	65	69	21
12	61	31	56	61	74	83	85	31	99	53	49	72
13	93	54	56	63	53	46	96	33	82	4	81	39
14	72	63	64	65	79	71	96	88	72	1	79	43
15	35	21	35	39	69	96	89	65	39	49	40	39
16	60	15	58	56	82	100	83	100	43	58	6	32
17	71	75	94	51	100	93	96	97	46	57	14	74
18	96	97	96	64	58	96	89	... etc.				

CHAPTER 7 MOVE MENU

The following processes move data by:

1 reading from the Input file set using the **FILE** menu,

2 reading the site, etc. set using the **DATA** menu,

3 possibly changing the values in some way,

4 then writing at the end to the **Destination File** set in the relevant process dialog.

TRANSFORM linearly transforms series and ratings, and interpolates a new time partition in series.

COMPRESS reduces the number of filed values.

SELECT selects values inside or outside a set range.

MAXIMA selects the maximum value in every 1/12th year.

MERGE 1 creates a new vector series and merges values into one item of it.

MERGE 2 merges values into an existing vector series.

MELEM merges values from two sites into the same item.

PSIM does calculations.

COPY copies series or ratings in a set time period.

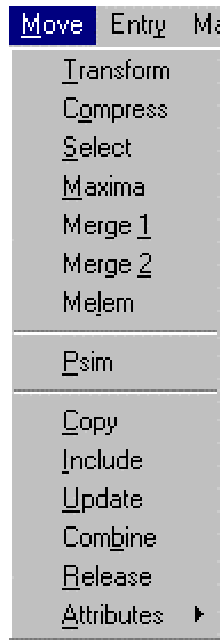
INCLUDE copies all components of a set site.

UPDATE copies newer data from the Input file to the destination file.

COMBINE copies the entire contents of the Input file.

RELEASE copies the entire contents of the Input file.

ATTRIBUTES copies the Attributes between Attribute files.



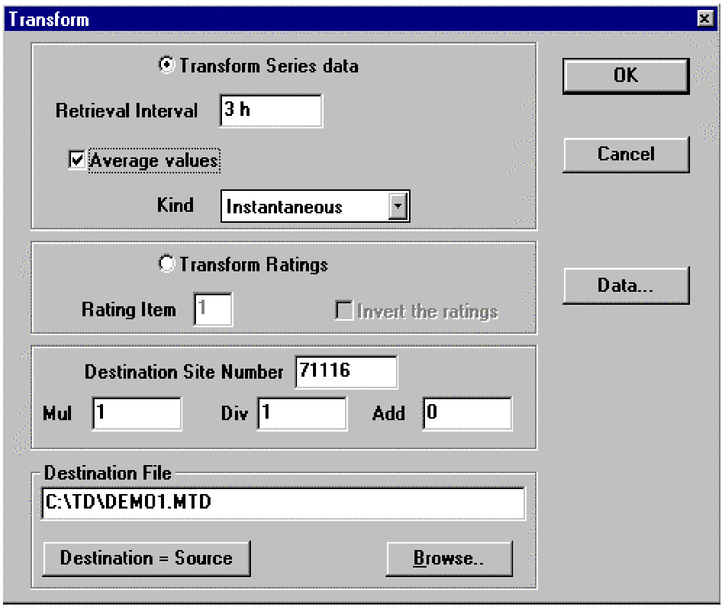
TRANSFORM

This process copies a time series, and simultaneously applies ratings and/or a linear transformation. If requested, it will also compute interpolated values on a specified time partition, instead of reproducing the times stored in the source file.

For multi-item data, only the specified item is transformed and the other items at each time are copied unchanged. However, if the process is interpolating, it interpolates for all items, not just the one being transformed. See page 13.3 for an illustration using this process.

Processes **MERGE 1** and **MERGE 2** described below will do the same with a series input but write only one item from a multi item source and into only one item of a multi-item destination.

Transform Series data options include:



application of a rating, which is set using the **DATA** menu;

Retrieval Interval set > 0 for interpolation, or set = 0 to get the filed data without interpolation;

Compute Average values can apply only when **Retrieval Interval** is set > 0;

Kind at the destination which can differ from the Kind of the source;

Destination Site Number which can differ from the site of the source;

Mul, Div and **Add** for the transformation:

$$Destination\ value = Source\ value \times Mul\ /\ Div\ + Add.$$

Transform Ratings options include:

Rating Item at the destination, which can differ from the source.

Invert the ratings exchanges the unrated and rated values, so that a subsequent application to a series of rated data will recover the original unrated series. This is useful when collating data archived in different places such as when preparing and checking long historical records.

Destination Site Number

Mul, Div and **Add** for the transformation:

$$Destination\ unrated\ value = Source\ unrated\ value \times Mul\ /\ Div\ + Add$$

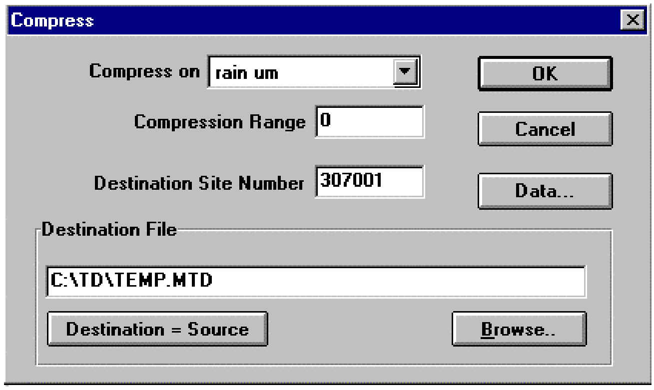
COMPRESS

This process copies a series, but only a reduced number of the discrete points, and these are the minimum required to define within a given tolerance the continuous line the series represents. See page 15.3 for an explanation of compression.

Compress on sets which item is tested for compression. This choice arises when compressing multi item series, and then when the value of this item meets the criterion for rejection, all items at that time are rejected.

Compression Range is a tolerance in file units. Values are rejected when the resulting graphed line lies within this tolerance of the original graphed line. With Histogram or Increment series, the values retained are adjusted to preserve the average and increment respectively.

When run with a compression range of zero on the Increment data series on the left, the result is the shorter series on the right.



Source: filed series

0	960115	11500
500	960115	13000
0	960115	54500
0	960115	60000
0	960115	61500
500	960115	63000
0	960116	54500
0	960116	60000
0	960117	53000
0	960117	54500
0	960118	53000
0	960118	54500
0	960119	53000
0	960120	191500
0	960120	193000
0	960121	181500
0	960121	183000
0	960122	24500
0	960122	30000
0	960122	144500
0	960122	150000
0	960122	181500
710	960122	183000
0	960122	193000

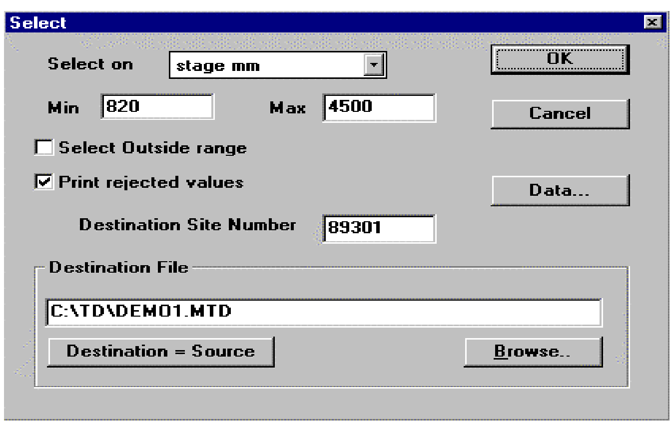
Destination: filed series

0	960115	11500
500	960115	13000
0	960115	61500
500	960115	63000
0	960122	181500
710	960122	183000
0	960122	193000

SELECT

This process copies the discrete points that are within, or outside, given limiting values. It operates on filed values without interpolation.

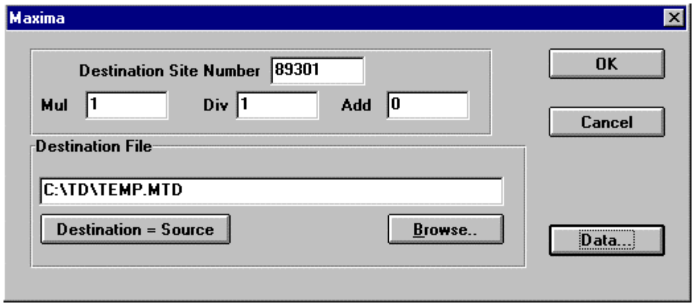
For example when set as shown here and with source data described in the report below, the report is:



```
Source is      C:\TD\DEMO.MTD  Site 89301 Whataroa R
Destination is C:\TD\DEMO1.MTD Site 89301 Whataroa R
From  961231 240000 to  970630 240000
REJECTED          4605 AT  970520 194500
REJECTED           819 AT  970630 210000
REJECTED           819 AT  970630 214500
```

The rejected values are omitted from the destination series without marking their times as Gaps.

MAXIMA

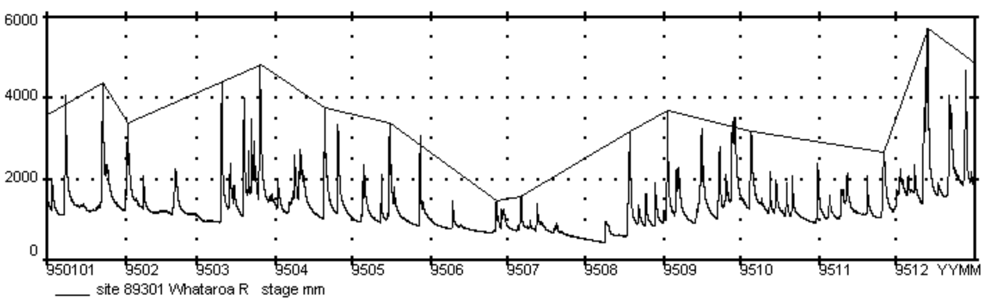


This process copies the maximum value in every 1/12th year. The time partition begins at the specified start, so to get calendar month maxima we must start at the beginning of a month.

It operates with the same limitation as process **QUICK EXTREMES** , see page 6.2; that is only on filed values without interpolation. The following illustrates use of this process, and also that because maxima should not be interpolated, they should be plotted as Gaugings. First we run **MAXIMA** on all the Site 89301 stage data in Demo.mtd with the dialog set as shown above.

```
Source is      C:\TD\DEMO.MTD Site 89301 Whataroa R
Destination is C:\TD\TEMP.MTD Site 89301 Whataroa R
From  931231 240000 to  971203  71500
1 month gap ending at  940802  13000
2 month gap ending at  941001 223000
1 month gap ending at  941201 193000
```

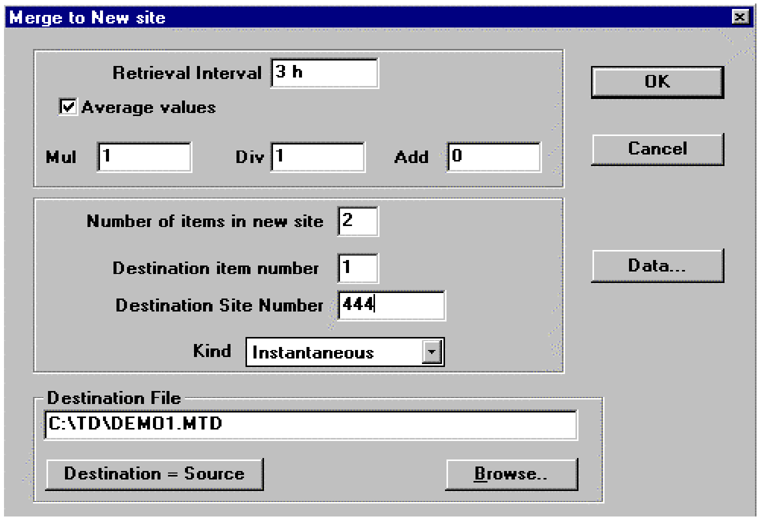
The following graph shows these monthly maxima from file Temp.mtd over-plotted on the



original series from Demo.mtd.

Process **MAXIMA** makes a good selection for frequency analysis. The next step is either to select the maxima from a longer time partition in which each interval is a fixed number of months, usually 12, or to select the peaks over a threshold. The data are relatively few and so much better than the original series for export to another program for this analysis.

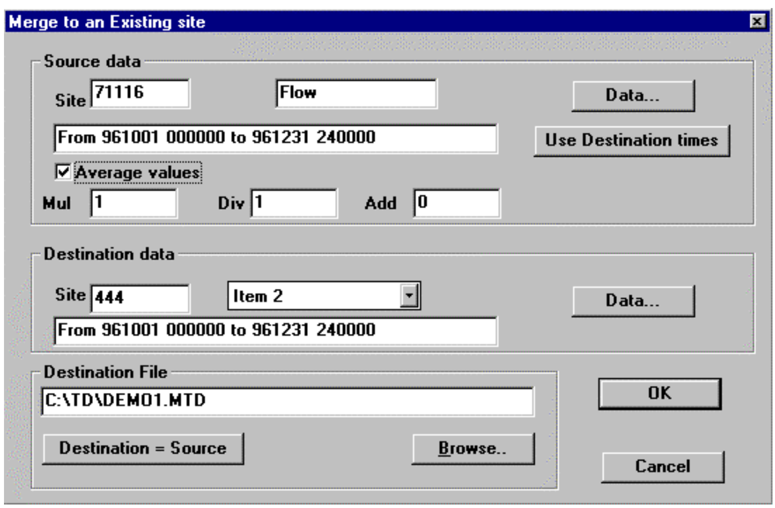
MERGE 1



This process is described in tutorial style on page 13.4. It calculates the same series as process **TRANSFORM** and writes it as one item in a multi-item series in which all the other items are set to zero.

It has only two options in addition to those offered by **TRANSFORM**: **Number of items in new site**, and **Destination item number**,

MERGE 2



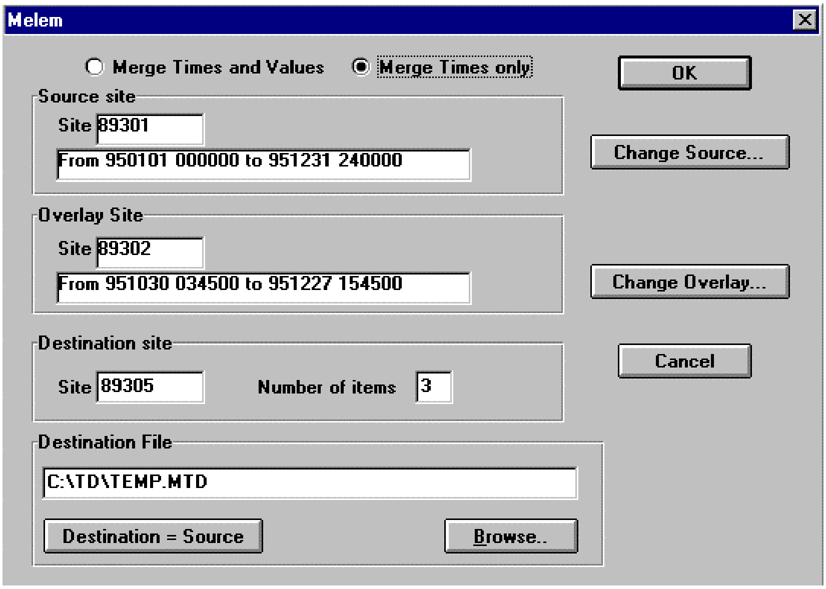
This process is described in tutorial style on page 13.6. It calculates values by interpolating the source series on the time partition of the destination series, which must already exist, and writes these values in the destination series.

The **Site**, **Item**, **From** and **To** of the source and destination are reported in the process dialog but are not set there. To set them click **DATA** in the relevant box on the **MERGE 2** dialog.

Time offsets can be introduced into a vector series by setting a different **From** and **To** times for the source and destination. This might be used to allow for the transit time of river flow; for example from a recorder to the location of other measurements.

MELEM

This process merges values at different times from two Sites, which must be of the same Kind, into a third site. It is used for merging Gaugings, and for constructing series with a particular time partition.



The following illustrates the contents of three files when processed using the dialog set as shown above. The destination series in this case contains the times but not the values from the two input series.

Source is C:\TD\TEST.MTD Site 89301

3540	950928	144500
2404	951031	34500
5718	951213	103000
4695	951228	154500

Overlay is C:\TD\TEST.MTD Site 89302

2404	951030	34500
5718	951212	103000
4695	951227	154500

Destination is C:\TD\TEST.MTD Site 89305

0	0	0	950928	144500
0	0	0	951030	34500
0	0	0	951031	34500
0	0	0	951212	103000
0	0	0	951213	103000
0	0	0	951227	154500
0	0	0	951228	154500

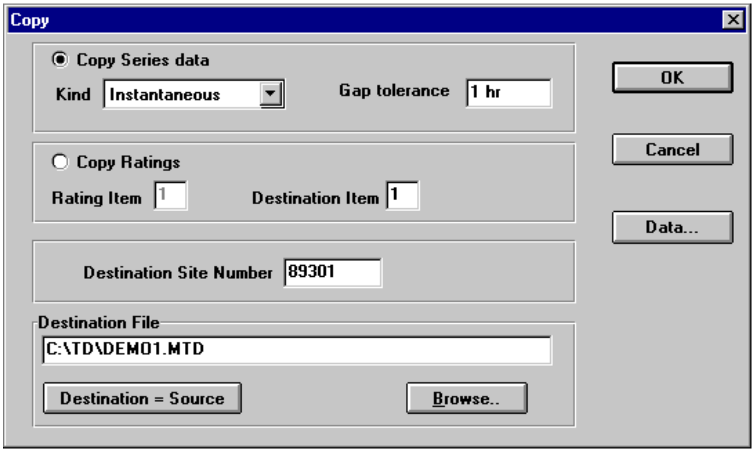
PSIM

This process does calculations and is described in Chapter 12 SIM LANGUAGE.

COPY, INCLUDE, UPDATE, COMBINE

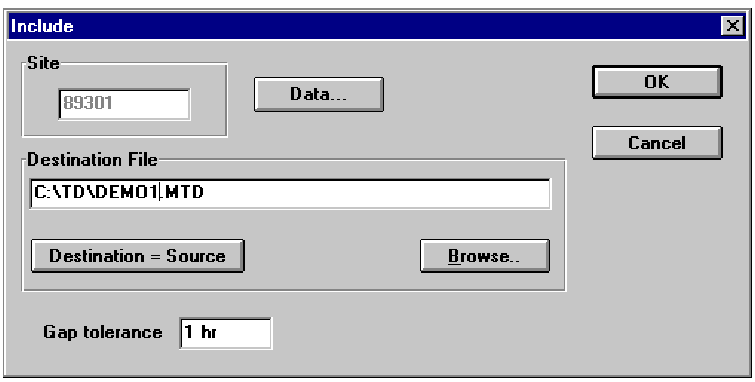
These four processes all copy data without changing values, and the latter three without changing properties.

COPY copies the source series or ratings in the time period set using the T**DATAT** dialog. It can change:
(a) the site and Kind of series and
(b) the site and Rating number of ratings.



The **Gap tolerance** specifies the maximum time duration between the last series value in the destination file and the first of the new values before a GAP mark will be written.

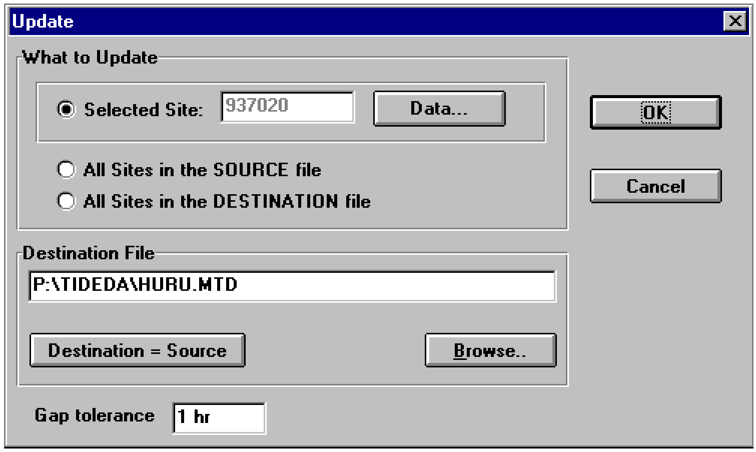
TINCLUDE copies the entire series, the ratings and the gaugings of the Site set using the **DATA** dialog.



UPDATE copies newer data from the input file to the destination file.

It can be used to update data in one of three ways:

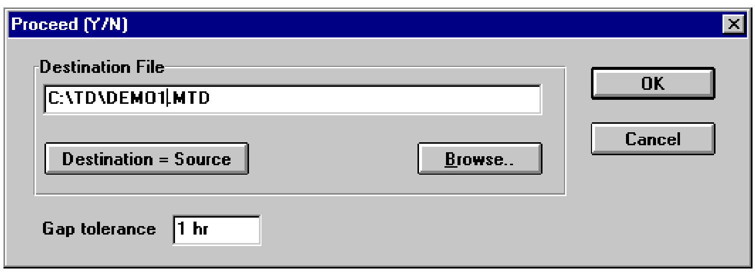
- for a single site;
- all sites in the input file;
- sites in the input file that already have some data in the destination file.



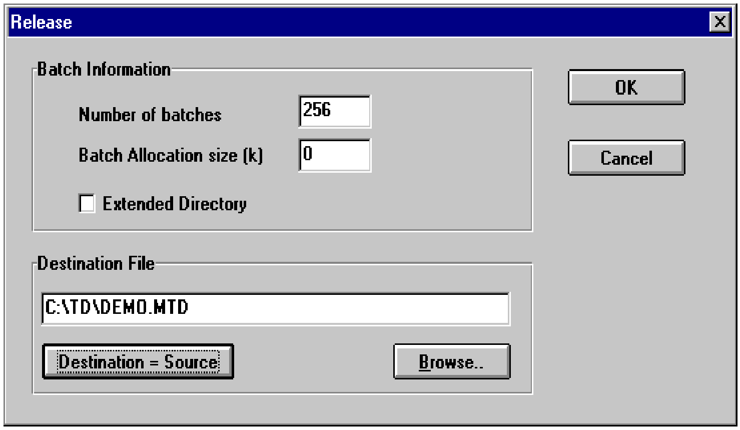
It is used to copy new data from a working file (such as a file of recently telemetered data) to another file with a single command and without specifying a time range.

It does NOT copy data earlier than the latest data already on the destination file.

COMBINE copies the entire contents of the Input file to the destination file.



RELEASE



(Also in the **MANAGE** menu)
This process releases the space occupied by non-current data on the disk. It copies all the data from the source file without changing the values, overwrites any existing contents in the destination file, and purges from the directory all references to data that are no longer current.

Before we **RELEASE** a file it is possible to recover any data on it but have been deleted or overwritten. Thus it is a good practice to **RELEASE** only when there is a good reason for doing so. An expert can help a beginner by diagnosing what been done to a file by examining its directory, provided that the file has not been **RELEASED**.

The **Number of batches** we specify will be rounded up to a multiple of 16 and cannot be larger than 32,000. We specify less than this maximum to save the storage space occupied by the directory at the start of the file. This space is 512 bytes for every 16 batches and is 1 Mb for 32,000 batches.

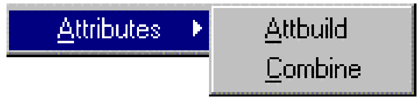
If we set the **Batch Allocation size** greater than zero, every time series batch will be allocated space that is an integer multiple of the specified size in kilobytes. Subsequent additions to that site will be written into any vacant space before a new batch is created so that the number of batches will be reduced. This feature is used for series that are updated with many small amounts of data. Many telemetry applications do this.

If we select the **Extended Directory** button the Destination file will have a directory into which we can copy directory information from other files without copying the contents of those files.

Click the **Destination = Source** button when this is the case, even when the name appears to be the same. This ensures that the process recognises the two names as the same and is not fooled by a minor difference such as a letter that is upper case in one version and lower case in the other.

ATTRIBUTES

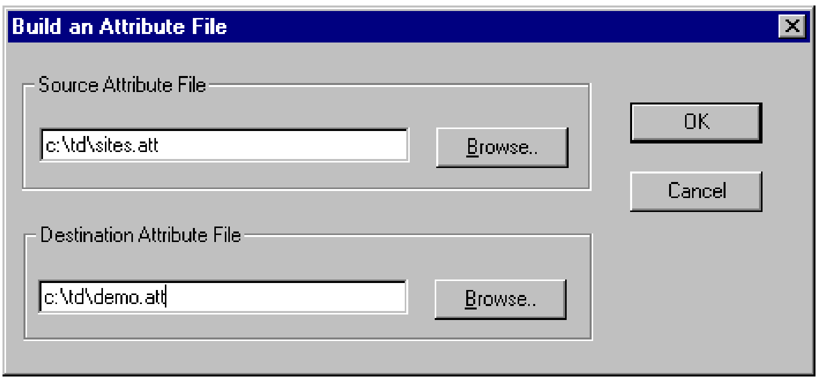
This is a submenu containing two processes >.



ATTBUILD

The **ATTBUILD** process copies the attribute records for source file sites from a master attribute file to the current Attribute file. It is used to build a small attribute file specific to the Input file by copying from a larger general purpose Attribute file that is usually called sites.att.

We can illustrate this process using the Demo.att supplied with Tideda to build a Temp.att file for the Temp.mtd file. First we use Windows Explorer, to make a copy of Demo.att and rename it sites.att, and then we can use the **ATTBUILD** process as follows.



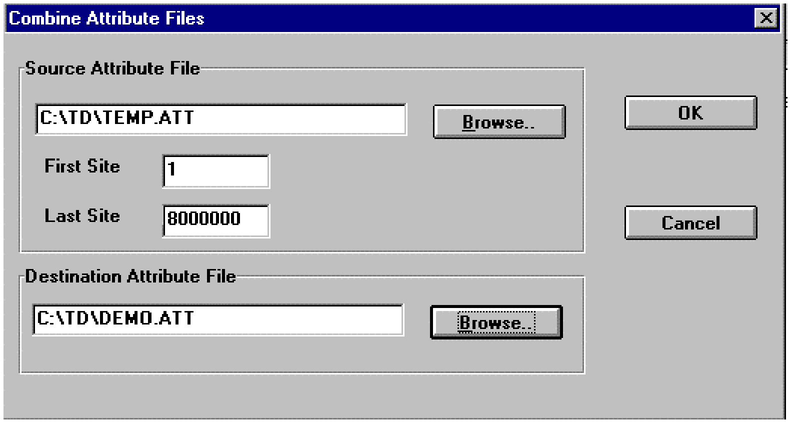

```
Input attribute file is C:\TD\SITES.ATT
Output attribute file is: C:\TD\TEMP.ATT
Site      296 HISTGRM lake gauge - smoothed
Site      295 HISTGRM lake vector
Site      297 INSTANT lake gauge
Site      698 INSTANT river gauge
```

If Sites.att cannot be found this message appears >, and the **Destination Attribute file** is unchanged, or if it did not exist it is created and empty.

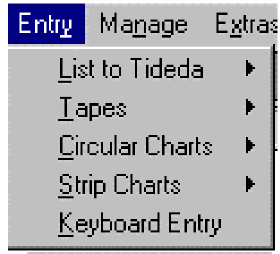
The other method to create a new attribute file is to set the attribute file name from the File menu then add new Attributes from the Edit menu (see chapter 4).

COMBINE

The **ATTRIBUTES COMBINE** process adds the attributes from one Attribute file to another attribute file.



CHAPTER 8 ENTRY MENU

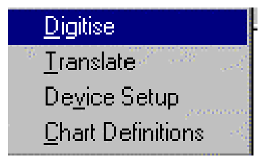


The **ENTRY** menu points to Tideda processes that read data that are not already in a Tideda file, and write them into a Tideda file.

There are four submenus and one process in the **ENTRY** Menu.

List To Tideda processes read text from the List file and write it to a Tideda file. The text may be **FULLY SPECIFIED** where the time of each value is specified (**TLIST**), or just **SERIES VALUES** at equal time steps (**TLSTAGE**), or **TIME STAMPS** at fixed increments of value (**TLRAIN**).

Tapes processes read binary numbers from a tape reader. **READ TAPE** copies the numbers to the Batch file. **TRANSLATE STAGE** and **TRANSLATE RAIN** read the Batch file and write it to a Tideda file. **DEVICE SETUP** identifies where the tape reader is connected. **LIST TO BATCH** enables the binary Batch file to be edited.



The **Circular Charts** processes read circular chart coordinates from a digitiser.

The **Strip Charts** processes read strip chart coordinates from a digitiser.

DIGITISE prompts us as we use the digitiser and writes the coordinates to the Batch file. **TRANSLATE** reads the Batch file and writes to a Tideda file. The **DEVICE SETUP** and **CHART DEFINITIONS** dialogs are used to set parameters.

KEYBOARD ENTRY prompts typing of text and lays it out so the process **FULLY SPECIFIED** can read it. This is the old **ADDATA** process.

Discussion of data entry

FROM PAPER

We capture numbers printed on paper into a computer readable file using a keyboard or optical character reader. For example the data might be a manual daily rain record or a table in an old yearbook. The file is then processed as a text file; see below.

FROM CHART DIGITISERS AND BINARY RECORDERS

The earliest machines that (since the 1920s) have recorded time series unattended use a pen to draw a line on a paper chart. The best way to capture this data into a computer is usually to use a digitiser, which sends the coordinates of points on the line on the chart to Tideda. Subsequently more precision and more convenient processing became possible with recording machines that store numbers on punched paper tape (since the 1960s), magnetic tape (since the 1970s) and various kinds of data pack containing magnetic memory chips. A two-stage procedure is necessary to enter this data into Tideda.

1. First we use a process that reads the output of the digitiser or recording device into the Tideda **Batch file**, which is called Batch.dat. We use the word Batch to refer to a set of data entered at one time. The **Batch file** contains the raw data, but no identification such as a Site number or times in calendar-clock notation. For more information about how we use the word Batch, see page 15.7.

Then we use a second process to translate the **Batch file** contents into a Tideda file with any time or value transformations applied.

FROM TEXT FILES

Since 1990 we have had electronic field recorders with memory chips from which data are extracted as a text file using proprietary software on a personal computer. Such recorders, including a sensor, a cable for connection to a personal computer and associated software can cost less than $NZ200. There are many different layouts of the proprietary text files produced by these recorders.

The standalone TLOGGER program, for translating particular recorder formats into a Tideda file, can be provided by NIWA (see chapter 9). Alternatively we have the following two-stage procedure:

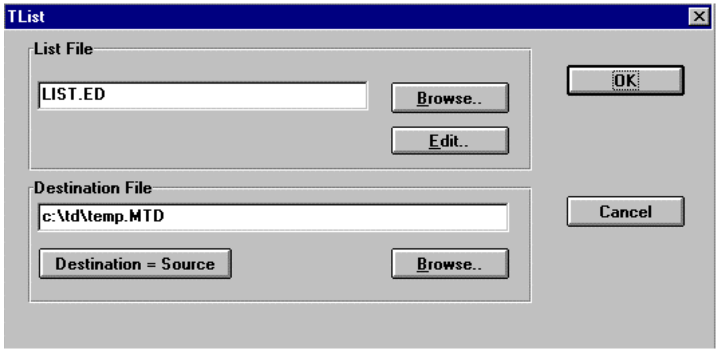
1. First we translate our proprietary text file into one of the **List To Tideda** layouts, and write it into a **List file** as text. This file can have any name. Chapter 16, ENTRY VIA EXCEL, illustrates several ways to do this.
2. Then we use one of the processes in the submenu **List To Tideda** to translate the **List file** contents into a Tideda file.

FROM TELEMETRY AND OTHER TIDEDA FILES

ENTRY menu processes are not required when using telemetry systems that file data directly into Tideda files, or when getting data from Tideda files at other locations. The following tools do this.

1. The FLOSYS program, available from NIWA, operates radio telemetry equipment and writes the data it gets into a Tideda file.
2. The Tideda Internet Server (TDSERVER/TDCLIENT), described in Chapter 9, gets data from other Tideda files on computers connected to the internet.
3. Processes in Tideda's **MOVE** menu get data from other Tideda files on mapped disk drives.

List to Tideda - FULLY SPECIFIED



This process translates data laid out in the same way as data written by the **TABLE** menu's **LIST** process. Numerical parameters are not required for translation so that its dialog requires only the names of a **List file** and a Tideda **Destination file**.

It can be used to edit the contents of a Tideda file. We start with process **LIST** to write the data to a List file, then run process **FULLY SPECIFIED** and click the **EDIT** button. This will display the data in the window of the text editor specified in our **FILE** menu's **PREFERENCES** dialog. We can now edit the data, while retaining the obvious essential features of the layout. When we close the editor window, click **Destination = Source** and click **OK**, the revised data replaces the original data in the **Source file**.

The layout of series:

- can have unequal time steps
- can have several Items at each time
- may include extra blank spaces or blank lines

The Site is the first number on the site header line (lines which have non-numeric characters), followed by the number of Items then the data Kind, which must be either **Instant, Histogram, Increment or Gaugings** (or sufficient letters from the beginning of one of these labels to be unambiguous).

For example a **continuous series with four Items and Kind = Histogram** > The number of items follows the Site number.

```
8704      4 HISTOGRAM
204718  69914 62636      0 960121 240000
204716  84129 62618 214501 960122 120000
211776 125151 62367 204221 960122 240000
```

A 2nd example is a **continuous series with one Item and Kind = Instant** > In this case we also specify a **Gap** from 941108 71500 to 941110 121500.

```
89301      1 INSTANT
 2618   941108    70000
 2613   941108    71500
 89301      1 INSTANT
 1985   941110   121500
 2088   941110   124500
 2134   941110   130000
```

Gaps are inserted in new otherwise continuous series by entering another header line.

Another Gap will be retained in the destination file if the newly entered Continuous series data either begins in a Gap that is already on that file, or is after the last data.

A 3rd example is a continuous **series with Kind=Increment** >
Note that the initial data value of 1000 cannot be plotted as part of a series unless this batch is inserted into a series which includes a previous time. This is because to plot this series as a continuous line versus time requires both an increment and a time interval to define a rate of increase.

```
307001      1 INCREMENT
1000   960202    43000
500   960202    44500
 0   960202    64500
500   960202    70000
1500   960202    71500
1500   960202    80000
2000   960202    81500
```

A 4th example is a **Gauging series** >
These are calendar monthly maxima from the continuous series at this site, and were actually calculated made using the **MOVE** menu's process **MAXIMA**; see page 7.4.

```
89301      1 GAUGING
3182   951005    3000
2656   951126    34500
5718   951213   103000
```

Our last example is a **Rating** >
In this case the text label is RATING which follows the Site number and is followed by the number which identifies which Rating series. Then follows parameters defined on page 15-5: start_date start_time effective_date effective_time

```
89301 RATING 1 951213 80000 951213 140000
 920    16000
1020    21600
1120    27900
1370    53500
```

Any combination of fully specified series and ratings, in any order, can be in the same List file. However the sequence of times of the data between any two header lines must increase.

It is also possible to enter Tideda **Attributes** from a text file in this way.

The format of an attribute in the list file is:

```
<site> <kind> ATTRIBUTE
TITLE=<title>
ITEM=<item>,<meas>,<file units>,<pres units>,<format>,<div>
RITEM=<item>,<meas>,<file units>,<pres units>,<format>,<div>
END
```

For example:

```
12345  INSTANT    ATTRIBUTE
TITLE=Limpopo AT Timbuktu
ITEM=1,Stage,mm,mm,######,1
ITEM=2,Velocity,mm/s,m/s,#.###,1000
RITEM=1,Flow,l/s,m3/s,####.##,1000
END
```

There can be up to 15 ITEM lines and 4 RITEM lines for each attribute. If the TITLE line contains "AT" then the title will be split into the two title fields.

Data for the Attribute file and the Tideda data file can be used in the same List file, but they should preferably not be mixed together. This is because the attribute file must be closed for writing before the data file can be opened for write access, and vice versa. Although it is possible to mix the types, it is better to do all of one before adding the other.

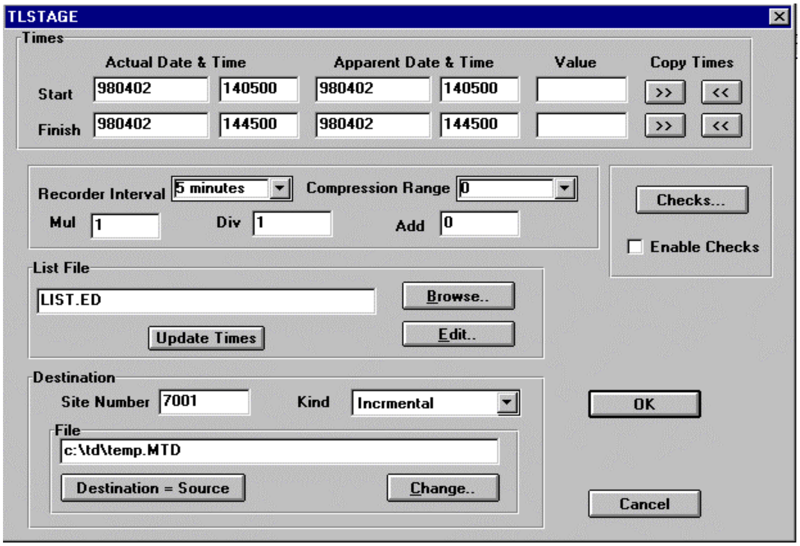
List to Tideda - SERIES VALUES

This process is for translating data at equal time steps called the **Recorder Interval**. It is similar to the **TRANSLATE STAGE** process in the **TAPES** menu, but it reads from the List file rather than the Batch file.

The List file must contain a series of integer numbers separated by blanks or line feeds, e.g. >.

```
   0   6656    10640 4867   0
2358   8095    7817   3330
```

Parameters for translating this data are set in this dialog >.



When the **Actual** time in the dialog differs from the Apparent time, which represents the recorder's clock and may have been checked and found to be different, a ramped correction is applied so that the times written to the Tideda file match the Actual times.

When the Value in the dialog is not blank and differs from the Value in the List file, a ramped correction is applied so that the Value written to the Tideda file matches the Value in the dialog.

Data in the List file may be preceded by one or two header lines beginning with the label TIMES and followed by 4 integers or INTERVAL followed by the recorder interval in seconds.

TIMES start_date start_time finish_date finish_time

INTERVAL interval

e.g., `TIMES    980402     140500     980402     144500`

When there is a header like this, the dates and times are automatically transferred to the **Apparent** dates and time boxes on the dialog. The **Copy Times** buttons can then be used to transfer them to the **Actual** dates and times if that is appropriate.

For example with the List file contents and dialog illustrated above, click the **OK** button to write the data to the **Destination file** and get this in the text window.

```
START-DATE   TIME  APPARENT-DATE TIME    STAGE
      980402 140500       980402 140500
FINISH-DATE   TIME  APPARENT-DATE TIME    STAGE
      980402 144500       980402 144500
NO MID-TAPE CORRECTIONS
OPTIONS:
LABEL:-
MUL     =      1  DIV    =    1  ADD    =     0
INTERVAL=       300
List File is   LIST.ED
c:\td\temp.MTD   SITE    7001
COMPRESS=         0
No. of values is     9
MINIMUM FOUND WAS         0 AT   980402  140500
MAXIMUM FOUND WAS     10640 AT   980402  141500
```

If the **Recorder Interval** is set to 6 minutes then nothing is written to the Destination file and the text ends with this error message >.

```
INTERVAL=       360
COMPRESS=         0
No. of values is     9
***** 30 -       2 VALUES EXTRA
```

If we do not know the precise **Apparent** time span covered by our data, then it is necessary to guess times and use this error message to calculate the precise times which we must be able to enter to successfully translate the data. This iterative, interactive procedure has been found to be a reliable way to capture this kind of data.

In this case, when we use process LIST on the resulting Tideda file we get >.

```
Source is c:\td\temp.MTD Site 7001 < Untitled >
 1 Item   INCREMENTAL From  980402 140500 to  980402 144500
     Item 1        Date        Time
*** GAP ***
          0      980402      140500
       6656      980402      141000
      10640      980402      141500
       4867      980402      142000
          0      980402      142500
       2358      980402      143000
       8095      980402      143500
       7817      980402      144000
       3330      980402      144500
```

List to Tideda - TIME STAMPS

This process is for translating series represented by the set of times for equal Increments of a recorded quantity. Typically the time of each tip of tipping bucket rain gauges. It is similar to the **TRANSLATE RAIN** process in the **TAPES** menu, but it reads from the List file rather than the Batch file.

The times are represented by a count of **Recorder Interval**s, e.g.

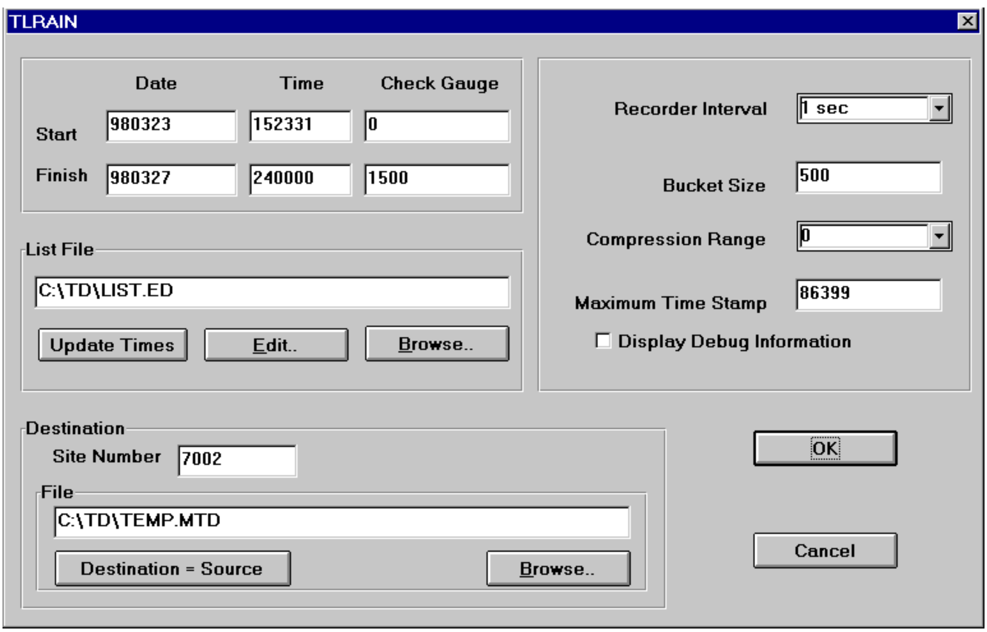
```
55411  0      0      83603  85135  0      306    0      0       1
```

Each Time_Stamp is the remainder when

(Time / Recorder_Interval) is divided by *(Maximum_Time_Stamp + 1)*.

Thus the series of time stamps increases to the maximum, which in this example is 86399, then goes to zero and increases again. The zeroes must always appear in the record even though there is no tip of the bucket. In fact if the bucket tips in the time interval of the zero, its time must be arbitrarily delayed and recorded as 1. More than one tip can be recorded with the same time stamp. The series must end with a non-zero time stamp, and this last time is assigned a zero increment.

Parameters for translating this data are set in this dialog:



Recorder Interval is the passage of time required for the Time Stamp to increase by one.

Maximum Time Stamp is the maximum possible value, typically 9999 or 86399.

Bucket Size when the Check Gauge = 0 is the Increment of value per Time Stamp.

When **Check Gauge** > 0, Bucket Size = Check Gauge / number of tips.

Data in the List file may be preceded by a header line beginning with the label TIMES and followed by 4 integers:

TIMES start_date start_time finish_date finish_time

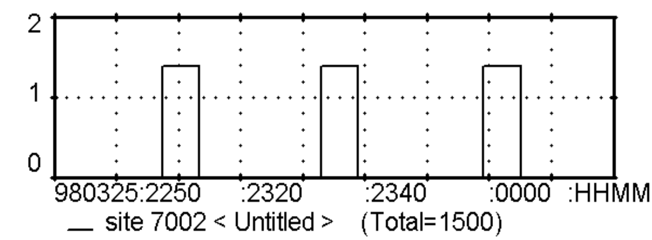
e.g., `TIMES 19980323 152331 19980328 0`

When there is a header like this, the dates and times are automatically transferred to the dialog.

For example with the List file contents and dialog illustrated above, click the **OK** button to write the data to the **Destination file** and get the following in the text window:

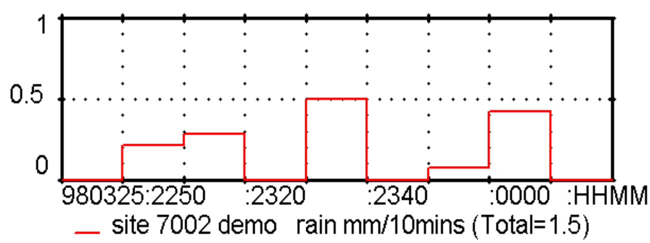
```
List File is   C:\TD\LIST.ED

C:\TD\TEMP.MTD   SITE    7002
      START-DATE     TIME     CHECK       STAMP
                              GAUGE
           980323   152331        0       55411
     FINISH-DATE     TIME     CHECK       STAMP
                              GAUGE
           980327   240000     1500           1

OPTIONS:

MAXIMUM TIME STAMP =      86399
INTERVAL=          1
SIZE     =       500
COMPRESS=          0
Final Check:
         Recorder Total before Check Gauge correction =         1500
         Recorder Total after  Check Gauge correction =         1500
         The recorder clock appears to be correct
```

A plot of this data in **File Units** with **Retrieval Interval** = 1 sec. looks like this >



The 500 µm rain that caused each tip of the bucket is filed as if it had taken 6 minutes to accumulate at a rate of 500/360 = 1.389 µm/s.

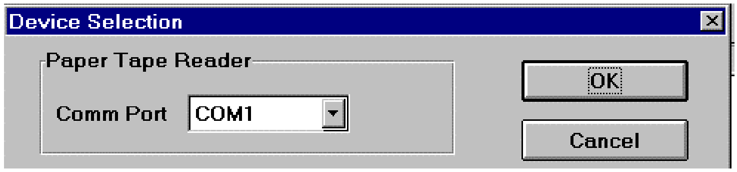
Another plot, this time in **Presentation Units** with **Retrieval Interval** = 10 min. looks like this >



The graphs 10 minute time partition has split some of the 0.5 mm amounts between two periods.

Tapes – DEVICE SETUP

In Tideda jargon a “tape” is any media on which data can be stored, and from which it can be read by the computer as a sequence of binary integers. Punched paper tape, and binary magnetic tapes and magnetic memories are in this category. A device, normally supplied with the recording equipment, which can be plugged into a computer communication port, reads the tape.



Use this dialog to set the computer port read by process **READ TAPE**. The communication parameters are hard coded as 9600 baud, No pariry, 8 bits, 1 stop bit.

Tapes – READ TAPE

This process reads data from the tape and writes it to the Batch file (see page 15.7). There are no options, and so no dialog. When we click on the process a message will appear in the text window like this >

```
Paper-Tape is COM1.DAT
Batch file is BATCH.DAT
End of process
```

If no suitable device is connected, the “End of process” message will not appear and the Batch file will be locked. To continue processing simply click on another process. However to unlock the Batch file we must exit the Tideda program.

The device that is read need not be a paper tape, and can be any device that outputs binary integers in the range –2147000000 to 2147000000. For example a magnetic tape or a memory chip.

To see the Batch file contents: click the **TABLE** menu
click process **LIST**
select **List Batch file to List file**
click the **EDIT** menu
click process **LIST FILE**
and the editor window will look like this >

```
1206 1206 1206 1207 1206 1206 1206 1206 1206 1206
1206 1207 1206 1206 1206 1205 1206 1206 1207 1207
1207 1207 1208 1209 1212 1216 1219 1222 1228 1234
1236 1237 1243 1248 1258 1267 1279 1288 1291 1292
1294 1298 1303 1304 1307 1307 1311 1315 1318 1334
1336 1332 1332 1336 1335 1335 1331 1332 1335 1334
1334 1335 1330 1331 1327 1325 1321 1321 1319 1319
1316 1314 1313 1312 1309 1308 1305 1302 1301 1299
1296 1291 1290 1291 1288 1286 1283 1281 1278 1279
1276 1274 1273 1271 1270 1269 1267 1265 1264 1264
1264 1262 1260 1259 1352 1782 1787 1786 1330 1259
1257 1251 1250 1247 1248 1247 1245 1246 1245 1243
1243 1242 1243 1241 1242 1241 1241 1239 1239 1237
1237 1237 1237 1238 1236 1237 1235 1234 1235 1236
1235 1235 1236 1234 1234 1234 1233 1233 1233 1233
1233 1232 1232 1237 1234 1234 1235 1234 1233 1233
1233 1233 1233 1232 1232 1232 1232 1231 1231 1231
1231 1231 1231 1230 1230 1230 1230 1230 1230 1229
1230 1229 1229 1229 1229 1229 1228 1228 1229 1228
1228 1228 1229 1228 1228 1228 1227 1227 1227 1226
1227 1227 1227 1227 1227 1226 1226 1227 1226 1226
1226 1226 1226 1226 1226 1225 1225 1225 1225 1225
1225 1225 1225 1225 1225 1225 1225 1225 1224 1225
1224 1225 1224 1225 1224 1224 1224 1224 1224 1225
1225 1224 1224 1224 1224 1224 1225 1225 1225 1223
1222 1220 1220 1220 1220 1220 1220 1220 1220 1220
1220 1220 1220 1220 1220 1220 1220 1220 1349 1392
1819 1824 1817 1821 1819 1822 1822 1819 1819 1825
1821 1813 1819 1818 1819 1812 1822 1815 1813 1816
1816 1816 1817 1821 1814 1813 1815 1815 1812 1809
1801 1813 1816 1813 1815 1814 1814 1815 1815 1816
1811 1812 1812 1815 1811 1809 1812 1811 1808 1812
1812 1812 1812 1811 1814 1808 1810 1808 1801 1808
1811 1818 1810 1814 1824 1813 1809 1810 1808 1817
1817 1812 1809 1815 1812 1808 1808 1814 1812 1811
1439 1222 1217 1212 1209 1209 1209 1208 1208 1208
1208 1207 1207 1207 1207 1207 1207 1207 1207 1207
1207 1207 1207 1207 1207 1207 1207 1207 1207 1207
1207 1207 1207 1207 1206 1206 1206 1204 1203 1202
1202 1202 1201
```

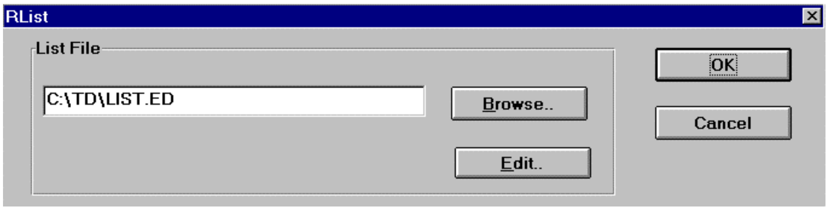
The Batch file may be too large for the text editor specified in our **FILE** menu's **PREFERENCES** dialog, and we must upgrade to a more powerful editor. In this situation the operating system will automatically upgrade the Notepad editor to the Wordpad editor.

Editing the Batch file

To edit the binary contents of the Batch file, we list it as described immediately above, then edit it, and copy it back to the Batch file as described immediately below.

Tapes – LIST TO BATCH

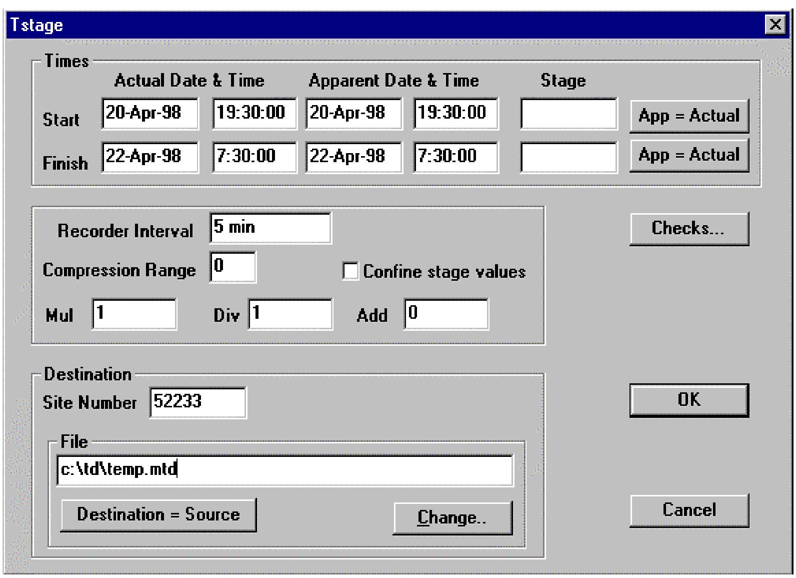
This process enables us to write the List file contents to the Batch file.



Tapes – TRANSLATE STAGE

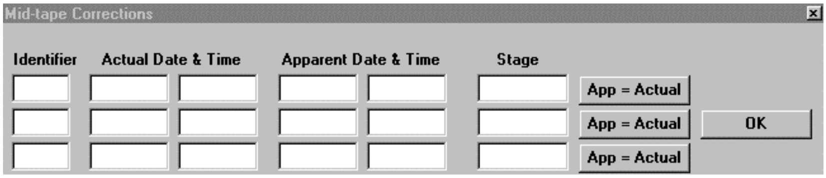
This process reads the Batch file and writes the contents to a Tideda file. Note that the word "stage" is used here synonymously with the phrase "Instant value" used elsewhere in this manual. The Batch file must contain a series of values equally spaced in time.

Apparent Date & Time represent the recorder's clock, which may be different from the actual time. When it is a ramped correction is applied so that the times written to the Tideda file match the Actual times.



When the **Stage** in the dialog is not blank and differs from the Stage in the Batch file, a ramped correction is applied so that the Stage written to the Tideda file matches the Stage in the dialog.

When the recorder has been checked at other times than when the tape was changed, and errors have been noted in the time and/or stage, then click **CHECKS** to get this dialog > where we enter the mid tape corrections.



Recorder Interval is the nominal time step used with the **Apparent** times to calculate how many data values there should be. If the batch file does not contain the calculated number of values, no data are filed and messages explain how much the **Apparent** time needs to be changed. Thus an iterative process can be used to determine these times. As recorder clocks have improved the need for this adjustment has decreased.

Confine stage values confines the values to the 10 000 range defined by the stage at the start time. For example, consider two consecutive tape readings 9500 and 0300 and an initial stage in the range 20 000 to 29 999. If confined, the data values will be 29 500 and 20 300 (i.e., no adjustment for "going around the clock"). If not confined, the values will be 29 500 and 30 300. Specify **Confine** only to stop large stage jumps from being interpreted wrongly.

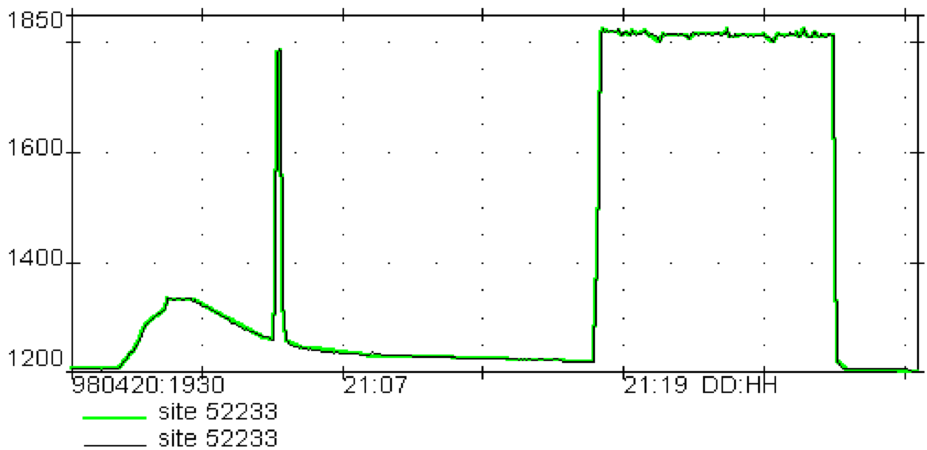
The Compression Range defaults to zero.

After entering **OK** the following summary appears:

```
C:\TD\TEMP.MTD  SITE  52233
START-DATE TIME APPARENT-DATE  TIME  STAGE
980420   193000    980420  193000
FINISH-DATE TIME APPARENT-DATE  TIME  STAGE
980422   73000    980422  73000
NO MID-TAPE CORRECTIONS
OPTIONS:
MUL   =   1 DIV  =   1 ADD  =   0
INTERVAL=    300
COMPRESS=     5
Batch file is BATCH.DAT
No. of values is  433
Batch file is BATCH.DAT
No. of values reduced from  433 TO  56
MINIMUM FOUND WAS    1201 AT  980422  73000
MAXIMUM FOUND WAS    1825 AT  980421 184500
```

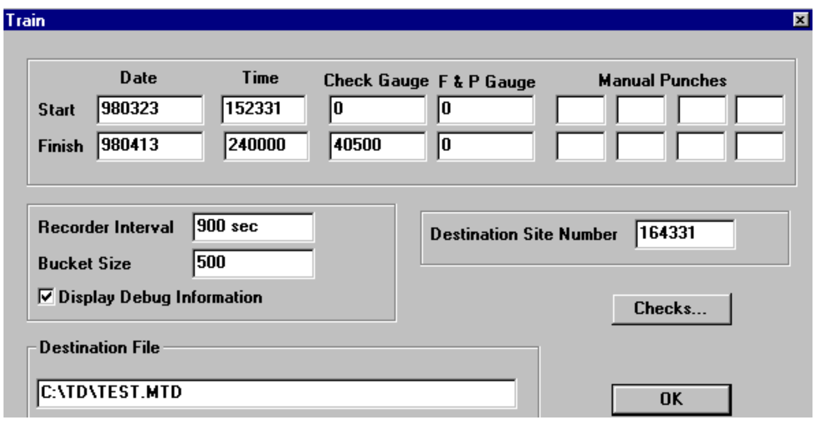
Note the 6-fold reduction in that this data which occupied 2598 (433 x 6) bytes as a text file now occupies 448 (56 x 8) bytes in a Tideda file, while retaining 5-minute time resolution and 5-mm stage resolution.

The following graph compares the original data with the filed data. The series is from a stage recorder downstream of a power station. There are stage rises of around 600 mm due to discharges from the station superimposed on stage rise due to a fresh from the natural catchment. Good resolution is necessary to get accurate flow volumes in this kind of situation.



Tapes – TRANSLATE RAIN

This process reads the Batch file and writes the contents to a Tideda file. Note that the word "rain" is used here synonymously with the phrase "Increment value" used elsewhere in this manual. The Batch file must contain a series of Time Stamps, which are a count in Recorder Intervals of the times when the recorded quantity has incremented by a constant amount.



Recorder Interval is the unit of the Time Stamps.

Check Gauge is the accumulated amount, usually the rain caught in a nearby storage gauge measured in micrometres.

Bucket Size is the Increment per Time Stamp when the Check Gauge difference = 0.

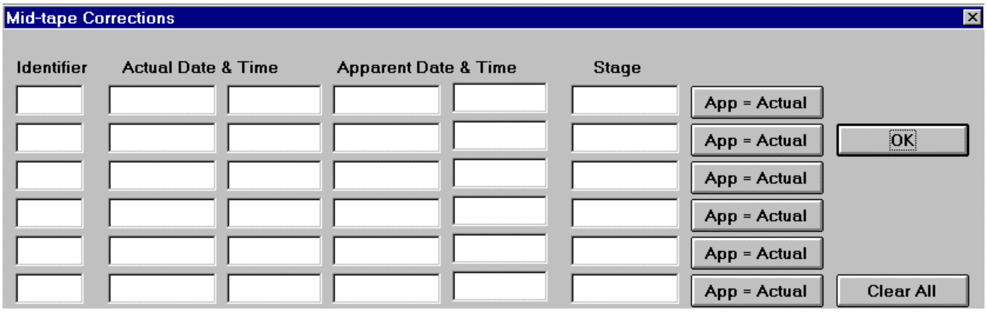
If the **Check Gauge** difference > 0 then:

Increment per Time Stamp = Check Gauge difference / number of tips.

F & P gauge is the accumulated amount measured in micrometres, collected under the tipping bucket mechanism. It provides a check as to the overall accuracy of the recorder, but is not used to apply any corrections to the data.

Manual Punches are spurious time stamps caused by the operator test the mechanism during visits to change the tape. Enter the values here and they will be excluded from the fileed record.

Click **CHECKS** to get this dialog >



Mid tape corrections are made at times when the gauge is visited but the tape is not changed, to test the mechanism. Up to four spurious Time Stamps may be punched in the tape, and the punched numbers and the time they were stamped are written in the gauge's log book. Subsequently, when translating that data, the manual Time Stamps and their times are taken from the log book and entered in this dialog, and excluded from the filed record. The presence of manual punches also provides a check on the accuracy of the recorder. If a recorder breaks down only data since the last check are lost. We should not change a tape, or introduce any manual punches while it is raining, because the process will be unable to distinguish our manual punches from the legitimate ones. If this unfortunate situation does occur, we must edit the batch file and change the value of the manual punches to be before or after the rainfall, as appropriate.

Error messages include:

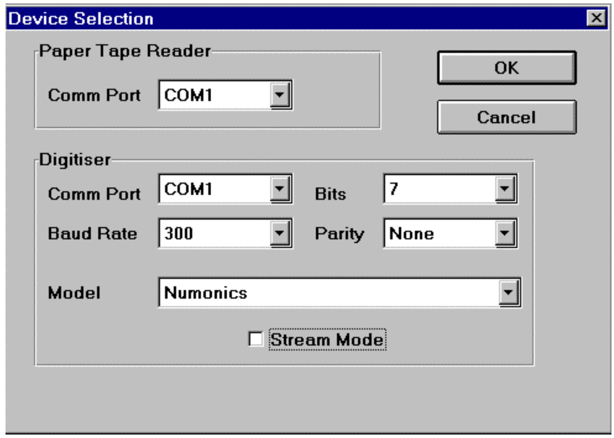
(a) "Questionable punch" where a negative value is found in the Batch file, possibly generated during the **READ TAPE** process as a tape-reader signal of an error;

(b) "Apparent drought" when there is a large time period between consecutive Time Stamps, usually caused by mis-punches.

Following an error message, it is likely that the data in the Batch file can be successfully edited as explained on page 8.9.

Charts – DEVICE SETUP

Use the digitiser part of this dialog > to identify the port on our computer to which we have connected a digitiser.



Model refers to one of the following digitisers supported by Tideda:

```
Numonics
Sumagraphics (original)
Sketchpad (Summagraphics)
Microgrid
Graphtec
Gtco Digi-Pad
HPA1 Plotter (in dig. Mode)
Kurta IS/3 (Mode 8)
Kurta IS/3 (Mode 5)
Logitech K-510
```

If our digitiser is not included in this list, contact a Tideda agent for advice. One of these protocols may be emulated by our digitiser, or it may be necessary to add to the list.

How to set up a digitiser

Setting up a digitiser differs for each model. We might be able to engage an expert to do this, and it is only done once. The following notes show us, or our expert, the scope of what is required. They are illustrated by the requirements of a particular model. Our Tideda agent may be able to supply equivalent notes detailing the requirements of our particular model.

DIP SWITCHES

Set dip switches on the digitiser according to the manufacturer's manual and the requirements of the relevant device driver that is in the Tideda program. The kind of detail that is required is illustrated with details about a GTCO - Digipad model 3648L.

The GTCO - Digi-pad has three arrays of eight dip switches mounted in the edge of the left side of the table. The arrays are labeled from the left S3, S2, S1. They are set as follows where 0 = off = down, 1 = on = up, and x = setting not used:

```
S3                S2                S3
x 0 0 0 1 1 0 0   1 1 1 1 1 x x x   0 0 1 1 0 x 0 1
```

After setting the switches, press the button to the left of the dip switches to cause them to take effect, and the digitiser will beep twice.

CURSOR PARAMETERS

The strip chart and circular chart **DIGITISE** processes detect two cursor buttons called the primary button (or button 1) and secondary button (or button 4). Digitising a point within 20mm of the lower left corner of the digitiser tablet's active area acts as the secondary button for use if our digitiser cursor only has a single button. Otherwise the secondary button is the button that sends a button code of "4" when pressed – it might not actually be labelled "4" on the digitiser cursor. The primary button is any other button on the cursor. You may need to connect to the digitiser with the Windows "Hyperterminal" program to see which button sends the correct button code.

Two of the dip switch settings given above and for the Gtco Digi-pad relate to the cursor:

```
(S3: 2=off) this & the next switch specify 12 coordinates per second when in line mode
(S3: 3=off)
(S3: 4=off) specifies point/line mode and cursor button "F" toggles these modes
(S3: 5=on) specifies a 16 button cursor
```

DATA FORMAT

The digitiser sends a sequence of coordinate values to the computer and may provide options for varying the format.

For example the Gtco Digi-pad output is specified by the following dip switch settings which define 14 bytes per coordinate pair:

```
(S3: 6=on) specifies metric units
(S3: 7=off) specifies ASCII text
(S2: 1=on) specifies there to be the button number before each x,y pair
(S2: 2=on) specifies a space between the x and the y numbers
(S2: 3=on) specifies a carriage return after each x,y pair
(S2: 4=on) specifies a line feed after the carriage return
(S2: 5=on) specifies 5 digits per coordinate number, and units of 0.1 mm
```

CABLE

Connect the digitiser to a serial port (e.g., com2) on the computer with a cable.

CONFIGURE SERIAL PORT

Use the Charts **DEVICE SETUP** dialog to configure the computer serial port to match the digitiser's dip switch settings.

Enter the five parameters consistent with the digitiser settings, for example:

GTCO digitiser dip switch settings	Computer port settings
(S1: 1=0, 2=0, 3=1, 4=1)	baud rate 9600
(S1: 5=0)	parity off
(S1: 7=0)	1 stop bit
(S1: 8=0)	8 data bits

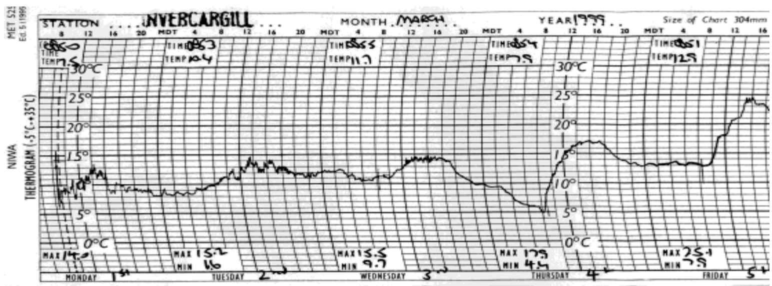
Using A Digitiser

A digitiser is a tablet attached to a computer on which we attach a chart. A cursor with cross hairs can be clicked over locations on the chart and this causes the tablet coordinates of each location to be sent to the computer. Only two cursor buttons, called the primary button and secondary button respectively, are recognised by the **ENTRY** menu's process **DIGITISE.** When unspecified, it is the primary button that is used. If our digitiser cursor only has a single button clicking this at the lower left corner of the digitiser tablet has the same effect as the secondary button. A RESET button is generally provided somewhere on the digitiser tablet.

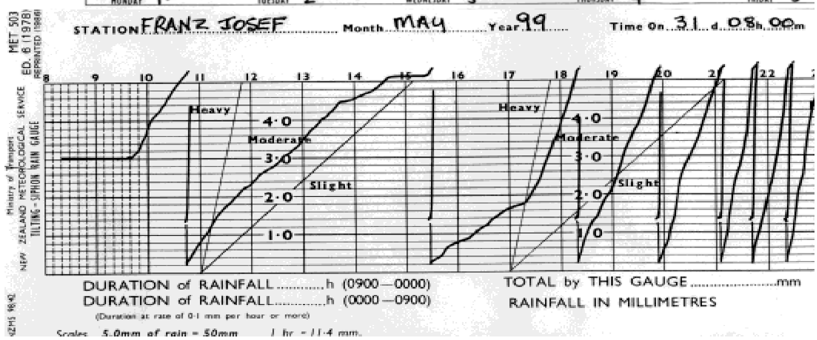
Firmly attach the paper chart to the digitiser tablet with tape. A strip chart need not be exactly parallel to the bottom of the tablet, and if it is longer than the tablet is wide simply attach the first part. Ensure that the information on the chart is over the active part of the tablet, not over the inactive margins.

Strip Charts – CHART DEFINITIONS

Part of a strip chart with a curved "barometric" y-axis. >

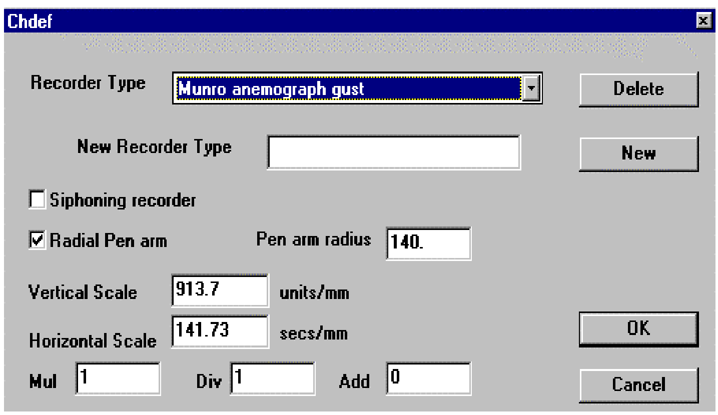


Part of a strip chart from a siphoning rain recorder. >



The digitising process relies on the start and finish times and values we specify and is not affected when the trace does not coincide with the printed grid, as in this last example.

Click **CHART DEFINITIONS** to get this dialog > where we enter the characteristics of a particular strip chart recorder to be kept in the Charts.def file



This is done only once for each kind of chart. Any Tideda agent or other specialist engaged to set up a digitiser tablet could also set up the Charts.def file required to translate the charts.

Example: A Kent water level chart 250mm high with a 5.0 metre vertical range where 7 days duration uses 480mm of chart would have a vertical scale of 20 units/mm (5000/250) and a horizontal scale of 1260 seconds/mm ((7*86400)/480).

Strip charts – DIGITISE

This process prompts us as we digitise a strip chart starting like this >
and writes coordinates from the digitiser tablet to the Batch file.

```
Digitiser is    COM1.Gtco
Digi-Pad
Batch file is BATCH.DAT
CHART  1     STRIP   1
Digitise two points on the
strip's baseline.
```

The following is a numbered list of the prompts (in bold) with some explanations.

1. **Digitise two points on the strip's base line.** Ensure that the two points are well separated. These locate the direction of the strip's *x*-axis, and its position for graphs that reverse or have a curved *y*-axis.
2. **Digitise a point on the top line of the strip.** This coordinate is used as the upper limit of graphs that reverse, and the extent of a curved "barometric" *y*-axis. The curved *y*-axis is assumed to be perpendicular to the *x*-axis halfway across the strip.
3. **Digitise the end of the trace for the strip**.
4. **Digitise the start of the trace for the strip**.
5. **Digitise a sequence of points in time order which define the trace.** The control points are always digitised in point mode. If we have requested stream mode using the **DEVICE** process, the digitiser will be asked to enter stream mode immediately after we digitise the start of the trace. Be ready with a steady hand!
6. **Re-digitise the end of the trace**.
7. Indicate the end of digitising for that strip by pressing the secondary button on the end point of the trace.
8. If further strips are to be digitised, move the chart along the digitiser and return to step 1 as prompted. The next strip starts in the same "Reversal Region" (see next page) as its predecessor.
9. If more charts are to be digitised, press the secondary button again then respond to the prompts to get back to step 1.
10. Otherwise pressing the secondary button twice completes the process.

Any interruption results in the process ending and no data being stored in the Batch file.

Errors detected during digitising result in a message and a beep from the digitiser to alert the operator. If a cursor button other than the primary or secondary button is pressed, an error message is displayed and that point is not digitised. If the digitiser malfunctions and produces invalid values, these values are ignored and pressing RESET on the digitiser may be necessary. If reset is pushed then the digitiser will beep every time a cursor button is pressed but it may be possible to complete the current chart. The beeping will stop next time we start the **DIGITISE** process.

Various types of chart can be digitised, for example, charts from siphoning rainfall recorders; radial from barometric recorders; and reversing charts. There are slight differences in the way they are treated, but the same basic procedure is used in all cases.

SIPHONING

Charts from siphoning rain gauges show a series of vertical transitions from the top to the bottom of the chart. These occur when the catch chamber empties. Digitise the trace as we would for an ordinary chart but digitise only the top and bottom point of each siphon. Set the **Siphon** option on when we subsequently use the **TRANSLATE** process. The siphons will then be correctly interpreted.

RADIAL

Radial charts from barometric recorders have a curved y axis. We still digitise the trace as we would for a chart with cartesian axes. Process **TRANSLATE** will assume the curved *y*-axis is perpendicular to the *x*-axis halfway across the strip, use the **Pen arm radius** specified in the **CHART DEFINITION** dialog, and calculate circular time corrections.

REVERSING

Some water level recorders have reversing axes so that when the trace reaches the limit of the chart, the recorder reverses direction and the pen moves back toward the centre of the chart again. The **DIGITISE** process relies on us to tell it when the recorder has reversed. To signal a reversal, digitise the points on the trace up to and including the reversal point, and then digitise a point more than 3 mm past the reversal line. This marker point need not be at a particular time; it only has to be outside the edge of the chart. The marker point should be on the same side of the chart as the reversal point, thus if the reversal point is at the top of the chart, digitise a point above the top line. The reversal line is determined by the two control points we digitised on the base line, and by the control point on the top of the strip.

After we digitise the marker point, move the cursor back onto the trace and continue digitising as normal. Do not digitise the reversal point again, but merely digitise the next point we wish to record. When we digitise a marker point, the computer screen displays the new Reversal Region number. This shows how many multiples of the chart range we are from the start of the chart. The Reversal Region also tells us the sense of direction for the chart trace. In even-numbered regions a rising trace correspond to rising water level, and in odd-numbered regions a falling trace corresponds to rising water level.

The **DIGITISE** process always starts a chart in Reversal Region zero. It does not matter if the actual water level corresponds to this region or another even numbered region because the process will compensate. If the chart starts in an odd numbered region where the trace and the water level move in opposite directions, we must first digitise a marker point somewhere outside the range of the trace before digitising the first data point. This changes reversal regions and we can now digitise the rest of the chart.

Take care when digitising a reversing chart as it is not always easy to distinguish between reversal points on the trace and real peaks or troughs in water level. Careful reconciliation of the trace with manual staff gauge readings will generally resolve any problems.

STRIP, CHART AND BATCH TERMINOLOGY

In Tideda parlance the word batch describes any contiguous set of data that represent a continuous time series. The following terms are more precisely defined for digitising:

A **strip** is the smallest unit for digitising. A strip is short enough to fit across the digitiser tablet without having to be moved while being digitised.

A **chart** is a group of up to eight strips. Distinguishing between strips and a chart means that process **DIGITISE** will accept charts which are longer than our digitiser tablet. Strips are automatically joined together to form a continuous series of data, and we do not specify the calibration time and value where strips are joined. Each chart is given a chart number, which identifies it when we run process **TRANSLATE**. At both ends of a chart we must specify the calibration time and value.

A **batch** is a group of several contiguous charts. The calibration time and value specified at the end of one chart may be the same or before the beginning of the next chart. Thus the small time it took to change charts may be allowed for and the charts are still treated as contiguous.

To see the Batch file contents:

click the **TABLE** menu

click process **LIST**

select **List Batch file to List file**

click the **EDIT** menu

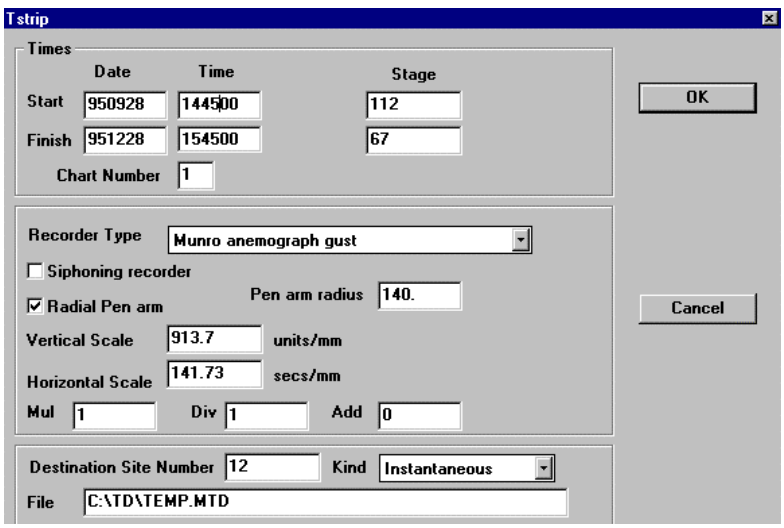
click process **LIST FILE**

We will see the five location coordinates at the beginning of each strip's data, and then after many zeroes there are the data coordinates.

WATCH OUT! We overwrite any data in the Batch file each time we start the **DIGITISE** process. Thus we must follow each use of this process with use of the **TRANSLATE** process to keep our results in a Tideda file.

Strip charts – TRANSLATE

This process reads newly digitised strip chart coordinates from the Batch file written by the **DIGITISE** process and writes it to a Tideda file.



Stage is the value of the charted quantity at the specified start and finish times.

Chart Number is used when more than one chart is digitised into the same Batch file, and then this process must be run separately for each chart. For a single chart this should be set to 1.

Recorder Type is a list of entries in the Chart.def file (see page 8.15) and selection of one of these causes the parameters defined for that entry to be entered into this dialog.

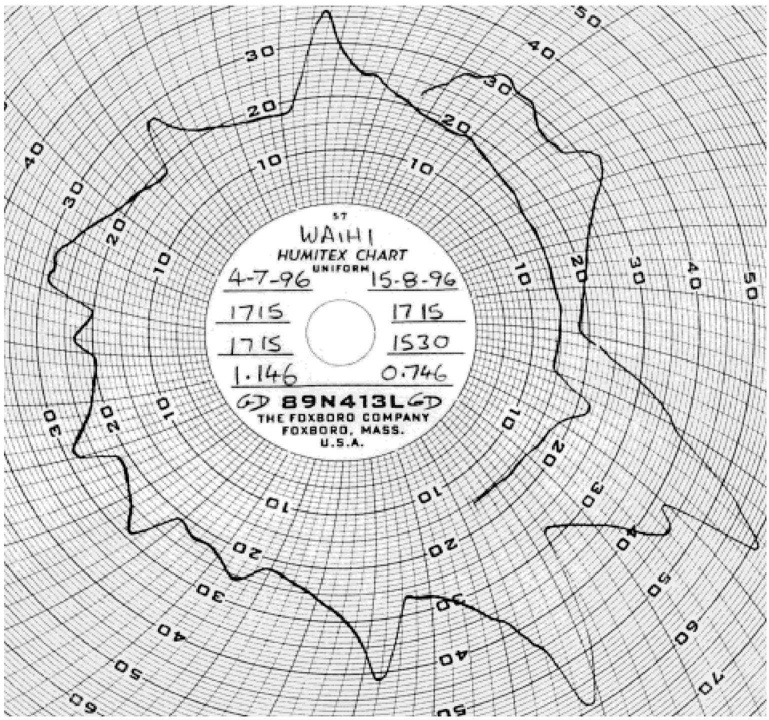
Note that this process relies on the **Stage** values supplied at each end of the chart, which are sometimes noted down at the time from an instrument that is separate from the recorder. It does not matter if a grid printed on the chart paper does not correspond to these values because the paper was fixed poorly in the recorder; in fact it does not matter if the chart paper has no grid marked on it at all.

Circular charts – CHART DEFINITIONS

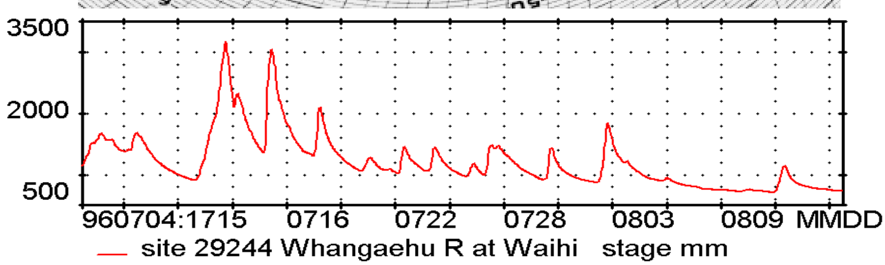
A circular chart >

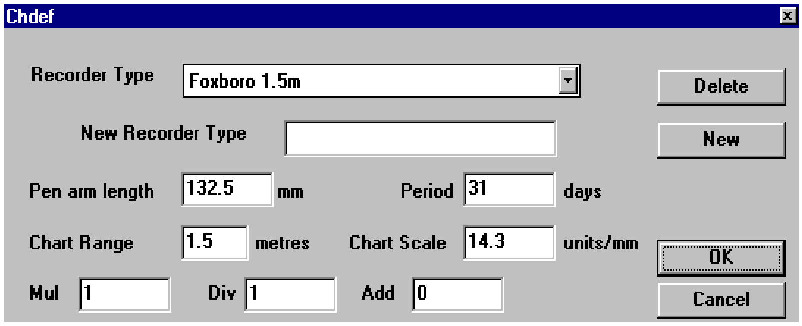
The start and finish values in chart coordinates are 23 and 15 respectively. However when digitising we specify the values 1146 mm and 746 mm observed at the time the recorder was visited.

Likewise we specify the actual time of the visits, 1715 in both cases and ignore the chart clock time of 1530.



The same data > plotted by process **GRAPH OVER TIME**.





This dialog > writes the characteristics of the particular circular chart recorder to the Charts.def file.

This is done only once for each kind of chart. Any Tideda agent or other specialist engaged to set up a digitiser tablet could also set up the Charts.def file required to use the tablet.

Circular charts – DIGITISE

This process prompts us as we digitise a circular chart starting like this >
and writes coordinates from the digitiser tablet to the Batch file.

```
Digitiser is    COM!.Gtco
Digi-Pad
Batch file is BATCH.DAT

Chart 1
Digitise four points on the
circumference of the inner
circle.
```

The following is a numbered list of the prompts with some explanations.

1. **Digitise four points on the circumference of the inner circle.** These are to locate the centre of rotation. If that centre is at a different location known by some other marks, then use a compass to draw a circle around the true centre and digitise four points on that circle.
2. **Digitise the end of the trace**.
3. **Digitise the start of the trace**.
4. **Digitise a sequence of points in time order which define the trace.** The control points are always digitised in point mode. If we have requested stream mode using the **DEVICE SETUP** process, the digitiser will be asked to enter stream mode immediately after we digitise the start of the trace. Be ready with a steady hand!
5. **Re-digitise the end of the trace**.
6. Indicate the end of digitising for that chart by pressing the secondary button.
7. If more charts are to be digitised return to step 1 as prompted

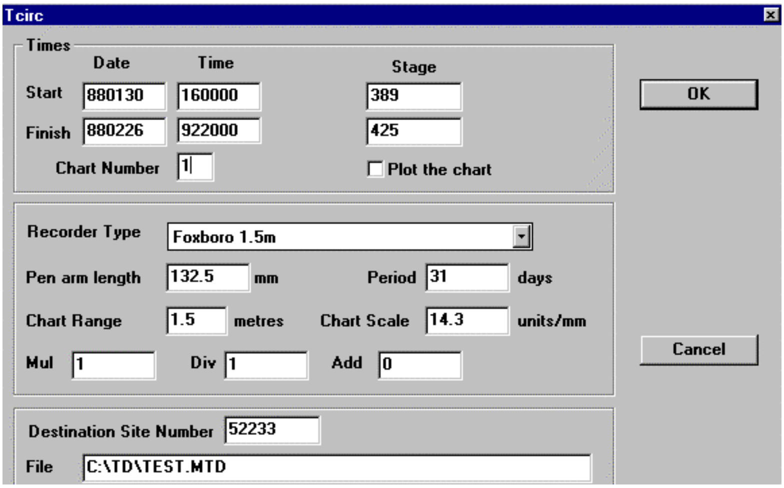
Otherwise pressing the secondary button completes the process.

Circular charts – TRANSLATE

This process reads newly digitised circular chart coordinates from the Batch file written by the **DIGITISE** process and writes it to a Tideda file.

Stage values are assigned at the specified start and finish times.

Chart Number is only used when more than one chart is digitised into the same Batch file, and then this process must be run separately for each chart.



Plot the chart graphs the coordinates of the digitised line using the same circular coordinates as the original chart. This plot can then be overlayed for visual check on how accurately the digitised record represents the original chart.

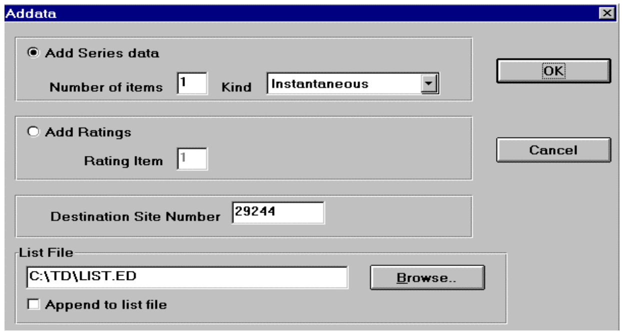
Recorder Type is a list of entries in the Chart.def file and selection of one of these causes the parameters defined for that entry to be entered into this dialog. These parameters are self explanatory, except for **Chart Scale**. This is the "vertical scale" for a circular chart and is the number of data units per millimetre along the arc of the pen movement. A different value is required for each type and range of chart. Some typical values are:

Foxboro:	9.452 units/mm for each metre of chart range (e.g. 28.356 for a 3m chart)
Cambridge:	9.294 units/mm for each metre of chart range (e.g. 46.47 for a 5m chart)
Teltherm:	7.648 units/mm for each metre of chart range (for 3,6 & 12m charts)

Note that this process relies on the **Stage** values supplied at each end of the chart, which are sometimes noted down at the time from an instrument that is separate from the chart recorder. It does not matter if a grid printed on the chart paper does not correspond to these values because the paper was fixed poorly in the recorder. In fact it does not matter if the chart paper has no grid marked on it at all, provided the centre of rotation is accurately known.

After translating a circular chart, we can plot the data again as a circular chart using the **GRAPH** menu's process **CIRCULAR CHART** (see page 5.27). This is useful for quality control because the new plot can be overlaid on the original chart and the traces should be the same. The **Plot the chart** option in the **TRANSLATE** dialog invokes this **GRAPH** process.

KEYBOARD ENTRY



Click **KEYBOARD ENTRY** to get this dialog >
then click **OK** to get prompts that assist keyboard entry of data in the layouts required by process **FULLY SPECIFIED**.
See page 8.3 for more about these layouts.

Note that the editing conventions differ from those now standard in Windows applications and so we may prefer to use a more familiar editor.

If we select **Add Series data** and click **OK** the text screen shows >

```
~~~ ADDATA ~~~
List File is LIST.ED
Do you want to enter data by
 1: Series
 2: Interval
Choose by Number:
```

If we type 1 <enter>
these 3 lines are added >

```
Undefined Items: use '?' for gap, '.' for filter
Press <End> key when complete
Enter <value> … <value>  <date> <time>
```

We now type in our data, which could look like this >

```
 2      980102          0
 1      980102      80000
 3      980102     100000
11      980102     110000
14      980102     130000
```

Then we click the <end> key and the data we entered is written to the **List file**, where in this case it looks like this >

```
2000  1   INSTANT
2    980102      0
1    980102    80000
3    980102   100000
11   980102   110000
14   980102   130000
```

To complete we use process **FULLY SPECIFIED** to enter the contents of the List File into a Tideda file.

If at step 2 we type 2 <enter>
we are prompted for a date, time and interval >.

```
Enter Start Date:-980401
Enter Start Time:-10
Enter Time Interval:-30
Undefined Items: use '?' for gap, '.' for filter
Press <End> key when complete
Enter <value> … <value>  <date> <time>
```

Now when we type in our data, dates and times are automatically entered for us.

If at step 1 we select **Add Ratings**,
we are prompted for a rating >.

```
~~~ ADDATA ~~~
List file is LIST.ED
Enter Start Date:-980401
Enter Start Time:-120000
Enter Effective Date [980401]:-
Enter Effective Time [120000]:-
Press <End> key when complete
    Unrated       Rated
         10         150
         20         400
         50        1200
```

CHAPTER 9 EXTENDED DATA ENTRY

There are many other methods for entering data into Tideda that require the use of other software.

Via the List File

Some of these programs process the data into the Tideda **List File** format, which can then be entered into Tideda using the **List to Tideda** processes described in the previous chapter. A widely available and flexible example of this is the use of Microsoft EXCEL macros. This is demonstrated by the example described in **chapter 16** of this manual.

With a small amount of programming knowledge it is possible to write a simple program to convert any well defined data format into the Tideda List file format.

Direct Data entry

It is also possible to write data directly to a Tideda data file with software that calls the Tideda database access routines. This is generally other software written by NIWA programmers as writing directly to the database requires knowledge of the Tideda software interface. Many of these programs are available from NIWA as standalone applications or as part of a larger package of products and services.

Examples of these applications are:

- **TLOGGER** – A utility program that can translate several common data logger formats, including: Unidata Starlogger, Campbell CR10, Greenspan Smart Sensor, Onset Hobo temperature logger, Hydrological Services RRDL event logger.
- **FLOSYS** – NIWA's telemetry software package for telemetry control and data aquisition from remote sites. Flosys includes automatic scheduling of data interrogations and running of external procedures, including reporting and forecasting applications. Flosys supports Aquitel, Unidata Starlogger and Campbell CR10 loggers via radio or telephone (land line or cell phone).
- **Tideda Internet Server (Tdserver/Tdclient)** – A secure client-server system for transferring data between Tideda files over a TCP/IP network (Internet). Internet transfer of computer files has become a standard procedure applicable to all kinds of computer files. For Tideda applications the source data must usually be extracted from a larger file into the file to be transferred, then at the destination this data must be merged into a larger file. Tdserver/Tdclient automates the otherwise time consuming and repetitive extraction and merging work by transferring the required data directly between the Tideda files. The main use for this system is to regularly update new telemetered data to a remote database or to download new data from a remote database.

- **Forecasting Models** – These models read telemetered data from a Tideda database and write the resulting forecast directly back to a Tideda file so that it can be used by Tideda or by another application.

For more information on any of these please contact NIWA.

CHAPTER 10 MANAGE MENU

These processes enable us to keep our data tidy.

SCAN prints a summary of a file's contents.

GAP views gaps and enables a selection to be removed.

DGAP removes gaps shorter than a set duration.

DELETE deletes data in the specified time range.

PURGE deletes older data from every site (use with care).

BATCH copies batches to another file, or deletes them.

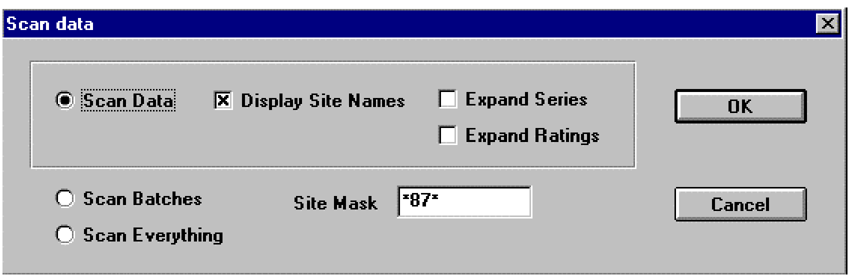
RENUMBER change the site number of existing data (use with care)

CREATE creates a new Tideda file.

RELEASE copies entire contents of the input file (see **MOVE** menu)

SCAN

This process prints a summary of a file's contents.



All the options are illustrated below.

- **Scan Data**

```
PSCAN of C:\TD\DEMO.MTD
  SITE       START  TIME        FINISH  TIME  ITEMS   KIND   KBYTES
   211     1060115 240000     1060206 240000   1    INSTANT    1.3
   226     1060115 240000     1060206 240000   1   HISTOGRM    1.3
   227     1060115 240000     1060206 240000   1    INSTANT    1.3
   297     1060116 230256     1060206  13120   1    INSTANT   14.2
   698     1030118 180000     1070710  80000    1    RATING    1.6
   698     1031229 240000     1071201  71500   1    INSTANT  131.8
   698      960106 113500     1071123 114500  15    GAUGING   10.0
   998     1060112  11500     1060206  80000   1    INCRMNT    2.8

Data Size =  163.0 K bytes  File Size=  173.5 K bytes  28 Batches in use
Free Space=25696 K bytes                               256 Batches maximum
```

Site number order, 1 line per series. The 2 line header and footer shown here also appear with the options illustrated below but have been omitted for clarity.

- **Scan Data** & **Display Site Names** & **Site Mask** = *9*

```
          lake gauge
     297    1060116 230256      1060206  13120   1   INSTANT   14.2
          river gauge
     698    1030118 180000      1070710  80000    1   RATING    1.6
     698    1031229 240000      1071201  71500   1   INSTANT  131.8
     698     960106 113500      1071123 114500  15   GAUGING   10.0
          rain gauge
     998    1060112  11500      1060206  80000   1   INCRMNT    2.8
```

Site names are the Title from the Attribute file. The Site Mask in this case has two wild card *s.

- **Scan Data** & **Expand Series** & **Site Mask** = 211

```
     211    1060115 240000      1060117  90000   1   INSTANT    0.1
     211    1060118  90000      1060205 120000   1   INSTANT    1.1
     211    1060205 150000      1060206 240000   1   INSTANT    0.1
```

One line per series batch.

- **Scan Data** & **Expand Ratings** & **Site Mask** = 698

```
   - - - - - - - -
     698    1061001 180000      1061001 210000    1   RATING    0.1
     698    1070210 230000      1070211 220000    1   RATING    0.1
     698    1070709 150000      1070710  80000    1   RATING    0.1
     698    1031229 240000      1071201  71500   1   INSTANT  131.8
     698     960106 113500      1071123 114500  15   GAUGING   10.0
```

One line per rating.

- **Scan Batches**

```
     1     211    1060115 240000      1060206 240000   1   INSTANT    1.3     0
     2     226    1060115 240000      1060206 240000   1  HISTOGRM    1.3     0
     3     227    1060115 240000      1060206 240000   1   INSTANT    1.3     0
     4     297    1060116 230256      1060206   5120   1   INSTANT   14.2     0
     5     698    1031229 240000      1071201  71500   1   INSTANT  131.8     0
     6     998    1060112  11500      1060206  80000   1   INCRMNT    2.8     0
     7     698     960106 113500      1071123 114500  15   GAUGING   10.0     0
     8     698    1030118 180000      1030118 240000    1   RATING    0.1     0
     9     698    1040108 180000      1040109 200000    1   RATING    0.1     0
    10     698    1040814  30000      1040814  70000    1   RATING    0.1     0
   - - - - - - -
```

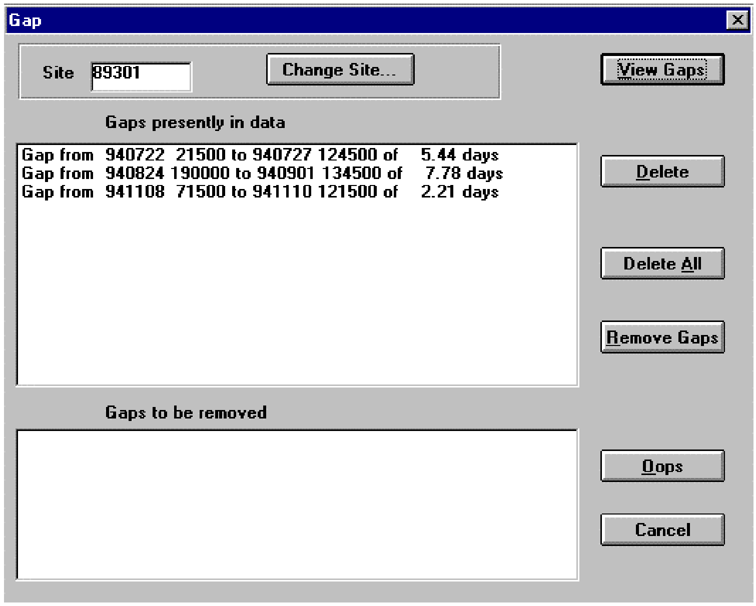
One batch per line, with the numbers at the left showing the order that the batches were written to the file. The number at the right refers to a particular file, and is only > 0 in an extended directory.

- **Scan Everything**

```
   - - - - - - - -
   2  211  1060115 240000  1060206 240000  1  INSTANT    1.3 GV   -1  0   8321
   9664
   3  226  1060115 240000  1060206 240000  1 HISTOGRM    1.3 GV    4  0   9729
   11072
   4  227  1060115 240000  1060206 240000  1  INSTANT    1.3 GV   24  0  11137
   12480
   5  297  1060116 230256  1060206   5120  1  INSTANT   14.2 GV   -1  0  12545   27120
   6  698  1031229 240000  1071201  71500  1  INSTANT  131.8 GV    8  0  27137  162112
   7  998  1060112  11500  1060206  80000  1  INCRMNT    2.8 GV    0  0 162177  165072
   8  698   960106 113500  1071123 114500 15  GAUGING   10.0  V    7  0 165121  174016
   9  698  1030118 180000  1030118 240000  1   RATING    0.1 GV   10  0 175361  175480
   10 698  1040108 180000  1040109 200000  1   RATING    0.1 GV   11  0 175489  175608
   11 698  1040814  30000  1040814  70000  1   RATING    0.1 GV   12  0 175617  175736
   - - - - - - -
```

One batch per line, with 5 codes at the right which determine how the directory works.

GAP



This process views all Gaps in a Site and enables us to select any subset and fill them in by deleting the Gap markers and so allow interpolation.

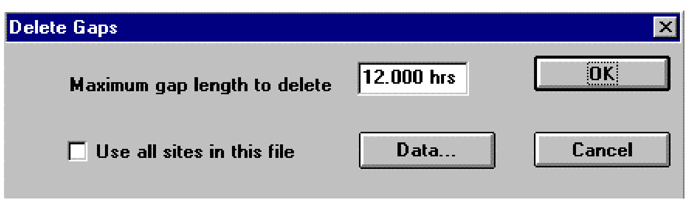
Select a gap to be removed by highlighting it then click **Delete**, and it will be moved to the lower box. **Oops** will move gaps in the lower box back up.

Finally click **Remove Gaps** to do it.

View Gaps copies the list of gaps to the text window where they can be printed, or copied to the Clipboard and thence to a spreadsheet for analysis, or whatever. In the text file they look like this:

```
Source is C:\TD\DEMO.MTD
Site   89301 Whataroa R

Data Starts at 940101      0
1: Gap from  940722  21500 to 940727 124500 of    5.44 days
2: Gap from  940824 190000 to 940901 134500 of    7.78 days
3: Gap from  941108  71500 to 941110 121500 of    2.21 days
Data Ends at 971203  71500
```

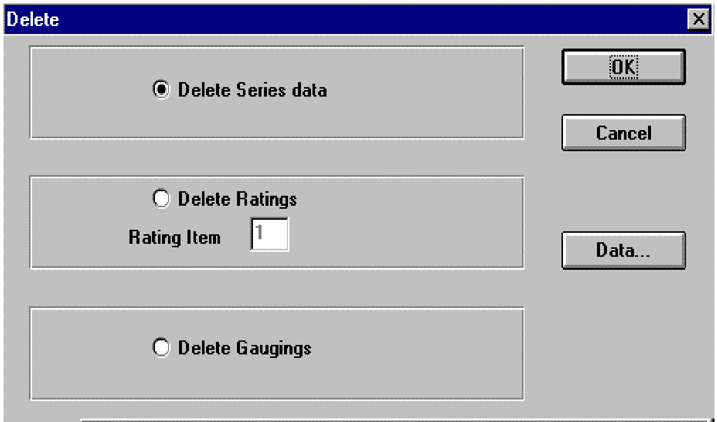
DGAP



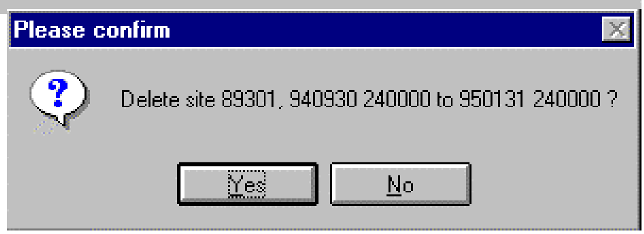
This process deletes all gap markers from a Site where the associated gap is not longer than a given duration.

DELETE

This process deletes what we specify in the **DATA** dialog



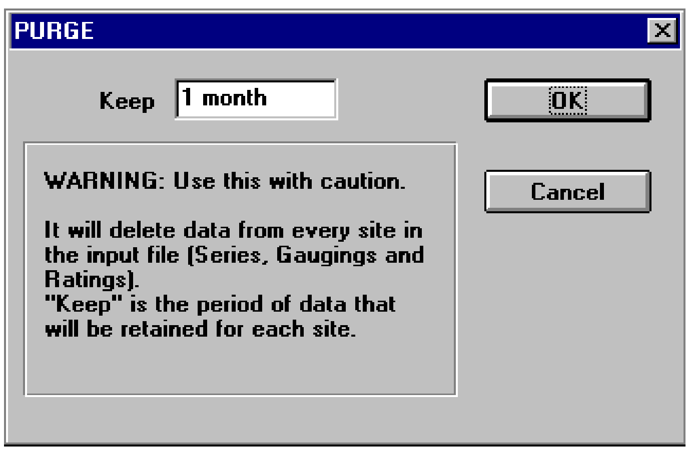
When we click **OK** we are asked to confirm >



A Gap is marked in a Continuous Series where data is deleted from it.

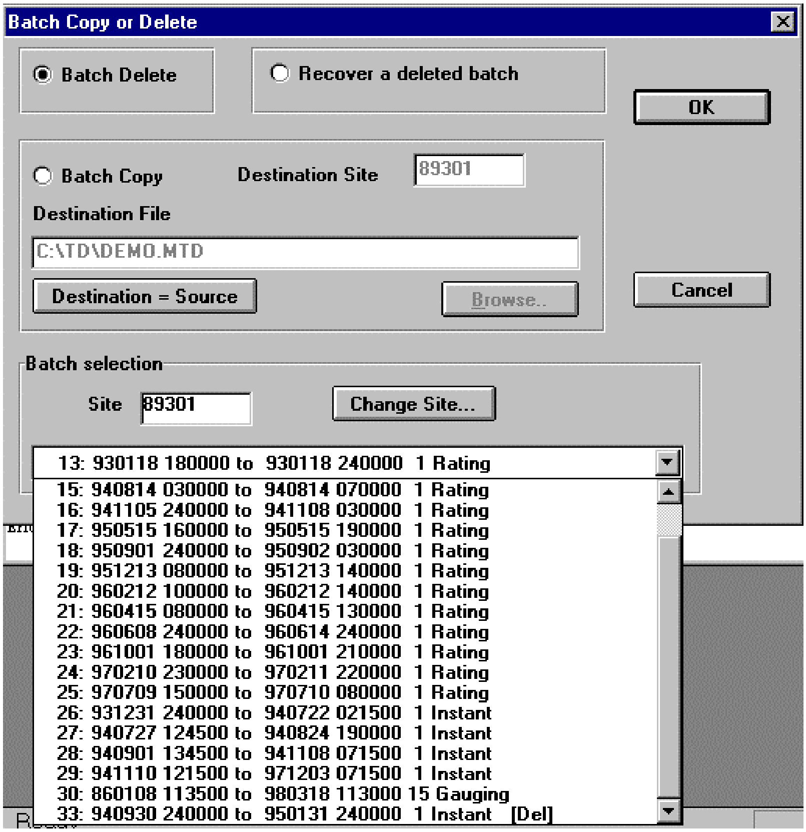
PURGE

This process deletes older data from every site in the input file. The **Keep** option specifies the period of data to be retained for each site. This can be used to easily remove unwanted data from a working file – such as a file used to temporarily store telemetered data. Take care to heed the warning in the option dialogue when running this process.



BATCH

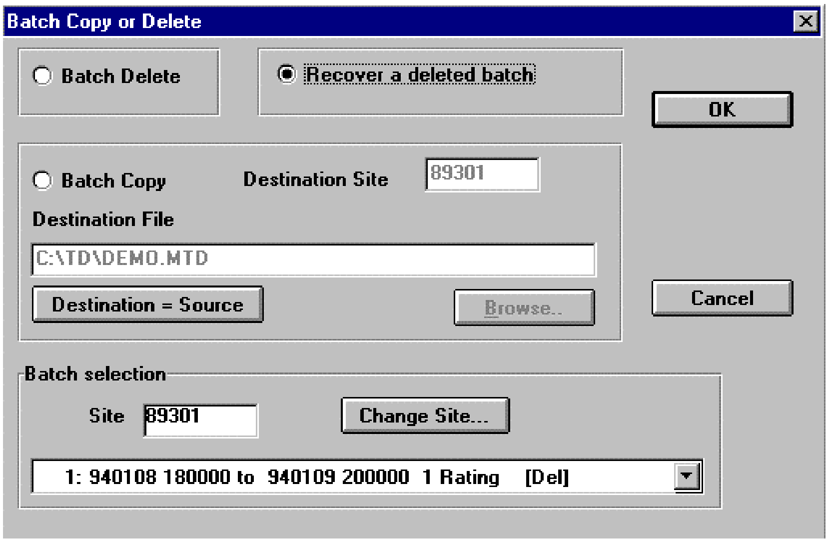
This process will copy or delete a selected Batch. The dialog includes a list of all the batches in the specified Site as follows. The **Copy** option is straightforward but the **Delete** options are more interesting as we will see.



This illustration shows a Deleting Batch at the end of the list, which is the result of using process **DELETE** on the Demo.mtd file as illustrated on the preceding page. We only see this batch when we select the **Delete** option in the Batch dialog, because this kind of batch cannot be copied or recovered with the **BATCH** process. However all other processes include the Deleting Batch as a part of the time series, and for example process Copy will move it with the rest of the series. If we select the Deleting Batch no.33, then click **OK** the file reverts to what it was before we used process **DELETE**. Try it.

To illustrate the **Recover a deleted batch** option we must first delete a real batch. Thus we might run the **BATCH** process again, select Batch no.14, select **Delete** and click **OK**. A new Gap is NOT marked in a Continuous Series where the **BATCH** process deletes data from it.

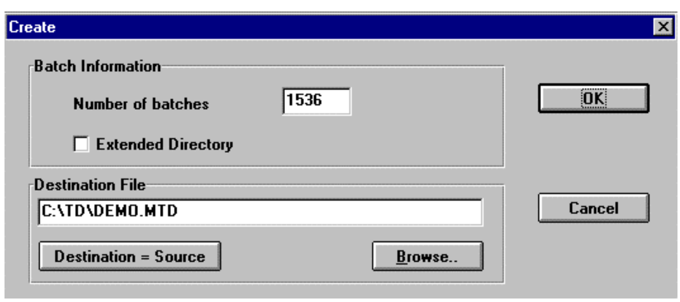
Then when we run the process yet again and select the **Recover a deleted batch** option, the dialog looks as follows. When we click **OK** the file reverts to what it was before we deleted Batch no.14.



Summarizing, when we delete records we simply label them but do not remove them. Thus they can always be recovered as we have just explained. The only exception to this recovery rule is when the **MOVE** menu's process **RELEASE** is used, and then deleted data are removed to release the storage space they occupy.

CREATE

Use this dialog to create a new Tideda file. It is also in the **FILE** menu and is more fully described on page 3.3.



CHAPTER 11 TSF LANGUAGE

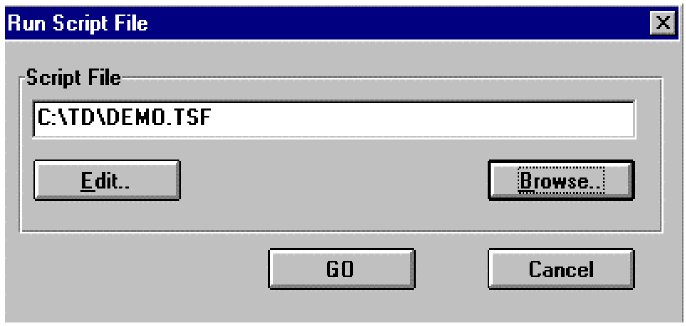
RUN SCRIPT

The **FILE** menu's process **RUN SCRIPT** causes the sequence of processes defined in a TSF file (i.e. a Tideda Script File) to be run. It is used to save time when using a sequence repeatedly.

It has this dialog >

We specify the processing in a Script file using the TSF words listed on page 11-4.

The Script file can have any name. By convention we use the suffix TSF as an acronym for Tideda Script file.

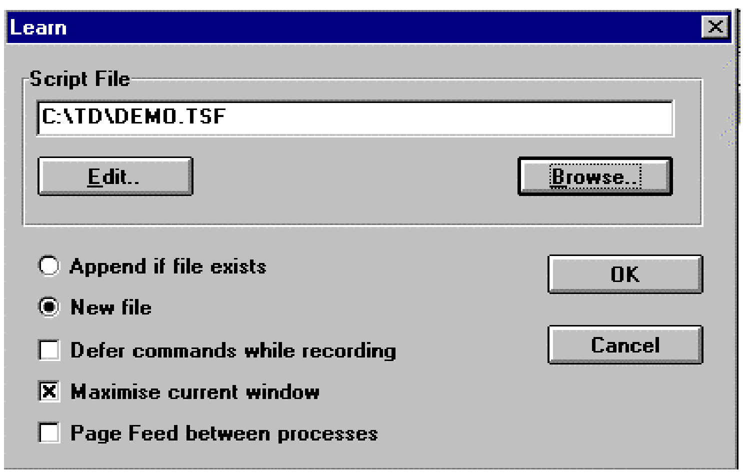


RECORD SCRIPT

The **FILE** menu's process **RECORD SCRIPT** records in a Script file a sequence of processes that we run interactively.

It has this dialog >

Thus we do not need to know about the TSF language to make a Script file, and conversely when we wish to know how the language represents a particular process we can find out by recording it as we use it.



FINISH RECORDING

The **FILE** menu's process **FINISH RECORDING** does just that, and does not have a dialog.

ARROWS <U> <D> <R>

Arrows enable a Script file to be used unchanged on different time ranges.When recording a process that has a data dialog, this dialog appears >.
A **Yes** answer causes the time range lines to be replaced by <U> which represents an up arrow and causes the available time range from the bottom of the Data dialog to be adopted at run time.

<D> or <R>, which represent down and right arrows respectively, may be typed into the Script file. They cause the time range of the previous process to be copied. <D> copies all values that define the range and <R> just one value, and may be repeated, and followed by <U> or by typed in numbers. The previous process may have been run prior to process **RUN SCRIPT**.

Then the other options are specified ending with GO. Deselected options are prefixed with N.

COMMAND LINE

The program can be run non-interactively with a Script file as the only input. To do this the command line might be:

"c:\Program Files\Niwa\Tideda\Td32.exe" @c:\tideda\working\Demo.tsf

The command can be issued in any of the following ways:

- Click **START**, click **RUN**, and type the command in the **Open** field, click **OK**.
- Type the command into a text file with file name suffix BAT, select this file with Windows Explorer then double click it.
- Select the file Tideda.exe with Windows Explorer, right click it and holding the button down drag it onto the desk top, select **Create shortcut here**, right click the shortcut, select **Properties**, select **Shortcut**, type @Demo.tsf at the end of the **Target** field, click **OK**. This shortcut will remain on the desktop and can be run whenever required.

EXAMPLE

The Demo.tsf file supplied with the Tideda program is listed on the next page. It was recorded while doing the interactive processing described in the CALCULATIONS TUTORIAL on pages 13.3 to 13.8.

Selecting the **New file** option in the Record script dialog has caused the initial settings to be included and these occupy most of the first column in the listing. Selecting the **Maximise current window** option added the WINDOW MAXIMISE statement after the initial settings.

The specifications for each process starts with the process name, usually followed by the **DATA** dialog parameters. These are the Site number and time range on 5 separate lines e.g.

```
TRANSFORM S          Process name with an S specifying the Series option.
8702                 Site number of the source data
960121               From date
223000               From time of day
960209               To date
13000                To time of day
```

DEMO.TSF

```
LSHUSH
$$$ INITIAL SETTINGS
IFILE C:\TD\DEMO.MTD
AFILE C:\TD\DEMO.ATT
OFILE C:\TD\TEMP.MTD
PLFILE DISPLAY
AUTOPEN ON
AUTOPEN PEN 1,31,3
AUTOPEN PEN 2,31,47
AUTOPEN PEN 3,31,23
AUTOPEN PEN 4,1,3
AUTOPEN PEN 5,1,47
AUTOPEN PEN 6,1,23
AUTOPEN PEN 7,31,3
AUTOPEN PEN 8,31,36
AUTOPEN PEN 9,31,26
AUTOPEN PEN 10,31,42
THRESHOLD OFF
CUSTOM OFF
FILTER OFF
PLDEF
TICKS
CALENDAR
NID
SITE
TITLE
NSCALE
UNITS
NDATE
GO
$$$ END OF INITIAL
$$$ SETTINGS

WINDOW MAXIMISE¶
TRANSFORM S
297
1060118
223000
1060206
13000
ITEM 1
NRATE
AVERAGE
INTERVAL 3.000 h
KIND HISTOGRAM
SITE 296
MUL 1
DIV 1
ADD 0
GOMERGE 1
295
227
1060118
240000
1060205
240000
MITEM 4
ITEM 1
DITEM 1
NRATE
MUL 1
DIV 1
ADD 0
INTERVAL 12.000 h
AVERAGE
KIND HISTOGRAM
GO

MERGE 2
295
1060118
240000
1060205
240000
226
1060118
240000
ITEM 1
NRATE
MUL 1
DIV 1
ADD 0
AVERAGE
DITEM 2
GO

MERGE 2
295
1060118
240000
1060205
240000
211
1060118
240000
ITEM 1
NRATE
MUL 1
DIV 1
ADD 0
AVERAGE
DITEM 3
GO

IFILE /O
PSIM INFLO.SIM
295
1060119
120000
1060205
240000
WIDTH 80
NLABEL
NLIST
INTERVAL 12.000 h
AVERAGE
ITEM 1
NRATE
SITE 295
KIND HISTOGRAM
VAR 1 0
VAR 2 0
VAR 3 0
VAR 4 0
GO
296
ITEM 1
NRATE
LEAD .000 s
NAVERAGE
GO

PEXTREME
295
1060118
240000
1060205
240000
ITEM 4
PUNITS
INTERVAL 12.000 h
MUL 1
DIV 1
ADD 0
AVERAGE
NYEARLY
NHORIZ
NUPDATE
PASS
NRATE
GO

PLGRAPH N
295
1060118
240000
1060205
240000
NVLOG
ITEM 4
PUNITS
NRATE
MUL 1
DIV 1
ADD 0
NLABEL
TOP 3
LEFT 20
HEIGHT 50
WIDTH 160
LENGTH 144
NLOW
NGAUGE
BPEN 1 62
APEN 41 3
DPEN 1 3
SPEN 1 47
VNUM
GRID
GO

PLVIEW
```

LIST OF TSF WORDS

We need only use enough initial letters of each word to distinguish it from all other words in this list. Thus Script files can be cryptic.

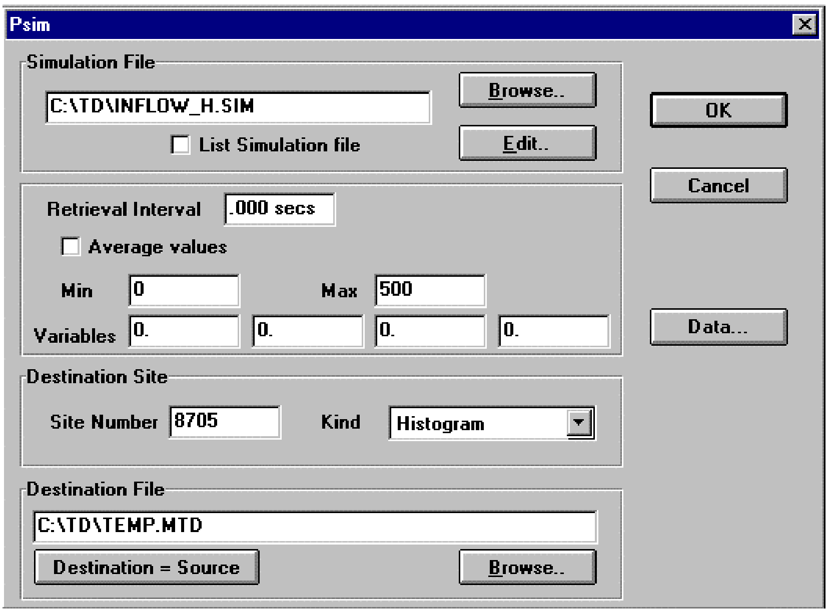
$	comment	Line in script (not within process)
ADD	option	Result=value x mul/div + ADD
AFILE	COMMAND	Attribute file name
APEN	option	Axis pen
ARMSCALE	option	Scale of wind rose arms
ARVEL	option	Area/velocity plots
ATTBUILD	PROCESS	Move - Attbuild
AUTOPEN	COMMAND	Auto pen selection
AVERAGE	option	Interp. averages reading source
BAND	option	Velocity bands for wind roses
BCOPY	PROCESS	Manage - Batch, copy sel. batch
BDELETE	PROCESS	Manage - Batch, del. sel. batch
BINS	option	Number of direction bins
BPEN	option	Background pen
BRIEF	option	Suppress header in Psummary
BSCAN	PROCESS	Table – List, batches
CALENDAR	option	Calendar date format in Pldef
CALM	option	Upper vel. for calm in Wind Rose
CFILE	COMMAND	Comment file
COMBINE	PROCESS	Move - combine
COMMA	option	date , time in Export
COMPRESS	PROCESS	Move - Compress
COPY	PROCESS	Move - Copy
CREATE	PROCESS	File - Create
CUMULATIVE	option	Graph Cumulative rainfall
CUSTOM	COMMAND	Custom axes
DAILY	option	Display daily tables in Pweek
DATE	option	Date format in Export
DATE	option	Print today top of graph in Pldef
DAY	option	Time day begins in Pday
DELETE	PROCESS	Manage - Delete
DGAP	PROCESS	Manage - Dgap
DIRECTION	option	Item for direction in wind table
DISCRETE	option	Just the data points in scatter plots
DITEM	merge	Destination item
DIV	option	Result=source x mul/DIV + add
DPEN	option	Data pen
END	COMMAND	End Tideda
ERROR	option	Gauging error bar % in Plrate
EXCEEDANCE	option	Pldist
EXPORT	PROCESS	Table - Export
FILTER	COMMAND	Sets time filter
FILTER	option	Direction filtering in wind freq. tables
FIRST	option	Minimum site to use in Pattribute
FIX	option	No scaling to fit space in tables
FLOW	PROCESS	Graph - Flow duration
GAP	PROCESS	Manage - Gap
GAP	option	Text to show a gap in Export
GAUGE	option	Plot gaugings with data plot
GRID	option	Mark a grid on the graph in Pldef
GUST	option	Item for gust speed in wind table
HEAD	option	Column headers in Export
HEIGHT	option	Height of graph y-axis (mm)
HITEM	option	Item for the horizontal of a graph
HMAX	option	X-axis maximum
HMIN	option	X-axis minimum
HORIZONTAL	option	horiz or vert limits with Pextreme
HSCALE	option	X-axis scale units/mm
ID	option	letters on Graph lines, set in Pldef
IFILE	COMMAND	Source file name
INCLUDE	PROCESS	Move - Include
INCOMPLETE	option	Char. for incomplete interval in Export
INTERVAL	option	Retrieval interval reading source
INVERT	option	Invert rating in Transform
ITEM	option	Source item number
KIND	option	Destination Kind
LABEL	option	Label
LAST	option	Max. site number in Pattribute
LEAD	option	Xget lead time in Psim
LEFT	option	Left margin mm
LENGTH	option	X-axis length mm, may be folded
LFILE	COMMAND	List file name
LIST	PROCESS	Table - List
LOG	option	Log values in distribution
LOW	option	Plot stage for low flow
LSHUSH	COMMAND	No option display running a script
LURK	COMMAND	Stop the script until key pressed
LWAIT	COMMAND	Timed wait in a script file
MARK	option	Mark data points on a graph
MAX	option	Y-axis maximum
MELEM	PROCESS	Move - Melem

MERGE1	PROCESS	Move - Merge 1
MERGE2	PROCESS	Move - Merge 2
MIN	option	Y-axis minimum
MITEM	option	Number dest. items, set in Merge1
MUL	option	Result=value x MUL / div + add
NUMBER	option	Number of items to list
OFILE	COMMAND	Destination file name
OUTSIDE	option	Select values outside of range
PAGE	COMMAND	New page for each process
PASS	option	Pass mean to Psim, in Q. Extrem
PATTRIBUTE		PROCESS Table - Attributes
PCAL	PROCESS	Table - Monthly
PCOM	PROCESS	Table - Comments
PDAY	PROCESS	Table - Daily
PDIST	PROCESS	Table - Distribution
PERCENT	option	Percent for valid value in Export
PERCENTILE	option	Percentile table in Pdist
PEXTREME	PROCESS	Table - Quick extremes
PGRAPH	PROCESS	Table - Printplot
PHOUR	PROCESS	Table - Hourly
PLANNOTATE	PROCESS	Graph - Annotate
PLBED	PROCESS	Graph - Bed plot
PLCIRC	PROCESS	Graph - Circular chart
PLDEF	PROCESS	Graph - Options
PLDIST	PROCESS	Graph - Flow duration
PLFILE	COMMAND	Plot file name = Display
PLGRAPH	PROCESS	Graph - Graph over time
PLOTTER	PROCESS	Add text or lines to a plot
PLRATE	PROCESS	Graph - Ratings
PLSCAT	PROCESS	Graph - Scatter plot
PLTLOG	PROCESS	Graph - Log of time
PLVIEW	COMMAND	Display the graph window
PLWDIR	PROCESS	Graph - Wind direction over time
PLWROSE	PROCESS	Graph - Wind rose
PMOVE	PROCESS	Table - Moving means/totals
POP	COMMAND	Restore prev. PUSHed val's in script
PRATE	PROCESS	Table - Rating
PREJECT	option	Print rejected values in Select
PSCAN	PROCESS	Manage - Scan
PSIM	PROCESS	Move - Psim
PSUM	PROCESS	Table – Summary statisticsl
PUNITS	option	Use presentation units, Data dlg.
PUSH	COMMAND	Save site's info for POP in script
PWEEK	PROCESS	Table - Weekly
PWFREQ	PROCESS	Table - Wind frequency table
PWIND	PROCESS	Table - Wind rose table
RAIN	option	Min rain for a rain day in Pday
RATE	option	Apply rating set in Data dialog
RELEASE	PROCESS	Move - Release
RITEM	option	Set rating number, in Data dlg
RUN	COMMAND	File - run DOS command
SCALE	option	Incl. scales in legend of graph
SELECT	PROCESS	Move - Select
SET	COMMAND	Set Tideda cntl options (e.g. Gapint)
SITE	option	Destination Site number
SITE	option	Incl. Site number in graph legend
SPEN	option	Synthetic data pen
STATISTIC	option	Stats displayed in weekly table
SYNCHRO	option	Export
TABLES	option	Wind Freq. tables displayed
THRESHOLD	COMMAND	Set threshold line options
TICK	option	Tick axes where numbers apply
TIMES	option	Distribute times in Pdist
TITLE	option	Incl. Title in legend in Pldef
TLIST	PROCESS	Entry - fully specified
TOP	option	Top margin mm
TRANSFORM	PROCESS	Move - transform
UNITS	option	Include units in legend in Pldef
UPDATE	PROCESS	Copy newer data, source to dest.
VAR	option	Set Psim variable
VELOCITY	option	Item for wind speed
VLOG	option	Make y axis logarithmic
VNUM	option	Label the y-axis with numbers
VSCALE	option	Y scale in units/mm
WEEK	option	Set day that week starts
WIDTH	option	Space for y-axis mm
WIDTH	option	Page width 80 or 132 char's
WINDOW	COMMAND	Min or max file window
YEARLY	option	Print extrem. in each 12 mths
ZERO	option	Midnight format in Export

CHAPTER 12 SIM LANGUAGE

PSIM

The **MOVE** menu's process **PSIM** does calculations. It is primarily for the tasks of quality control, rearrangement and presentation. Several applications are illustrated in the next chapter. It has this dialog:



Set **Retrieval Interval** to a time step, e.g., 1 h, or zero to use the time steps in the filed data.

Specify a **Simulation File** containing a text in **PSIM** language.

Set **Min** & **Max** as the range of the print-plots requested by DISPLAY statements when no other range is set in those statements.

Specify up to 4 **Variables** as initial parameters for the calculation. New values can be output at the end of the calculation for input to a following calculation.

Check **List Simulation file** to print the text in **PSIM** language on the screen, with the executable statements numbered. This code is checked every time the process is initiated and calculations proceed only if no fatal errors are found, and otherwise the erroneous statements are printed with error messages.

SIM LANGUAGE

The **PSIM** language was developed in the 1970s and its syntax is not standard. However it has a look and feel familiar to many users of Tideda, and the process is fast, and so the language will be retained for as long as it is used.

The **PSIM** process loops through its code once for each time step. The same calculations are performed each time through the loop. After the last time step control transfers to any code after an ENDLOOP statement and then stops.

The code is prepared using any text editor, and put in a file, which is conventionally named with suffix .SIM. The following code, which calculates the mean value of an unequal step time series, illustrates the layout.

```
GET v                       $$$  Get a value and call it "v"
STEP s                      $$$  Get the time step and call it "s"
sum = sum + v * s           $$$  Sum the product of value and time step
totaltime = totaltime + s   $$$  Compute total time
DISPLAY v                   $$$  Print-plot the series
ENDLOOP                     $$$  Loop back for next time

mean = sum/totaltime        $$$  Calculate mean value
PRINT mean                  $$$  Print the result
```

STATEMENTS

Statements are the building blocks of a PSIM program. Statements have a keyword, which identifies the required operation, and this is followed by parameters relevant to the keyword. The compiler checks only as many letters of the keyword as are required, and this lets us shorten the keyword. Thus DISPLAY and DISP are equivalent.

Statements are entered as lines of text using an editor. Only one keyword is allowed per line, but a statement may span several lines. The & character acts as a continuation marker and tells the compiler to treat the next line as a continuation of the present statement. The two Get statements that follow are equivalent.

```
GET VALUE

GET &
VALUE
```

Blanks are required to separate the keyword from its parameters. For clarity, we can use more than one blank, or tabs. Blanks and LABELS may appear before a keyword.

VARIABLES

PSIM manipulates our data using variables. These are symbols that we give names to and then perform arithmetic with. Variable names can be up to 32 characters long, and may include the underscore character for clarity. Variable names may have numbers in them, but the first character must be a letter. Examples of valid variable names are:

```
B
B5
WATER_LEVEL
FLOW
```

Variables may also have the same name as the statement keywords, however, this can be confusing to read. The value of all variables is initially zero.

LABELS

The PSIM language lets us build programs that can test the contents of variables and transfer control to given statement lines. A label at the start of the line identifies these lines. The syntax of a label is the same as a variable name, thus label names may contain letters, numbers, and the underscore character. The distinction is that labels always end with a colon. Examples are:

```
L1:
WRITE_DATA:   PUT VALUE
```

KEYWORDS

Descriptions of the various PSIM keywords follow. They are grouped according to their function:

```
Data access      GET, STEP, TIME, XGET, XSTEP, XTIME, XLOCK, PUT, GAP
Computation      COMPUTE, INTERPOLATE, INITIAL, VAR, FUNCTION
Program flow     IF, GOTO, SKIP, ENDLOOP, ENDPROG, CALL, RETURN
Statistics       STAT, REGRES, XSTAT, YSTAT, RSTAT, NSTAT
Text             LIST, DISPLAY, PRINT, $$$
```

DATA ACCESS KEYWORDS

GET

Get statements fetch data from the source site. Each Item in the source Site is included with the variables named in the Get statement. If data from a particular Item is not required, an asterisk replaces the variable name. The data is always in file units, not presentation units. The format of the get statement is:

```
GET <variable name> (<variable name> or *)
                    [/<gap variable>]
```

For a site with three Items the following Get statement would assign the value of Item 1 to X, discard Item 2, and assign Item 3 to Y:

```
GET X * Y
```

If the number of Items in the get statement does not match the data, a warning message is given when the simulation runs, but execution continues.

The gap variable allows PSIM to detect a gap in the source site. Usually the content of the gap variable is one, but when a gap occurs, the gap variable is set to zero. Thus if the get statement used in the previous example included a gap variable (named G), the format would be:

```
GET X * Y /G
```

Take care when getting Increment data, which is read as a rate in units per second. We must multiply the value by the time step to get the increment as follows:

```
STEP timestep                          $$$ seconds
GET rainrate                           $$$ micrometres/second
raindepth = rainrate * timestep/1000   $$$ millimetres
```

STEP

Step statements return the time step, measured in seconds, between the current value and the previous one. The first step returned is zero and can be used to detect the first calculation. The format is:

```
STEP <variable>
```

The statement to place the time step in T is :

```
STEP T
```

TIME

Time statements return the time of the current element as a date and hour. The format is:

```
TIME <date variable> [<hour variable>]
```

The date and hour variables are assigned values in the format year, month, day for the date variable eg. 980428, and hour, minute, seconds for the hour variable eg 223000. Examples of the time statement are:

```
TIME D H                 Assign date and hour to D and H
TIME D                   Assign only the date to D
```

XGET

Xget statements read data from a site other than that being read by the Get statement. This auxiliary get operates independently from the main one and may return variables at different times. The format is the same as the Get statement:

```
XGET <variable name> (<variable name> or *) [/<gap variable>]
```

For a site with three Items the following Xget statement would assign the value of Item 1 to X2, discard Item 2, and assign Item 3 to Y2:

```
XGET X2 * Y2
```

If the number of Items in the Xget statement does not match the data, a warning message is given when the simulation runs, but execution continues.

The gap variable allows PSIM to detect a gap. Usually the content of the gap variable is one, but when a gap occurs, the gap variable is set to zero. Thus if the Xget statement used in the previous example included a gap variable (named GX), the format would be:

```
XGET X2 * Y2 /GX
```

XSTEP

Xstep statements return the time step, measured in seconds, between the current auxiliary value and the previous one. The format is:

```
XSTEP <variable>
```

The statement to place the time step in TX is:

```
XSTEP TX
```

XTIME

Xtime statements return the time of the current auxiliary value as a date and hour. The format is:

```
XTIME <date variable> [<hour variable>]
```

The date and hour variables are assigned values in the format year, month, day for the date variable e.g., 980428, and hour, minute, seconds for the hour variable e.g., 233000. Examples are:

```
XTIME D2 H2                Assign date and hour to D2 and H2
XTIME D2                   Assign only the date to D2
```

XLOCK

Xlock statements force the auxiliary Xget statement to return values at the same times as the main Get statement, interpolating if necessary. Otherwise Xget can operate independently from Get, and this may not be the desired effect. Xlock is an initialisation statement, usually placed near the start of our program. The format is:

```
XLOCK
```

To retrieve two Items from the main site, and then interpolate another value in the auxiliary site:

```
XLOCK
GET  X Y
XGET B
```

PUT

Put statements write up to 15 variables to the destination Site. The format of the put statement is:

```
PUT <variable> (<variable>) [/<time offset>]
```

Examples are:

```
PUT X
PUT X Y Z
```

The time offset allows the data to be filed at a different time from that at which it was retrieved. The output time is then the current time plus the time offset given in seconds. The following example shows how to shift a time series back by 15 minutes A negative sign is not allowed in the Put statement.

```
INIT T -900
GET X
PUT X /T
```

Tideda stores only the integer part of values, and so sometimes it is necessary to multiply values by a large number eg. 1000 to preserve accuracy.

GAP

Gap statements file a gap in the destination Site. The format is:

```
GAP
```

Sometimes, our PSIM program will execute the gap statement a number of times without writing any data using the PUT statement. This is not a problem and we need not construct our program to avoid this scenario.

COMPUTATION KEYWORDS

COMPUTE

Compute statements perform arithmetic and allow us to place the result of any combination of variables in a result variable. The word "compute" is optional, and so the format can be either:

```
COMPUTE <result variable> = <arithmetic statement>
```

or

```
<result variable> = <arithmetic statement>
```

An arithmetic statement can include any combination of constants, variables, operators, and functions.

Operators specify the form of a computation. Valid operators are:

```
+       the addition operator
-       the subtraction operator
*       the multiplication operator
/       the division operator
**      the exponentiation operator
-       the unary minus operator
```

Examples of these operators are:

```
X=Y+Z           X is the sum of Y and Z
X=Y-Z           X is Y minus Z
```

```
X=Y*Z          X is Y multiplied by Z
X=Y/Z          X is Y divided by Z
X=Y**Z         X is Y raised to the Zth power
X=-Y           X is opposite sign of Y
```

Functions take variables or constants and perform pre-defined computations. Each function returns a value which can then be used in other computations. The valid functions are:

```
MOD     returns the remainder after division
MIN     returns the minimum value
MAX     returns the maximum value
ABS     takes the absolute value
LOG     takes the natural log
L10     takes the log to base ten
COS     takes the cosine
SIN     takes the sine
TAN     takes the tangent
ATAN    takes the inverse tangent
EXP     takes the value of e raised to a power
SQR     takes the square root
SGN     returns the sign
INT     returns the integer part
```

Examples of these functions are:

```
X=MOD(Y,Z)     X is the remainder after division of Y by Z
X=MIN(Y,Z)     X is the minimum of Y and Z. If Y < Z, then X=Y otherwise X=Z.
X=MAX(Y,Z)     X is the maximum of Y and Z. If Y > Z, then X=Y otherwise X=Z.
X=ABS(Y)       X is the absolute value of Y.
X=LOG(Y)       X is the natural log of Y.
X=L10(Y)       X is the log in base ten of Y.
X=COS(Y)       X is the cosine of Y.  Y must be in radians.
X=SIN(Y)       X is the sine of Y.  Y must be in radians.
X=TAN(Y)       X is the tangent of Y.  Y must be in radians.
X=ATAN(Y)      X is the inverse tangent of Y.  X is in radians.
X=EXP(Y)       X is e raised to the Yth power.
X=SQR(Y)       X is the square root of Y.
X=SGN(Y)       X is the sign of Y. If Y < 0, then X=-1, otherwise X=1.
X=INT(Y)       X is the integer part of Y.
```

Functions and operators can be combined in any desired way and parentheses can be used to force a particular order of calculation. Arithmetic expressions are evaluated in an order determined by the operators. The operators are ranked and operations of higher precedence are performed first. The order of precedence is:

```
First Precedence  **
Second Precedence unary minus
Third Precedence  * and /
Fourth Precedence + and -
Fifth Precedence  functions
```

When two or more operators are of equal precedence, they are evaluated in a left-to-right order. Examples of the use of precedence are:

```
X=A+B-C       This is evaluated as A plus B then minus C
X=A*A+B*B     This is evaluated as A squared plus B squared
X=A/B+C       This is A divided by B and the result added to C
```

When part of an expression is enclosed in parentheses, that part is evaluated first, and the resulting value is used in the evaluation of the rest of the expression. Thus the expression

```
X=A/(B+C)
```

is evaluated as A divided by the sum of B and C, whereas

```
X=A/B+C
```

is evaluated as A divided by B and the result added to C.

Expressions can be embedded inside function calls and such expressions can include other functions. Examples of this are:

```
X=SIN(A*B)     X is the sine of the product of A and B
X=SIN(SQR(B))  X is the sine of the square root of B.
```

The previous examples have all used variables, but constants can be used as required. Examples of this are:

```
X=SIN(2*A)      The sine of two times A.
X=A/(B+2)       This is A divided by the sum of B and two.
```

INTERPOLATE

Interpolate statements calculate intermediate values of the pre-defined function z = f(x) or z = f(x,y). To define a functions use the FUNCTION statement described below. The format of the statement is:

```
INTERP <z-variable> <x-variable or constant> [<y-variable or constant>]
```

The Function statement determines if the function is of one or two variables and so the number of independent variables in the interpolate statement should match. Examples of the interpolate statement for a function of one variable are:

```
INTERP Z X                 Interpolate Z=f(X)
INTERP Z 4                 Interpolate Z=f(4)
```

Examples for a function of two variables are:

```
INTERP Z X Y               Interpolate Z=f(X,Y)
INTERP Z X 4               Interpolate Z=f(X,4)
INTERP Z 4 6               Interpolate Z=f(4,6)
```

INITIAL

Initial statements initialise variables to specified values, and otherwise variables are initialised to zero.

```
INIT <variable> <constant>
```

For example:

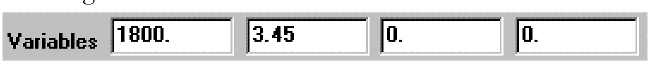
```
INIT X 2.5
```

VAR

Var statements read up to four parameters from the dialog. This allows a simulation to be repeated with different parameters without having to edit the calculation code. The format is:

```
VAR <number> <variable>
```

The following example changes the values of P and Q from one simulation run to the next. Set them in the dialog:



In the calculation code before the Endloop statement, P and Q are given the initial values 1800 and 3.45 respectively:

```
VAR 1 P
VAR 2 Q
```

OUTVAR

Outvar statements write up to four parameters to the dialog. This allows a following simulation to use the values as initial parameters without having to edit the calculation code. The format is:

```
OUTVAR <number> <variable>
```

After the Endloop statement two Variables in the dialog could be assigned the final values of P and Q:

```
OUTVAR 1 P
OUTVAR 2 Q
```

FUNCTION

Function statements define a function of one or two variables and values may be derived from it using INTERPOLATE statements. The function takes the form of z=f(x) or z=f(x,y) and is defined at discrete points. We to specify the function by way of x-z pairs for a given y-value, and separate statements are used for each y-value. To create a function of one variable, a function statement with a nominal y-value is used. The format is:

```
FUNCTION <y-value> <x-value> <z-value> [<x-value> <z-value>]
```

When entering the coordinates, we must adhere to three rules:

1. For a given y-value, at least two x-z pairs must be given.
2. For each function statement, the x-z pairs must be in order of increasing x -value.
3. The y-values must increase between function statements.

The following example defines the function z=x*x + 2*y, for x and y in the range 0 to 4.

```
FUNCTION  0    0 0  1 1  2 4  3 9  4 16
FUNCTION  4    0 8  1 9  2 12 3 17 4 24
```

Interpolate statements calculate intermediate values, using linear or quadratic interpolation depending on how many points are available. In the above example, only two y-values were needed because the z-value is a linear function of the y-value.

Function statements must be grouped at the start of the simulation and must not appear after any Interpolate statements. A list of coordinates may be continued on the next line after an & character at the end of the preceding line. An obsolete feature also works, and in this case the list of coordinates continues if the next line begins with three dots, ...

PROGRAM FLOW KEYWORDS

IF

If statements allow conditional logic to control program flow. The statement tests the value of a variable against a constant or another variable, and then jumps to a label if the test is correct. An obsolete feature also works and we can simply skip a given number of lines. The format is:

```
IF <variable> <condition> <variable or constant>  GOTO <label>
IF <variable> <condition> <variable or constant>  <skip number>
```

The condition operator can take one of the following forms:

```
LT          Less than
LE          Less than or equal to
EQ          Equal to
NE          Not equal to
GT          Greater than
GE          Greater than or equal to
```

Examples are:

```
IF X GT Y GOTO L5
IF X EQ 3 GOTO L5
L5:
IF X EQ 5 2
```

The last example causes the program to skip two lines if X equals 5. This means the next line of program that is executed is the third one after the if statement.

GOTO

Goto statements unconditionally transfer control to the line named by the given label. The format is:

```
GOTO <label>
```

To transfer control to the line labelled L1:

```
GOTO L1
```

SKIP

Skip statements unconditionally skip a given number of lines of executable statements. The skip value can be negative.

```
SKIP          2          Skip the next two statements
SKIP         -1          Execute the previous statement
```

ENDLOOP

Endloop statements mark the division between statements which work at each time step and statements which are executed only at the end. The format is:

```
ENDLOOP
```

It is required only if we have calculations to be done at the end.

ENDPROG

Endprog statements marks the division between statements in the main procedure and any subroutines. The format is:

```
ENDP
```

It is required only if we have subroutines in our program.

CALL

Call statements transfer control to a given label. Thus subroutines simply begin with a label at the start and need no other heading. A RETURN statement in the subroutine returns control to the statement following the original call statement. Thus portions of our code which we wish to use more than once can be subroutines, and can be placed after the ENDPROG statement to prevent them from being run un-called as part of the main program. The format of the call statement is:

```
CALL <label>
```

The following example is a subroutine, which interpolates a function value, does some arithmetic and returns the result in a variable called RES.

```
CALL INTF
INTF:      INTERP RES X
           RES = RES + 5 - X
           RETURN
```

RETURN

Return statements transfer control to the line that follows the most recently execute CALL statement. The format is:

```
RETURN
```

STATISTICS KEYWORDS

PSIM can accumulate the statistics for two variate data and can calculate the mean, standard deviation, and regression. The calculations are performed using double precision arithmetic and so give more accurate results than could be achieved by a PSIM program itself. This is important for linear regression of large numbers.

STAT

Stat statements pass the values of given variables into a statistics accumulator where totals are kept for subsequent calculation of statistics. The format is:

```
STAT   <x-variable>     [<y-variable>]
```

The y-variable is optional so that we can compute statistics of just the x-variable. The following example accumulates totals for linear regression:

```
GET      X   Y
STAT     X   Y
ENDL
```

REGRES

Regres statements return the linear regression coefficients between the x and y variables specified in a STAT statement: y = mx + c. The format is:

```
REGRES <m> <c> <coefficient of determination> <number of samples>
```

All the parameters in the statement are optional and can be omitted or have an * as a place holder. Typical examples are:

```
REGRES   M   C   R2  N
REGRES   M   C
REGRES   M   C   *   N
```

XSTAT

Xstat statements return the mean, standard deviation and number of samples of the x variable specified in a STAT statement. The format is:

```
XSTAT   <mean>   <std deviation>   <number of samples>
```

All the parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
XSTAT   MEAN   SD   N
XSTAT   MEAN
XSTAT   *      SD
```

YSTAT

Ystat statements return the mean, standard deviation and number of samples of the y variable specified in a STAT statement. The format is:

```
YSTAT   <mean>   <std deviation>   <number of samples>
```

All the parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
YSTAT   MEAN   SD   N
YSTAT   MEAN
YSTAT   *      SD
```

RSTAT

Rstat statements return the contents of al the statistics accumulators. This is useful when we wish to compute statistics other than the mean and standard deviation. The double precision arithmetic used when accumulating the data means that accuracy will be preserved. The format is:

```
RSTAT   <sum of x>     <sum of y>     <sum of x*x>
        <sum of y*y>   <sum of x*y>   <number of samples>
```

All the parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
RSTAT         SX    SY    SXX   SYY   SXY   N
RSTAT         SX    *     SXX   *     *     N
RSTAT         SX    SY
```

NSTAT

Nstat statements clear all the statistics accumulators to zero. It is not normally required because the accumulators are always zero when the process starts, but is useful where we are regressing part of our data and then wish to reset the accumulators and regress the rest of our data. The format is:

```
NSTAT
```

TEXT KEYWORDS

LIST

List statements print the values of variables at every time step that the statement is executed. A multiplier and a constant can scale the values. Several list statements are allowed but the values are grouped together and printed at the end of the loop. The format is:

```
LIST <variable> [<multiplier> ] [<constant>]
```

If no multiplier is given 1 is assumed, and if no constant is given 0 is assumed. The parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
LIST X                  Print X
LIST X 2                Print X multiplied by two
LIST X * 1000           Print X plus 1000
```

The printed line includes the current time. List cannot be used after the Endloop statement.

List statements can also print messages. The format is:

```
LIST '<message>'
```

An example is:

```
LIST 'FLOOD PEAK FOUND'
```

DISPLAY

Display statements print_plot the values of variables at every time step that the statement is executed. Specifying a minimum and maximum value control the range of the plot, and the values can be scaled by multiplier and a constant. Several display statements are allowed and only one print-plot is produced with all values plotted on the same line. If two values over plot at the same point, the first takes precedence. Only 1 value is printed as a number on the print-plot. The format is:

```
DISPLAY <variable> [<minimum>] [<maximum> [<multiplier>] [<addition>]
```

If a minimum or maximum value is not given, the parameter from the previous display statement is used, and if it is the first display statement MIN or MAX from the process dialog are used. If no multiplier is given 1 is assumed, and if no constant is given 0 is assumed. The parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
DISPLAY X               Display X using range from options
DISPLAY X 0 5000        Display X within a range of 0 to 5000
DISPLAY X * 4000        Display X using previous minimum
DISPLAY X * * 2         Display X multiplied by two
DISPLAY X * * * 1000    Display X plus 1000
```

The following example shows the use of two display statements. The first statement displays A and sets the range. The second one displays B using the range set by the previous statement.

```
DISPLAY A 0 5000
DISPLAY B
```

This produces the following output. The time is printed on the left-hand side and the value of A is listed on the right-hand side. The values of A and B are print-plotted along the axis defined by the minimum and maximum values.

```
830801 120000 |         |         |         |    A    B         | 3500
```

The display statement cannot be used after the Endloop statement.

PRINT

Print statements print the value of a variable after the Endloop statement, and multiplier and a constant can scale the value. The format is:

```
PRINT <variable> [<multiplier>] [<addition>]
```

If no multiplier is given 1 is assumed, and if no constant is given 0 is assumed. The parameters in the statement are optional and can be omitted, or have an * as a place holder. Typical examples are:

```
PRINT X                 Display X using range given from options
PRINT X 2               Display X multiplied by two
PRINT X * 1000          Display X plus 1000
```

The print statement does not have any times associated with it and must be after an Endloop statement.

Print statements can also print messages. The format is:

```
PRINT '<message>'
```

An example of this is:

```
PRINT 'FINAL RESULT'
```

$$$

$$$ statements are comments, have no effect on the execution, and are not counted as executable statements. Comments may also be added to the end of executable statements. Examples are:

```
$$$ This is a comment
GET X Y  $$$ Retrieve the data
```

Hydrometric gauging example

The following code checks hydrometric gaugings for validity. In New Zealand, these gaugings are filed in a standard layout as 15 Items, and this allows a number of hydraulic parameters to be stored. This code checks these values for consistency, and also checks that the gauged water level matches that stored in the associated continuous Series. Small variations are permitted because each gauging is filed at an instant of time, but would have taken a period of time to complete.

Note: This is the program listing from Tideda when the **List Simulation File** option is used – the actual SIM file source does not have the line numbers. The line numbers show executable lines of code and don't include comments or directives (such as XLOCK or INIT).

```
        $$$ To check the consistency of filed gaugings and to flag
        $$$ any gaugings which appear to be inconsistent.
        $$$ Checks are made that:
        $$$ (a) the filed water level is the same as the gauge height.
        $$$ (b) the flow equals the area multiplied by the mean velocity.
        $$$ (c) the area equals the wetted perimeter multiplied by the hyd.rad.
        $$$ The filed water level is read from the instant water level
        $$$ data for the site by the psim 'xget' command.
        $$$ the other values are read
        $$$ from the filed gauging data for the site.  no data
        $$$ preparation is needed before running this procedure,
        $$$ other than to check that both sets of data are available
        $$$ on the input tideda file.
        xlock  $$$ force xget to use the same data times as get
     1: if number_of_gaugings gt 0 goto not_first
     2: list ' '
     3: list '            listing of inconsistent gaugings'
     4: list '            ================================
   not_first:
     5: if nbad eq lbad goto get_gauging
     6: list ' '
     7: lbad=nbad
   get_gauging:
     8: get   gauge_height    flow    area    velocity  dep  slp  wid &
     8:       radius   perim   sed   temp   stg_change   meth  samp &
     8:       gaugeno $$$ read the 15 items from the gauging
     9: xget  stage $$$ get the filed recorder stage at the same time
        $$$ flow is litres/sec, distances are mm, velocity is mm/s
        $$$ so area should be in m2 to keep consistency in calculations.
    10: area=area/10000   $$$ cm2  ==> m2
   check_stage:
    11: stage_diff=abs( gauge_height - stage )
    12: if stage_diff lt 10.0 goto check_flow  $$$ allow 10mm diff.
    13: call bad_stage
   check_flow:
    14: vmean=flow/area
    15: vel_diff=abs( velocity-vmean)
    16: if vel_diff lt 1.0 goto check_area  $$$ allow 1mm/s
    17: call bad_flow
   check_area:
    18: p=perim/1000   $$$ convert wet.perim. to metres for calc.
    19: h_radius=area/p *1000   $$$ convert hyd. radius back to mm
    20: rad_diff=abs( h_radius-radius )
    21: if rad_diff lt 1.0 goto next_gauging  $$$ allow 1mm
    22: call bad_area
   next_gauging:
    23: number_of_gaugings=number_of_gaugings +1
        endloop
        $$$ print summary after all gaugings are processed
    24: print number_of_gaugings
```

```
 25: number_of_errors=nbad
 26: print number_of_errors
 27: if nbad gt 0 goto endall
 28: print 'all gaugings have passed the consistency checks'
 29: print 'you can now go and have a cold beer.'
endall:
 30: endprog    $$$ end of program
     $$$
     $$$ subroutines to display bad gaugings
     $$$
bad_stage:
 31: list '                filed stage differs from gauge height'
 32: list gauge_height
 33: list stage
 34: list stage_diff
 35: nbad=nbad+1
 36: return
bad_flow:
 37: list '                velocity not equal to flow/area'
 38: list flow
 39: list area
 40: list velocity
 41: list vel_diff
 42: nbad=nbad+1
 43: return
bad_area:
 44: list '                hyd. radius not equal to area/perim.'
 45: area_mm2=area*1000000  $$$ list area in mm2
 46: list area_mm2
 47: list perim
 48: list radius
 49: list rad_diff
 50: nbad=nbad+1
 51: return
```

Typical print out from running this example:

```
                FILED STAGE DIFFERS FROM GAUGE HEIGHT
 850226 120600  GAUGE_HEIG =    941.0000     STAGE     =    954.8257
                STAGE_DIFF =    13.82568
                FILED STAGE DIFFERS FROM GAUGE HEIGHT
 850418 122000  GAUGE_HEIG =    780.0000     STAGE     =    791.4361
                STAGE_DIFF =    11.43610
                FILED STAGE DIFFERS FROM GAUGE HEIGHT
 850815 152800  GAUGE_HEIG =    1099.000     STAGE     =    1109.421
                STAGE_DIFF =    10.42090
 NUMBER_OF_GAUGINGS = 18
 NUMBER_OF_ERRORS =    3
```

CHAPTER 13 P CALCULATIONS TUTORIAL

Discussion about calculations

SIMULATION

Archived time series of quantities which vary with the weather, such as rain and river flow, are used to simulate what would have happened if some proposed public work (e.g., a dam) had been built long ago. A simulation like this can also represent how a natural habitat has varied so that statistics can be obtained. In this way a proposal or issue can be quantified. We assume in such calculations that:

- the measurements of the effects of past weather are our best estimates of the effects of future weather; and
- the sequences in these measurements cannot be effectively characterized except by a simulation which uses the sequences exactly as measured.

CALCULATIONS BY OTHER PROGRAMS

The Tideda program is primarily for archiving time series for use by other programs that do calculations. Thus the first version in 1970 interfaced to IBM's Continuous System Modeling Program. It has been interfaced to DHI's Mike 11 river hydraulics program and to NIWA programs for calculating river flow ratings, drought frequencies, flood frequencies, lake inflows and ocean tides. It has a programmer interface which can be used to provide access to Tideda data from many other applications.

CALCULATIONS USING TIDEDA

The **MOVE** menu's process **PSIM** is for calculations. Sometimes we must use other processes to prepare data for **PSIM**. This is because a Tideda file is not a relational database that can immediately answer any kind of query. The prepared data is usually put in a temporary file.

In this chapter an elaborate example illustrates what we can do. It is presented in a tutorial style and you can replicate the example as a learning exercise using the data supplied with Tideda in a file called Demo.mtd. In this tutorial we:

- assemble data into a temporary file;
- see how the Site, Numb and Kind properties affect the arrangement of data in the file;
- make use of the differences these properties distinguish;
- calculate a series of lake inflow using measured series of outflow and level and the formula:

natural_inflow = nett_outflow + lake_area × change_of_level / time_step

(The adjective "natural" is used here to mean the part that fluctuates with the weather, and which can therefore be correlated with other natural records, and forecast.)

- manually correct an error;
- compare series using a double mass curve;
- exponentially smooth a series.

QUALITY CONTROL

The tutorial starts with measured series that fluctuate in an arbitrary manner that makes them difficult to check. They are gated flows into and out of a lake. A good check is to calculate the natural inflow implied by this data and compare it with measured flow in a nearby river subject to the same weather. We compare them with a double mass curve. This quality control calculation is also an application of the data. The best quality control methods are usually pseudo applications.

The tutorial also describes four alternative manual methods for correcting an error.

DEMO.MTD AND DEMO.TSF FILES

The tutorial uses data from a Tideda file, Demo.mtd, which is assumed to be in folder c:\td\. Its contents listed using the **MANAGE** menu's process **SCAN** are as follows.

```
PSCAN of C:\TD\DEMO.MTD
SITE     START  TIME        FINISH  TIME  ITEMS   KIND  KBYTES
     canal in
211    1060115 240000      1060206 240000   1   INSTANT    1.3
     canal out
226    1060115 240000      1060206 240000   1  HISTOGRM    1.3
     spillway
227    1060115 240000      1060206 240000   1   INSTANT    1.3
     lake gauge
297    1060116 230256      1060206  35120   1   INSTANT    6.5
     river gauge
698    1030118 180000      1070710  80000    1   RATING    1.6
698    1031229 240000      1071201  71500   1   INSTANT  131.8
698     1060106 113500      1071123 114500  15   GAUGING   10.0
     rain gauge
998    1060112  11500      1060206  80000   1   INCRMNT    2.8

Data Size=155.3  K bytes File Size=165.2  K bytes 21 Batches in use
Free Space=165728 K bytes                    256 Batches maximum
```

This demonstration data is in the 21st Century; e.g. 1060115 represents 15 January 2006.

The tutorial in pages 13.3 to 13.8, where it is described as an interactive process, is also provided as a script in a file called Demo.tsf. Demo.tsf is listed on page 11.3. The outcome of this part of the tutorial is data used in the subsequent parts, we can use process **RUN SCRIPT** with Demo.tsf as its input script if we wish to go straight to the subsequent part of the tutorial.

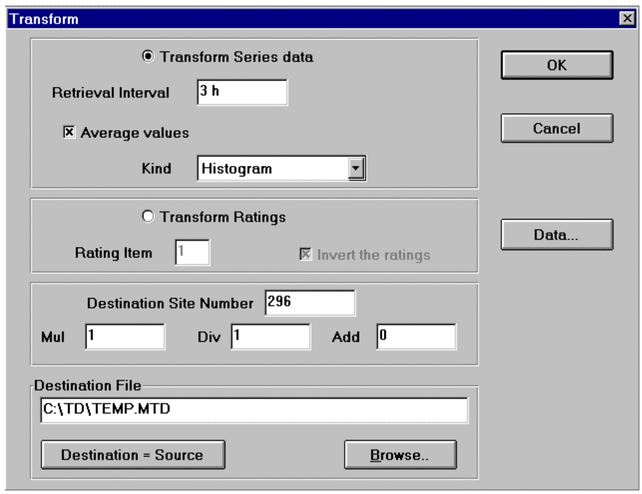
Using process TRANSFORM

We start with the lake gauge levels and calculate 3-hour averages to smooth the effect of a seiche in the lake. We file the averages as a Histogram in Site 296 on a temporary file. Subsequently we will get values from this Site at midnight and midday, and so we set the 3-hour time partition so that there are averages centered at these times by starting the calculation 1.5 hours before midnight.

Click **FILE**, click **OPEN INPUT FILE**, select Demo.mtd. This file opening process is described in much more detail beginning on page 2.1.

Click **MOVE**,
click **TRANSFORM**,
Click **DATA**,
select **Site** 297, set
From: 1060118 223000, **To**:
1060206 13000,
click **OK** to exit the **DATA** dialog.

Set the dialog thus >



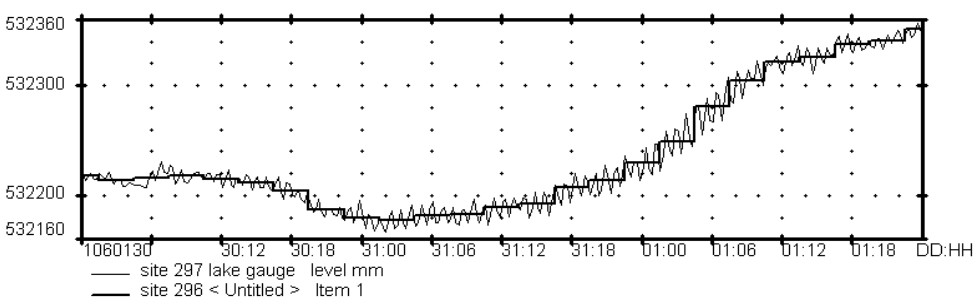
Click **OK** to exit the **TRANSFORM** dialog and do it.

Click **YES** when asked to create the new file `TEMP.MTD`.

The text window now shows:

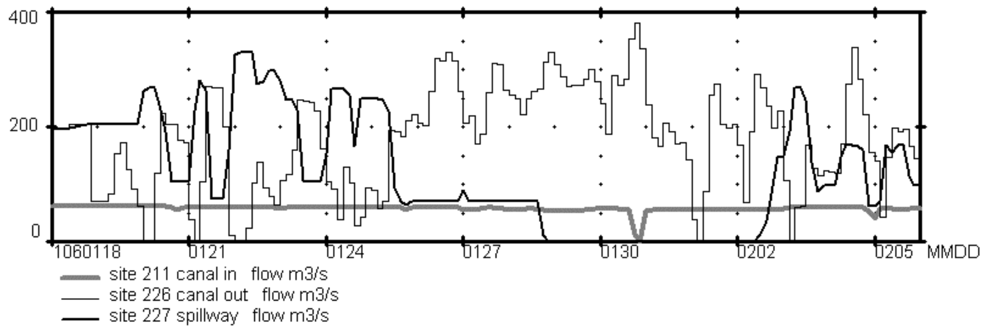
```
~~~ TRANSFORM ~~~ VER 2.5
Source is   C:\TD\DEMO.MTD Site 297 lake gauge level
Destination is c:\td\temp.mtd Site 296 lake gauge - smoothed level
From 1060118 223000 to 1060206 13000 Interval  10800
AVERAGE VALUES
```

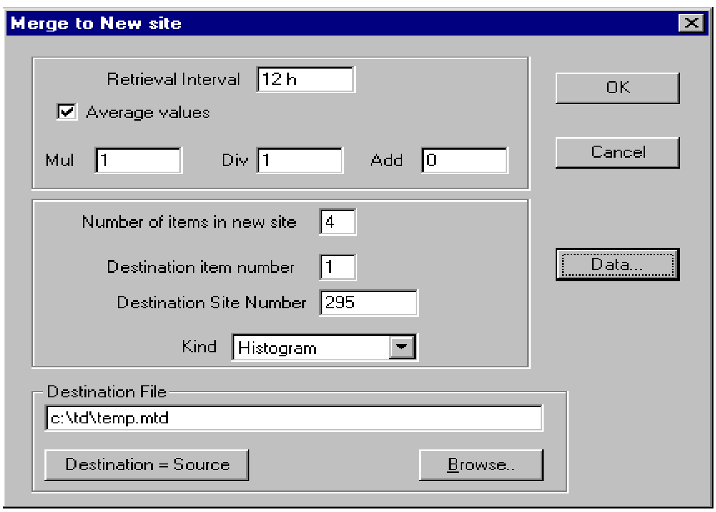
To illustrate the effect of this, part of the source record and the corresponding destination record are graphed below, showing that taking 3-hour averages has eliminated evidence of the seiche.



Using process MERGE 1

We use the phrase "net outflow" to be the sum of flows out of and into the lake that do not fluctuate in a natural way. The following graph shows 3 components: a gated spillway and two gated canals one which discharges in to the lake and the other which receives discharges from the lake. We treat the canal discharge into the lake as a negative component of the net outflow.





We now calculate 12-hour averages of these three flow records and file them as 3 Items in a temporary Site 295. We also make space in this Site for a fourth Item to hold the inflow when we calculate it.

Click **MOVE**,
click **MERGE 1**,
click **DATA**,
select site 227 (spillway)
From: 1060119 0,
To: 1060206 0,
Click **OK** to exit the **DATA** dialog. Fill in the rest of the Merge dialog as shown.
Click **OK** to exit the **MERGE 1** dialog and do it.

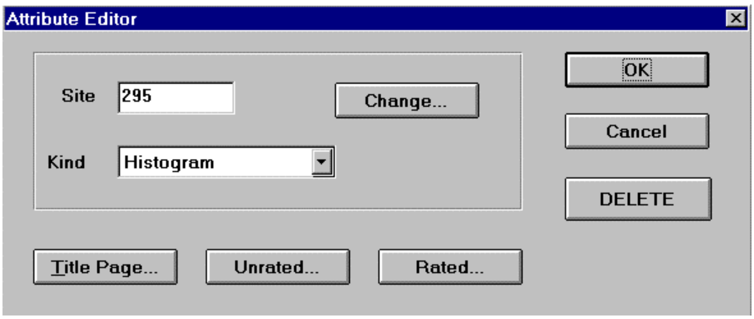
Editing attributes

We now enter Attributes for the new data in the temporary file, even though it is only temporary, to help avoid mistakes during subsequent processing.

Click **FILE**, click **SOURCE=DESTINATION** to change the input file.

Click **FILE**, click **Attribute file**, type in **temp.att** then click **open**. Ignore any "Attribute file not found" messages if temp.att does not exist yet.

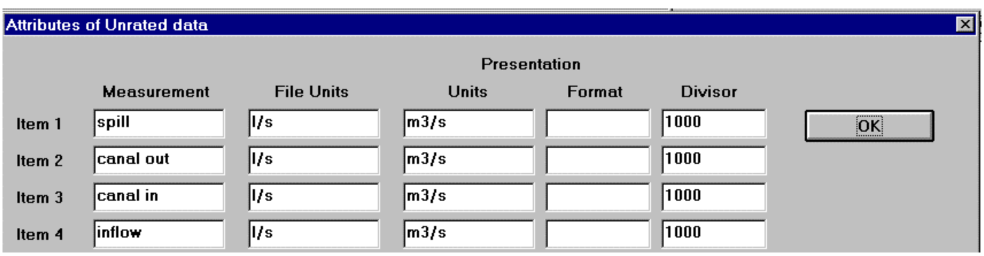
Click **EDIT**, click **ATTRIBUTES**: Click **Change**, select Site 295.



Click **TITLE PAGE**, type in Title: **lake vector** (vector is a multi-item value)

Click **OK** to exit the Title page.

Click **UNRATED**, and enter the following details:



Click **OK** to exit the Unrated dialog.

Click **OK** or **Save** to save this sttribute to our attribute file temp.att.

We also enter an attribute for the smoothed lake levels graphed above on page 13-3.

In the Attribute editor dialog Click **Change**, select Site 296.

Click **TITLE PAGE**, type in Title: **lake gauge - smoothed**, click **OK**.

Click **UNRATED**, type in Measurement: **level**, File Units: **mm**, Presentation Units: **mm**, click **OK** to exit the Unrated dialog then **OK** or **Save and Exit** to save this attribute.

Using process MERGE2

Now merge the canal-out and canal-in values into Items 2 and 3 of the four-item Site 295.

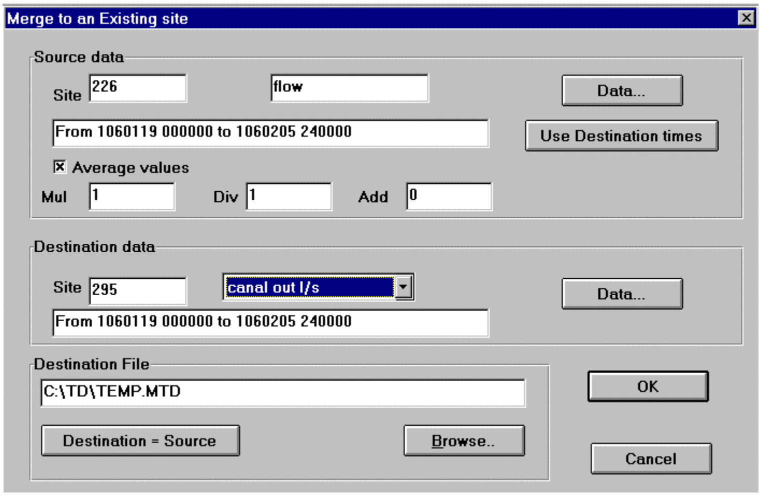
Click **FILE**, click **CHANGE INPUT FILE**, select Demo.mtd

Click **MOVE**, click **MERGE 2**

Click the Source site **Data** button and select **Site 226** (canal out), click **OK** to exit the Data dialog.

Set the Merge2 dialog thus >

Click **OK** to exit the Merge 2 dialog and do it.



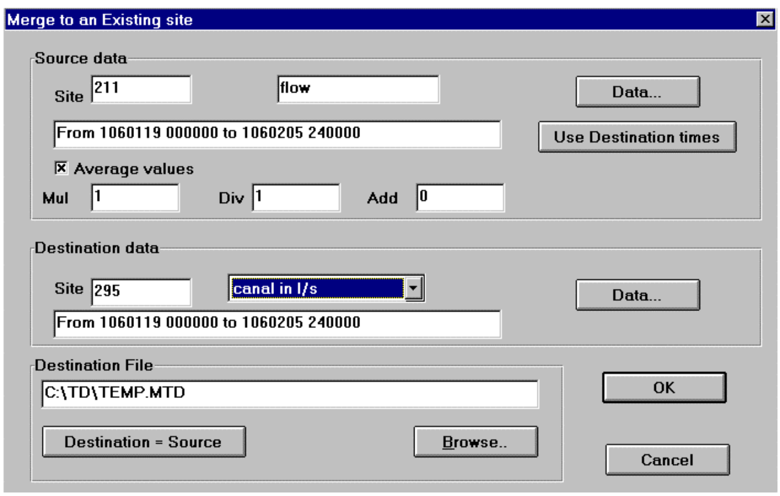
Repeat for item 3 (canal in):

Click **MOVE**, click **MERGE 2**

Click **DATA**, select **Site 211** (canal in), click **OK** to exit the Data dialog.

Set the Merge2 dialog thus >

Click **OK** to exit the Merge2 dialog and do it.



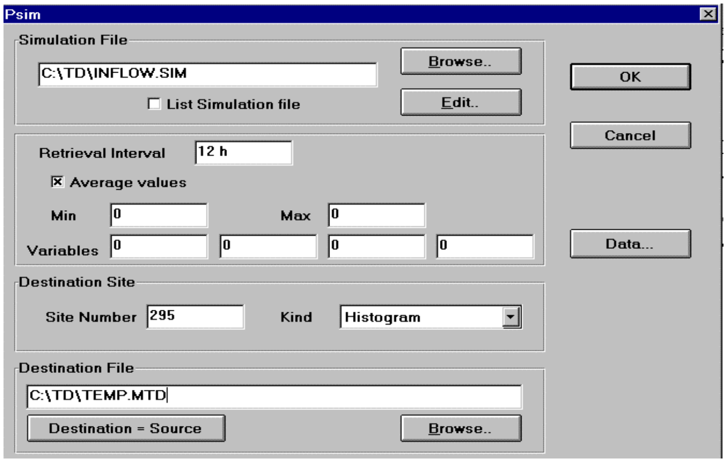
Using process PSIM to calculate an inflow Histogram

We now use process **PSIM** to calculate the natural flow into the lake using the data we have assembled in temporary Sites 296 and 295.

Click **FILE**, click **SOURCE=DESTINATION**

Click **MOVE**,
Click **PSIM**,
Click **DATA**,
set **Site**: 295,
From: 1060119 0,
To: 1060206 0,
click **OK**

set the dialog thus >



Click **EDIT**, click **YES** to open the **Simulation File** in the editor, and it should have the PSIM source code shown below. Otherwise you can type it in.

```
xlock
get   spill  canalout  canalin  *             $$$ litre/s
level0 = level1
xget  level1                                  $$$ mm
step dt                                       $$$ s
if dt gt 0 goto calc
   dt=1                                       $$$ dummy divisor
   level0 = level1
calc:
area = 169 - 0.00262 * (532500 - level1)      $$$ km2
dvol = 1000000 * area * (level1 - level0)     $$$ litres
inflow = spill + canalout - canalin + dvol / dt
put spill canalout canalin inflow
endl
put spill canalout canalin inflow
```

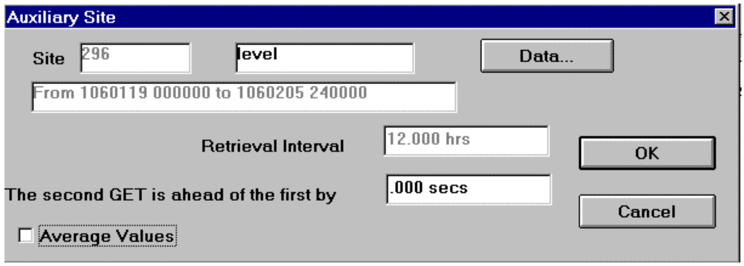
This code is reasonably intuitive, and so the following few tips are probably sufficient.

- $$$ makes the rest of the line a comment.
- **Get** reads a new value from the source file, and from the Site that was specified above using the Data dialog.
- The * after **Get** is a placeholder for an input Item we do not wish to use.
- Level0 keeps the previous value of level1 and then **Xget** updates level1.
- **Xget** and **Get** read different Sites, but the same source file. **Xget**'s Site and other parameters are specified using an Auxiliary Site dialog (see below).
- **Step** sets dt to the time since the last **Get** in seconds, and is zero first time through.

- 3 arithmetic statements after the **Calc:** label implement the inflow formula on page 13-1.
- **Put** writes the results to the destination file.
- After Put, processing loops back to the beginning of the script.

Exit the editor, saving the script file's contents

Click **OK** in the **PSIM** dialog, and this dialog appears >



Click **DATA**, select **Site**:296. Click **OK**.

Deselect **Average Values** because we want the lake levels at instants of time, then click **OK** to do it, and the text window shows:

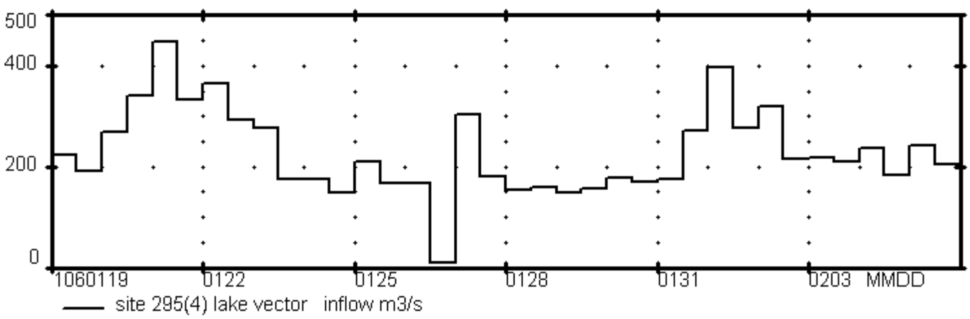
```
Name of Simulation File: C:\TD\INFLO.SIM
Source is   C:\TD\TEMP.MTD Site 295 lake vector
Destination is C:\td\temp.mtd Site 295 lake vector
From 1060118 240000 to 960205 240000
INTERVAL=  12.000hrs
AVERAGE VALUES
Second input from site 296 beginning at 1060118 240000
```

To see the result which has been written as Item 4 of Site 295:

Click **TABLE**, click **QUICK EXTREMES**, Click **DATA**, select **Measurement**: inflow, click **OK**.

Set the Interval option to zero and Update limits to Min&Max, click **OK** to do it.

Click **GRAPH**, click **GRAPH OVER TIME**, click **OK** to get this graph:



It is a good strategy to write the output of a **PSIM** run as an extra Item in a multi Item source site, as in this example, because it can then be rerun repeatedly without making other changes. This is particularly convenient when creating and debugging a new script.

A Script file called Demo.tsf that is supplied with Tideda does the processing up to this point.

It is listed as an illustration on page 11.3.

The results plotted in this graph are sensitive to apparently minor differences in the data preparation, such as specifying Kind= Instant rather than Histogram for Site 296. Thus if a difference is evident in a plot obtained interactively, to find out why: (a) rename the Temp.mtd file created interactively, (b) run the script file to make another Temp.mtd file, (c) compare the contents of the two Temp.mtd files.

Using process PSIM to change the Kind of the series

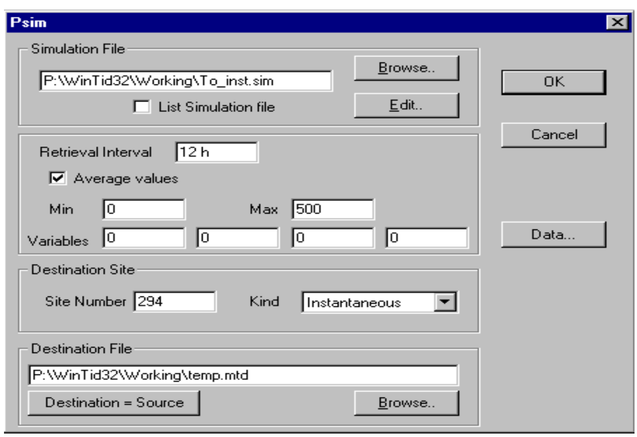
River flows are normally presented as Kind=Instant series, but the inflows we have calculated are a Histogram series as shown in the graph immediately above. To better compare inflows to river flows we change this Histogram to an equivalent Instant series using a **PSIM** procedure called TO_INST.SIM, and which is supplied with Tideda.

```
$$$ TO_INST.SIM
$$$ Gets an equal time step series of averages
$$$ Puts an Instant series at half time steps
get2=get1              $$$ Keep two previous values
get1=get0
get * * * get0         $$$ EDIT THE *S TO MATCH YOUR INPUT
n=n+1
if n ne 4 goto not4
 step dt          $$$ Do this only once at the start
 t3 =-3.0*dt
 put3= (3*get2 -get1)/2
 put put3  / t3
 t2 =-2.5*dt
 t21=-2.0*dt
not4:
if n lt 4 goto next            $$$ Do not Put for first 3 time steps
 put2=get2 + x   $$$ Do this every subsequent time step
 put21=(get2+get1)/2
 put put2 / t2
 put put21 / t21
 x=(get1-.5*get2-.5*get0)/2
next:
endloop
t2 =-1.5*dt     $$$ Do this after Get has reached the finish
t21=-1.0*dt     $$$ Have made the 2 assignments before the Get
put2=get2 + x
put21=(get2+get1)/2
put put2 / t2
put put21 / t21
t2 =-0.5*dt
put2= get1
put21=(3*get1-get2)/2
put put2 / t2
put put21
```

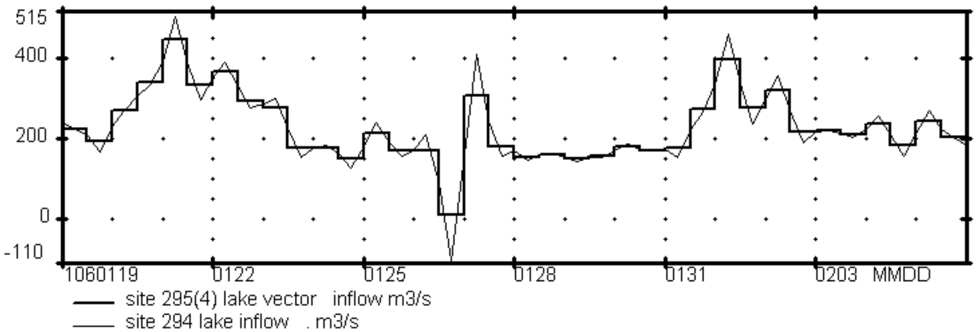
To use this procedure:
click **MOVE**, click **PSIM**,
click **DATA**, set **Site** 295
Measurement: inflow
click **OK**.to exit **DATA**.

Set the dialog thus >

Click **OK**.to run **PSIM**.



We need an Attribute for the new Site 294, which we enter as explained on page 13.5.



This graph compares the Histogram with the Instant inflow series we have just calculated.

DISCUSSION ABOUT KIND

Kind is a property assigned to a series of numbers, which determines how they are interpolated. We can simply re-label a series (using process **COPY** for instance) and it will be interpolated differently, represent something different, and look very different on a plot; see page 15.3.

Note that when a continuous graph is represented by a series of numbers, the values depend on the Kind. If we want to represent the same graph with a different Kind, we must calculate different numbers. The TO_INST.SIM script does this calculation when converting from Histogram (or Increment) to Instant.

The following code fragment of 11 statements from TO_INST.SIM is sufficient if we specify extra source data beyond the required start and finish, and then discard the corresponding parts of the result:

```
get2=get1
get1=get0
get * * * get0                    $$$ EDIT THE *s  TO MATCH YOUR INPUT
step dt
t2 =-2.5*dt
t21=-2.0*dt
  put2=get2 + x
  put21=(get2+get1)/2
  put put2 / t2
  put put21 / t21
  x=(get1-.5*get2-.5*get0)/2
```

The other 18 active statements in TO_INST.SIM just tidy up the start and finish. Rather than write this relatively difficult extra code, it is often better to simply delete the untidy ends.

To change an Instant or Increment series to the equivalent Histogram series use this code:

```
get * * * val                 $$$ EDIT THE *s TO MATCH YOUR INPUT
put val
```

To change an Instant or Histogram series to the equivalent Increment series use this code:

```
get * * * val               $$$ EDIT THE *s TO MATCH YOUR INPUT
step dt
val=val * dt
put val
```

Thus we have three code fragments to deal with all the possible changes of Kind.

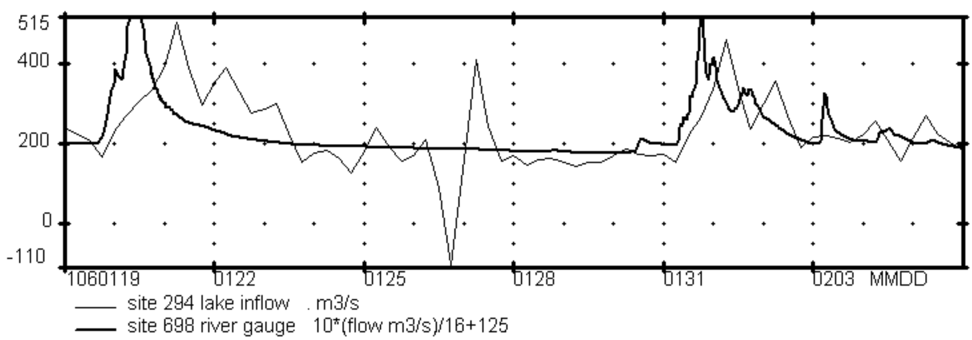
When setting the **PSIM** dialog to use these procedures:

- always select **Average Values**;
- we may set **Retrieval Interval** non zero, and must do so for TO_INST.SIM;
- we may set **Retrieval Interval** = 0 for the other PSIM procedures and then the destination has the same time partition as the source.

Checking for errors on graphs

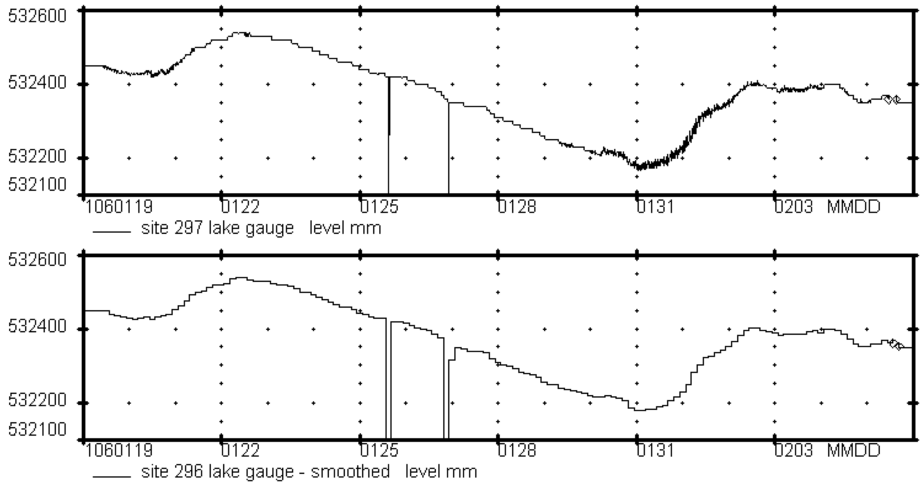
We can check the inflow record we have just calculated by graphing it with the flow. Record from a nearby river. After some trial and error we choose the following formula to get the following graph.

lake inflow = 125 + (river flow)/1.6



Melting snow has added relatively more base flow into this lake when compared to this particular river.

The wobble in the lake inflow record on 26–27 January is not matched in the river flow record and is therefore likely to be an error. We prove that it is an error by finding the cause when we graph the lake level measurements, Site 297, and the 3-hour averages we calculated, Site 296.



There are two spikes in Site 297 due to failure of the lake level sensing equipment. This affects three of the averages in Site 296. However, only the third of these erroneous averages has affected the calculation of inflow, because it is the only average with a 3-hour time range across noon or midnight.

In practice it is easy to automatically detect and ignore large spikes, but this one is insidiously small. The instrument that made Site 297 measured every minute and filed a moving average of the preceding 24 measurements every 15 minutes; which eliminated the seiche. Unfortunately this instrument occasionally failed to set a bit, and hence the downward spikes. This instrument occasionally failed altogether, and hence there was an insertion of record from another instrument for 5 days including the period plotted on page 13-3. This insertion shows an effect of a seiche because this other instrument simply records the instantaneous lake level every 15 minutes.

Manually correcting errors

We now illustrate four methods to correct the error we have found in the lake inflow:

1. We use process **LIST**, a text editor, then process **FULLY SPECIFIED** to correct Site 297;
2. We use process **SELECT** to correct Site 297;
3. We use process **DELETE** to correct Site 296;
4. We use process **GRAPH AND EDIT** to correct Site 294.

We will now illustrate all four alternative correction methods with this small amount of data. However, with more extensive data one method may work better than the others. Also, there are other methods not illustrated here. To start we copy Site 297 to the temporary file and correct it there, so that we can repeat this exercise if necessary.

Click **FILE**, click **CHANGE INPUT FILE**, select **File**: c:\td\demo.mtd, click **OK**.

Click **MOVE**, click **COPY SERIES DATA**

Click **DATA**, select **Site**: 297, click **From** and **To** buttons to select full range, click **OK**.

Set **Destination File**: temp.mtd, click **OK** to exit from the dialog and copy the data.

Click **FILE**, click **SOURCE = DESTINATION** to change the input to temp.mtd.

1ST CORRECTION METHOD

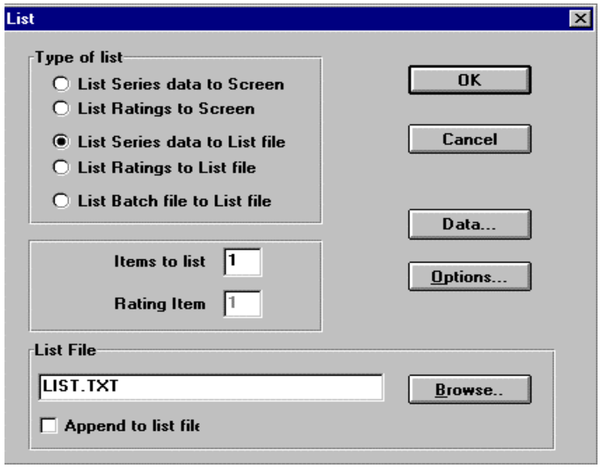
We make a text listing of the series where the errors are, edit that list to correct the errors then enter the corrected list back into the original Tideda file.

Click **TABLE**, click **LIST**

select **List Series data to List file**

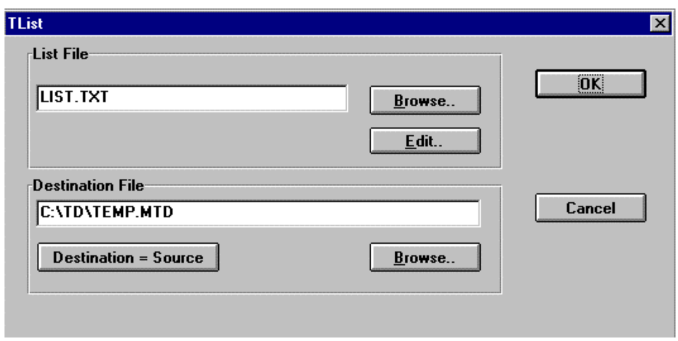
Click **DATA**, select **Site**: 297,
From: 1060126 180000,
To: 1060126 240000, click **OK**.

set the dialog thus >



Click **OK** to do it.

Click **ENTRY**,
click **LIST TO TIDEDA**,
click **FULLY SPECIFIED**
set the dialog thus >
click **EDIT**.



A text editor will open its window with the data we have selected listed in it, including the following excerpt on the left.

We delete all data lines except the two lines before and after the error value and add the NOGAP keyword to the header line so it becomes the excerpt on the right. NOGAP in the header prevents a gap from being inserted when this data is updated.

```
297           1 INSTANT
. . .
532350     1060126       220500
525100     1060126       222000
532350     1060126       223500
```

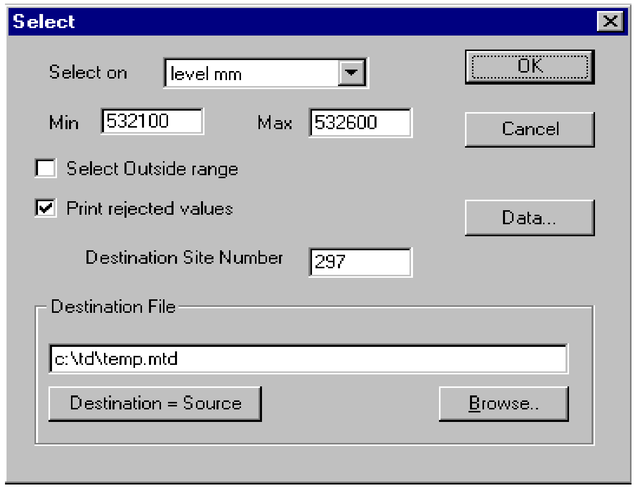
```
297           1 INSTANT   NOGAP
532350     1060126       220500
532350     1060126       223500
```

Exit from the editor, saving the contents, and click **OK** in the **FULLY SPECIFIED** dialog, which will update Site 297 without the problem spike. We can check this by graphing Site 297.

2ND CORRECTION METHOD

To prepare for the next example, we copy Site 297 from Demo.mtd to Temp.mtd again.

We now use process **SELECT**, which rejects values that are not within a valid range. This is easy to do in the case of Site 297 because the offending values lie outside the range of valid values. This is not so with Site 296.



Click **MOVE**, click **SELECT**, Click **DATA**, set **Site**: 297, **From**: all, **To**: all, click **OK**.

Set the Select dialog thus >

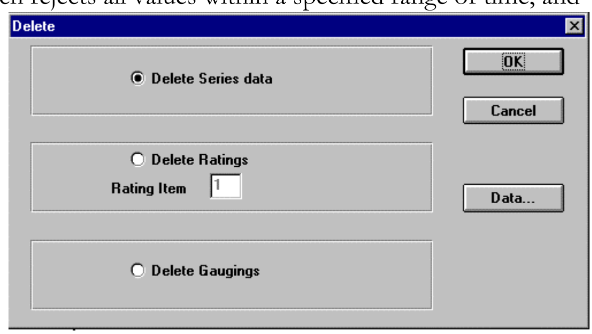
Click **OK**.

The rejected values are listed in the text window:

```
~~~ SELECT ~~~    VER 1.6
Source is      C:\TD\TEMP.MTD Site 297 lake gauge
Destination is C:\td\temp.mtd Site 297 lake gauge
From 1060125    0 to 1060131 240000
REJECTED            0 AT 1060125 152945
REJECTED       525100 AT 1060126 222000
```

3RD CORRECTION METHOD

We now use process **DELETE** which rejects all values within a specified range of time, and which is easy in the case of Site 296 because we know the time of the one offending value.

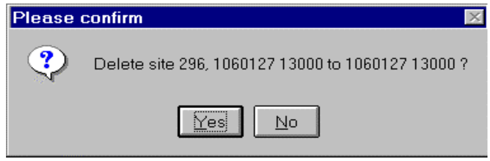


Click **MANAGE**, click **DELETE** set the dialog thus >

Click **DATA**, set **Site:** 296, **From**: 1060127 13000 **To**: 1060127 13000 click **OK** to exit the **DATA** dialog.

This process does not consider interpolation. With a Histogram the relevant file time is at the *end* of the time step.

Click **OK** in the **DELETE** dialog and this confirmation window appears >.



Click **YES** to confirm.

The text window will show:

```
Site    296 between 1060127 13000 and 1060127 13000
```

To see the effect on the file:

Click **DATA**, select **From**: 1060126 13000, **To**: 1060128 13000, click **OK**.

Click **TABLE**, click **LIST**, select **List series data to Screen**, click **OK**.

Scroll the text window, find the following, and note that only one value has been removed, but the resulting gap is 6 hours, not just 3 hours.

```
532376     1060126     193000
531789     1060126     223000
*** GAP ***
532350     1060127      43000
532346     1060127      73000
```

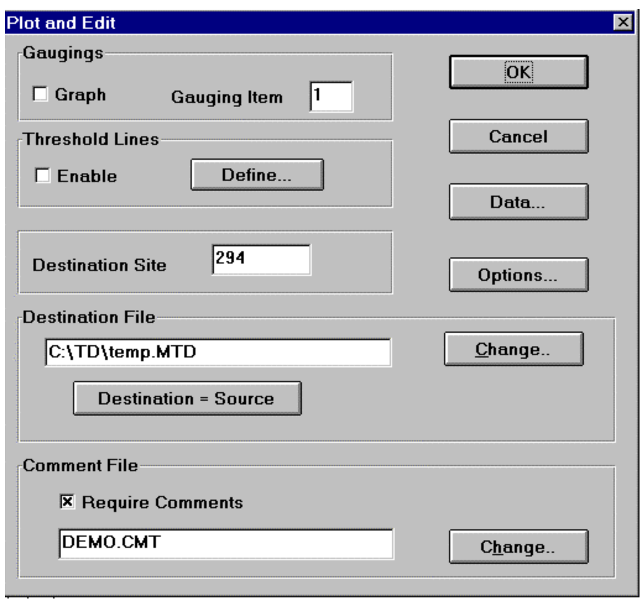
This is Histogram data so that when we calculate inflow we will get the following value, 532350, interpolated back to midnight. (Note that **PSIM**'s Get and Xget statements interpolate across gaps. The PSIM procedure can test for a gap by examining the value of an optional parameter that can be added to the statement e.g., Xget level1 /gx as described on page 12-4.)

4TH CORRECTION METHOD

We now use process **GRAPH AND EDIT** which graphs the series and we point at the bad data points with a mouse and delete them.

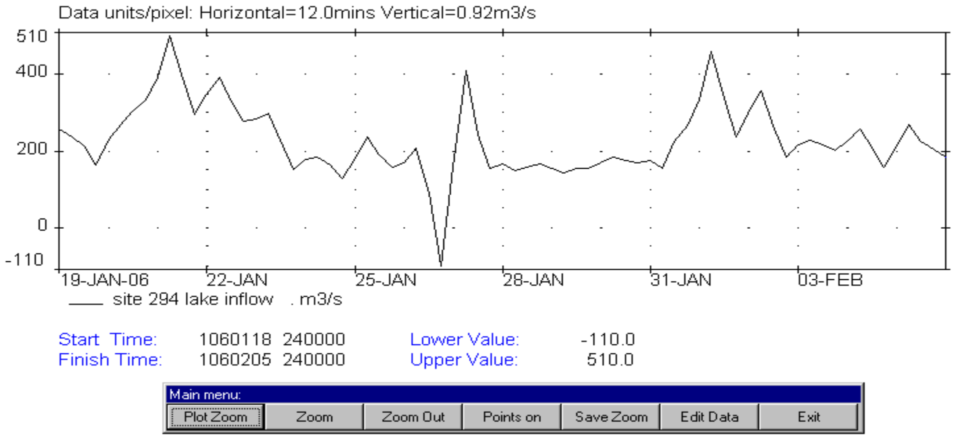
Click **GRAPH**,
click **SPECIAL**,
click **GRAPH AND EDIT**,
Click **DATA**,
set **Site**: 294,
click the **From** and **To** buttons to select all data,
click **OK**.

Set the dialog thus>

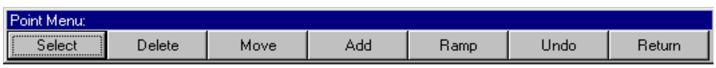


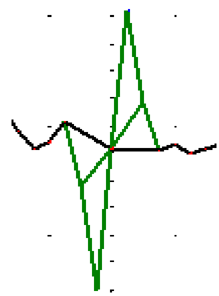
Click **OK**.

This appears in the graph window:



Click **Edit Data** then **Point Menu** to get:





Click **Select**, click near a bad data point, RIGHT click, click **Delete.**

Repeat this sequence four times to remove the "wobble".

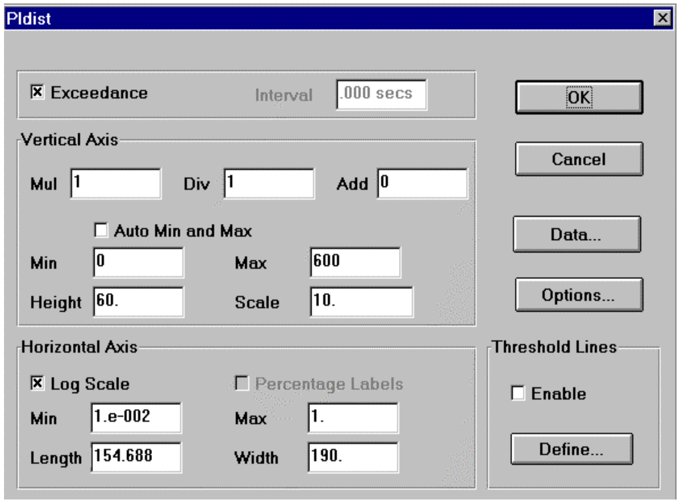
Click **Return** then **Exit** click **Save changes**, and the edited series is written to the destination file.

We selected **Require Comments** in the process dialog and so we are now asked to type in a Comment. We might type "**to remove the effect of a spike in lake level at 1060127 0**". Save this and return to the graph window where we click **Exit**.

The **GRAPH AND EDIT** process enables measurements to be massaged in ways that are difficult to detect. A subsequent user of the data who suspects this but has no details may discard all the data as unreliable. Thus this process should be used parsimoniously and encourages us to file a comment about what we do. For example the comment we entered, listed here using the **TABLE** menu's process **COMMENTS**, describes what we have just done.

```
Comment file is DEMO.CMT
Site 294 lake inflow
From  1060118 240000  to   1060205 240000

Comment at   1060118  240000
Data from  1060118 240000 to  1060205 240000 was edited by process PLEDIT,
using edit options: Delete
The data was read from site    294 on C:\TD\TEMP.MTD
and was written to site        294 on C:\TD\temp.MTD
This was done by SMT
to remove the effect of a spike in lake level at 1060127 0.
```

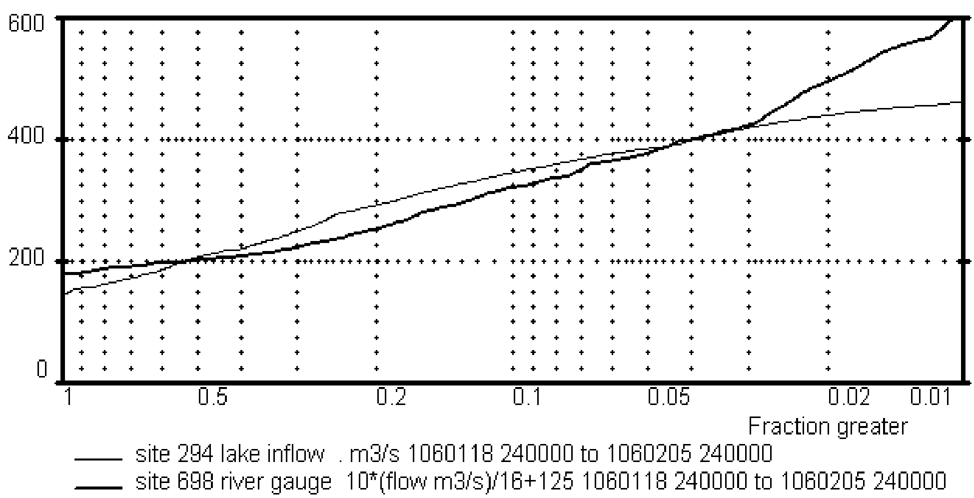
Distributions



We can check the formula on page 13.11, which relates the lake inflow to the river flow, by plotting the distribution of the two data series.

Set the input file to temp.mtd and select the **DISTRIBUTION** process from the **GRAPH** menu. Select all of site 294 with the Data dialog,set the Pldist dialog as shown an click OK to produce the plot.

Then we change the input file to demo.mtd and **overplot** the distribution of the river flow from site 698 on the inflow distribution. We get the following graph as evidence that the fit of this formula is reasonably good. We enter the values **Mul** = 10, **Div** = 16 and **Add** = 125 in the Distribution dialog to represent the formula.



Warning: Distributions can show excellent agreement between two records when **the fluctuations are not coincident in time.** Thus this is not a conclusive comparison.

CUSUMS

We can check that fluctuations are coincident in time using a cusum, which like the distribution statistic, also represents a jagged series by a smoother curve. A cusum is

the cumulative sum of the deviation from the mean.

For our example, we start by using process **TRANSFORM** to copy the relevant part of the river flow record from Demo.mtd to Temp.mtd and apply the stage to flow rating. We need to do this because when process **PSIM** subsequently reads the two Sites they must be in the same file. We end by making Temp.mtd the source file.

Click **FILE**, click **CHANGE INPUT FILE**, select Demo.mtd

Click **MOVE**, click **TRANSFORM**, click **DATA**, select **Site 698** (river gauge) **Measurement:** flow (rated) **From**:1060119 0 **To**:1060206 0, click **OK.**

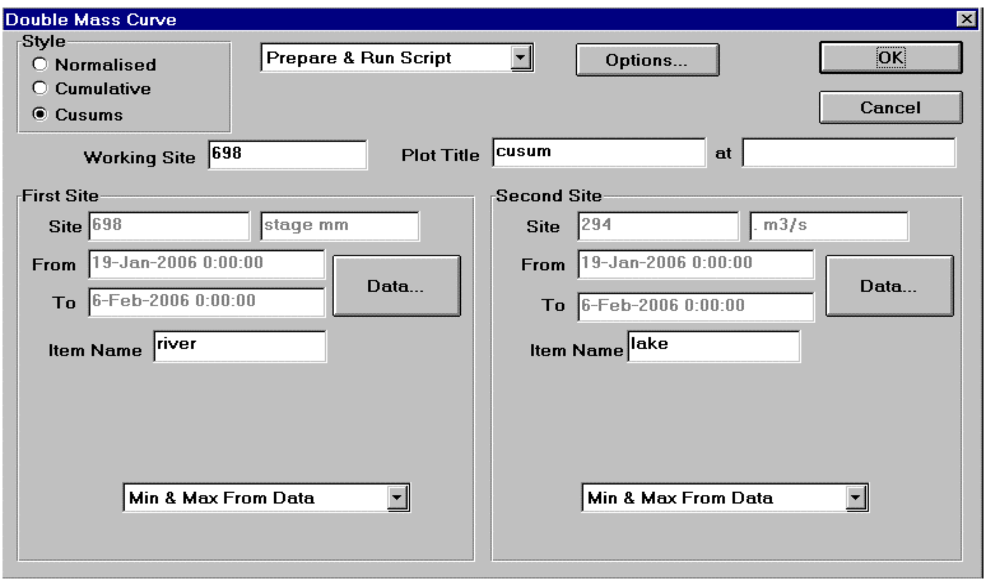
Set the **Interval** to zero and the **destination file** to temp.mtd, click **OK** to do it.

Click **FILE**, click **SOURCE = DESTINATION**

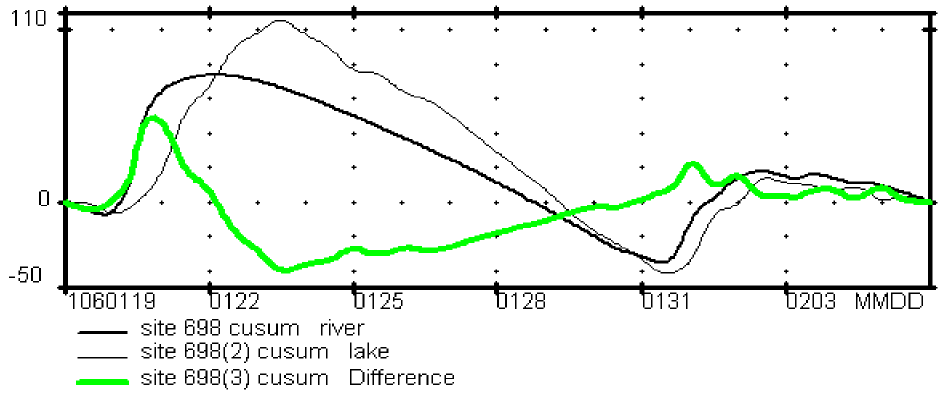
We now have a problem with attributes because the river gauge values in Temp.mtd are flows l/s but the attributes describe the "unrated" values at this Site as stage mm. We could have adopted a new Site number with a new Attribute for the output from Transform, but since this is only a temporary file and only a quality control calculation we wont bother.

We now plot the cusum which is a kind of double mass curve:

Click **GRAPH**, click **SPECIAL**, Click **DOUBLE MASS CURVE** to get this dialog.



We specify the two sites using the **DATA** dialogs then click **OK** and get the following graph.



The largest fluctuations in the Difference line are because the river cusum rises about half a day before the lake cusum. This suggests that we can forecast the lake using measurements of the river. We test this potential for forecasting by delaying the river record by exponentially smoothing it, and compare the result with lake record again.

Click **MOVE**, click **PSIM**,
click **DATA**, select **Site** 698: (river gauge) **From**:1060119 0 **To**:1060206 0, click **OK**
set **Retrieval Interval** = 3 h, deselect **Average values**, set **Destination Site** = 697 and **Kind** = Instantaneous, click **DESTINATION=SOURCE**

Set the **SimulationFile** to: C:\TD\TEMP.SIM, click **EDIT** and type in the following script:

```
ini smooth 0.85
get  a
as = smooth * as + (1-smooth) * a
n = n+1
if n gt 1 1
as = a
put as
```

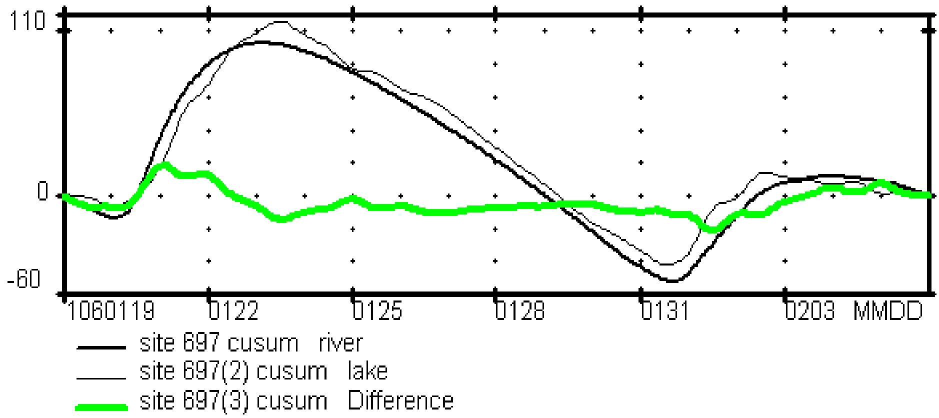
Exit the editor and click **OK** to calculate Site 697

Click **DATA**, select Site 697

Click **EDIT**, click **ATTRIBUTES**, type in **Title** = river-smoothed. Set the unrated attributes > save this attribute.

Measurement	File Units	Units	Format	Divisor
flow	l/s	m3/s		1000

We now graph the **DOUBLE MASS CURVE** of river smoothed against the lake inflow and the agreement is much better.



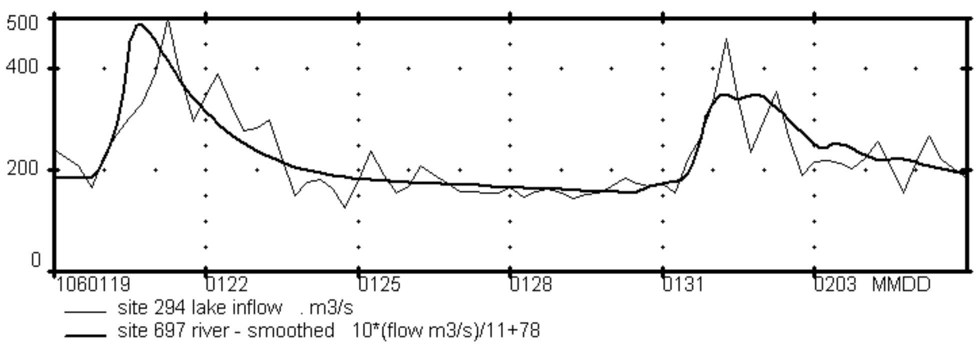
To get the formula that makes the cusums agree to this extent, we click the menu **MOVE**, click process **PSIM** and look in the dialog which will still show the following 4 **Variables** calculated by CUSUM1!.SIM:

The first two variables are the means and the second two are the standard deviations x 0.001. Thus the formula that predicts lake inflow given river-smoothed is:

inflow = 229.208 + (river-smoothed - 171.160) x 77.2 / 87.2

= 78 + river-smoothed / 1.1

We can test this formula by overplotting the predictor on the predicted as follows:



CHAPTER 14 ORGANISATION OF DATA

This chapter describes the kinds of data that can be stored with Tideda and the arrangement of the data files.

DATABASE KEYS

The first database key in a Tideda archive is to the **Tideda files**. These are named and managed by the computer operating system. They are binary files and so cannot be taken apart or joined by text processing tools, but they can be joined together, without copying, to become a single file by using a Tideda feature called an Extended directory. Each file can hold up to 2 gigabytes, and, by use of the directory, file access time has been made almost independent of file size.

The second database key is called the **Site Number**. A site number is assigned to each location where data were collected, and Tideda then uses this number as a key to stored data from that location. Site numbers are integers in the range 1 to 8,388,607.

The third database key distinguishes the three series objects that can coexist at each Site: **Continuous Series**, **Gaugings** and **Ratings**.

The fourth database key is **Time** and every value in the archive has an associated time. Times are represented in the files as number of seconds since the datum time (giving a time resolution of one second). However users never see these numbers because the user interface always represents each time in terms of: year, month, day, hour, minute and second. The time range for existing files (at the date of publication) is 1 January 1872 to 31 December 2007. This will be increased greatly in future versions of Tideda.

A fifth database key for Continuous Series and Gaugings identifies the **Item** when the value is a vector (multi-item value), and for Ratings identifies which **Rating Item**. This key can have integer values in the range 1 to 15.

A database with only five keys is relatively flat, but even so it is not necessarily simple. Tideda treats collections of the data values as "objects" that have "properties" assigned when the data are captured. The properties determine how data processing "methods" are subsequently applied.

SITES

A **Site** can consist of up to 4 components:

- Continuous Series
- Gaugings
- Ratings
- Attributes

The first three of these components are stored together in the Tideda data file. Attributes are stored in a separate file and are described below in the section headed Related Files.

The **Site number** property of a site is an integer number in the range 1 to 8388607. This number is a key and so must be unique within any one Tideda file to just one object, but appears many times in the file to identify the Site's many components.

Each Site may include only one Continuous Series which has unique properties, such as its Kind property described in the next section. Thus if two different Kinds of continuous data are sourced from one location, they must be stored as two different Sites. For this reason the Tideda word "Site" is not a strict synonym for the ordinary word "location".

Continuous Series

Tideda is designed for data that are collected continually. When a measurement is repeated and the time of each measurement is recorded, a list of measurements results that we call a **time series**. The series is continuous if the measurements are sufficiently frequent that we can estimate valid values at intermediate times by interpolation between one measurement and the next. A **Continuous Series** in Tideda is such a series.

The **Site number** of a Continuous Series is the key to its Site.

The **Number of Items** property of a Continuous Series object is the number of quantities measured at each time. Many series have only a single Item at each time, particularly the data from field recorders. However Tideda can store vector values of up to 15 Items; and these are most often used when processing results from mathematical models, but there are also field situations where vector values are used.

Field recorders that measure two or more quantities give rise to the following choices when assigning the Site and Number of Items to new data.

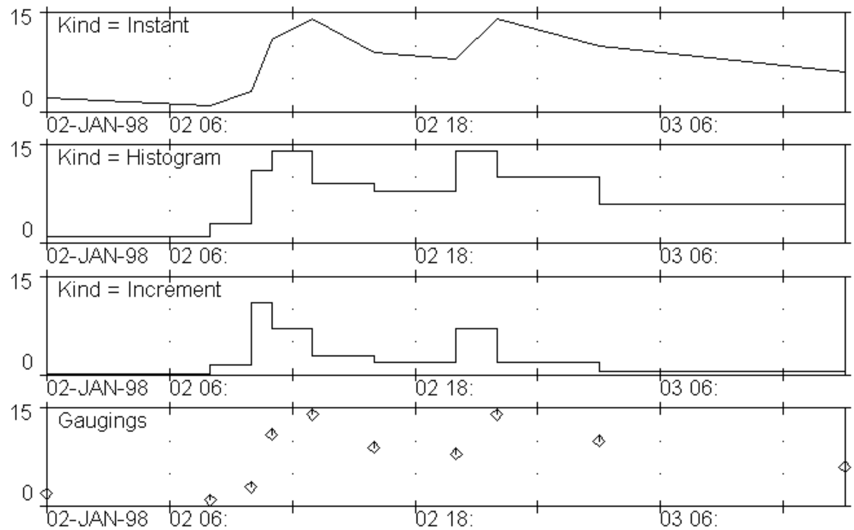
Where one quantity (such as water level) is archived and the other (such as battery voltage) has only transient relevance, assign separate Sites with Number of Items equal to one so that the transient quantity can be easily discarded after its "use by" date.

Where the time resolutions appropriate to the quantities are different, for example water temperature is sampled hourly and water level is sampled every 15 minutes, assign separate Sites with Number of Items equal to one so that storage space can be saved by discarding unnecessary data.

Where two quantities are subsequently analysed together, for example wind speed and direction which is subsequently filtered to obtain the distribution of speed classified by direction, assign a single Site with Number of Items greater than one.

Where computer code in the field can be minimised by telemetering a vector from field equipment which contains data from several sensors at the same time, assign a Site solely for use in the telemetry link with Number of Items greater than 1. Then use Tideda to repack the data into separate Sites at the archiving computer. Field computer code is usually more difficult to maintain and in a more obscure language than Tideda code. In this case the Kind property (described immediately below) can be chosen arbitrarily because the data are separated into different Sites before the Kind property becomes relevant.

The **Kind** property of a Continuous Series object specifies how Tideda will interpolate between the times in the file. There are three Kinds of interpolation, and Gaugings which are not interpolated:



This graph shows the differences when the same values (tabulated below) are filed as Kind = Instant, Histogram or Increment or as Gaugings respectively. The Increment data are graphed here with Time unit = 1 hour.

Item value	2	1	3	11	14	9	8	14	10	6
Time filed (hours)	0	8	10	11	13	16	20	22	3	15

Kind = Instant data are the most common and originate from recorders which make measurements at instants of time, e.g., of water level. Interpolated values graph as sloping lines.

Kind = Histogram data have constant values over each period of time, e.g. spillway gate openings, and the data are interpolated so that they plot as a histogram. In cases where averages are computed and stored, they are usually labeled Kind = Histogram.

Kind = Increment data have each value measuring an accumulation over a period of time, e.g. rainfall. Interpolated values at intermediate times represent a rate of accumulation and graph as a histogram. The numerical value of the rate depends on a time unit which users must specify e.g. the same rain rate can be presented as mm/hr or mm/day. (Note that this particular time unit is not a property of the data.)

Thus the quantities captured are stored as data values without doing any arithmetic on the numbers. In fact the same information can be represented using the other Kinds, and the only reason for providing a range of Kinds is to reduce the decisions made when data are captured. The arithmetic to convert to other kinds is delayed until an application requires it.

Gaugings

Gaugings are a time series of measurements taken at such infrequent intervals that there is no valid interpolation for times in between. Gauging data are graphed as symbols; see the previous figure.

There may be many different gauging measurements. For example at the Site of a continuous water level record, these measurements might be river flow, suspended sediment load and water temperature. They are stored as separate Items, and it is customary to add as the first Item the value of the continuous quantity at the time of each gauging. Any Item that was not measured at the time of a particular gauging is given a special value called “undefined” in the vector value of that gauging.

The **Site number** property of a Gaugings object is the key to its Site.

The **Number of Items** property of a Gauging is the number of quantities measured at each time, and is the same for each gauging of that Site.

Rating Curves

A **Rating Curve** (or **Rating**) is the transformation from a measured quantity to another related quantity. It is a calibration of the measuring instrument but instead of being part of that instrument it is stored with the records from the instrument. There are two reasons for this:

- where the calibration is determined retrospectively some time after the measurements are made; and
- where it is preferable to use standard instruments in the field, for instance to facilitate servicing by replacement, and to keep a record of site specific calibrations where it can be checked when the data are used.

For example, water levels are recorded at many hydrological recording sites where it is a river flow record that is required. Relatively few flow measurements are made to determine the relationship between water level and flow. The graph of flow against water level is called the rating curve, and a value of flow for any water level is read off this curve.

- Many rating curves can be filed for a single Site:
- A Rating Series is a time series of Rating Curves.

Each rating curve is filed as pairs of (Unrated,Rated) points which define the transformation when quadratically interpolated. A Tideda rating consists of between 3 and 50 pairs that should span the range of the measured (unrated) quantity. Values outside of this range will be extrapolated, but interpolation within the range is more reliable.

The **Site number** property of a Rating is the key to its Site.

The **Rating number** property of a Rating series is a key and is a number in the range 1 to 15 which distinguishes it from other Rating Series at its Site.

For example a Continuous record of lake level might be transformed into two different quantities:

Rating 1 might transform it to the record of river flow discharged through the lake outlet;

Rating 2 might transform it to a record of lake level relative to mean sea level by adding the differences between the local datum and mean sea level, which may vary from time to time.

The **Data Item number** property of a Rating Series is a number in the range 1 to 15 that identifies the Continuous Series Item that it transforms. It is assumed to be 1 and can be assigned another value in the Attributes of that Site. This is a minor exception to the general rule that Attributes are optional, because there is no other way to make this assignment.

The **Time** property of each Rating Curve is the Time key that determines where the rating is in its series. It is also called the start time.

The **Effective Time** property of each Rating Curve object is a second time on or after the start time. The preceding rating applies up to the start time of the new rating, then between the start time and the Effective Time the rating is smoothly changed so that it becomes fully in effect from the Effective Time. The interpolation method for this smoothing is described on the next page.

DATES AND TIMES

Date and time are represented as numbers. The date is presented as "YYMMDD" if after 1900 and before 2000, as "-YYMMDD" if before 1900 and as "YYYMMDD" if after 1999, where:
YY is the year in the 1900's or 1800's and YYY is the year in the 2000's with a leading "1"
MM is the number of the month
DD is the number of the day
For example:

3 December 1885 is represented as –851203
30 October 1983 is represented as 831030
19 August 2003 is represented as 1030819

Note, however, that the first 8 days of 1900 are written as –101,-102,…,-108, to avoid conflict with the abbreviated form for the years 2000 to 2008, which are 100,101,102,… 108, as explained below.
It is acceptable to omit the month and year when referring to the beginning of a year. Thus 1875 and 2005 can be abbreviated as –75 and 105 respectively.

The time of day is presented as HHMMSS where:

HH is the hour (0-24)
MM is the minutes past the hour (0-59)
SS is the seconds past the minute (0-59)

For example:

9.15 am is represented as 091500
25 seconds later as 091525
9.15 pm is represented as 211500

The start of day can be entered as 0. For example 951130 240000 and 951201 0 are equivalent.

INTERPOLATION

Compression

The times at which measurements are stored need not be evenly spaced. This allows compression of a Continuous Series by removal of intermediate values which can be deduced by interpolation. With Kind=Instant, such values are the middle of three which, when interpolated form a straight line as follows:

```
Uncompressed:
Value
|   *             *
|     *         *
|       *   * *
|*       *        Time
```

```
Compressed
Value
|   *             *
|
|               * *
|*       *        Time
```

Compression is important because it significantly reduces the size of storage space occupied by data. Smaller files are cheaper to own and transfer, and data are processed more quickly. When retrieving data from a file, the interpolation calculation is always invoked, so there is no speed benefit in keeping uncompressed data in a Tideda file.

The Compress process allows a tolerance to be specified. Intermediate values are omitted when the resulting graphed line lies within this tolerance of the original graphed line. The tolerance used in compressing data should be consistent with the resolution of the recording instrument. If we compress data with zero tolerance, they will always be retrieved exactly as they were originally.

When a non-zero tolerance is specified for Histogram or Increment series, the values retained are adjusted to preserve the average and increment of the uncompressed series over the time intervals in the compressed series.

Interpolation along a rating curve

Rating Curves are entered into Tideda as a series of pairs of values (X,Y), where X is the unrated value (i.e. the value originally recorded, e.g. water level). A minimum of three and a maximum of 50 pairs may be entered. The X-values must always increase, but the Y-values need not.

Quadratic interpolation is made on the Rating Curves by fitting a parabola through three points:

(Xi-1,Yi-1) (Xi,Yi) (Xi+1,Yi+1)

Where possible, the three points are chosen so that X is between Xi-1 and Xi, and then the parabola is chosen to be either X on Y or Y on X, depending on which has the least curvature at the point of interpolation. When (Xi,Yi) is a maximum or minimum point, a Y on X parabola with its vertex at (Xi,Yi) is used. Extrapolation is carried out, where necessary, with the choice between a Y on X or X on Y parabola, again determined by which has less curvature at the point of extrapolation.

Interpolations and extrapolations are quadratic rather than linear because a smooth rating curve can be accurately defined by fewer X-Y pairs. Typically, six pairs are sufficient for a natural river's stage to flow rating. The use of curvature to control the choice of interpolating parabola ensures that a second application of the rating, with the X-Y pairs reversed, recovers the original values exactly.

Rating changes

The Rating Curve for a particular site does not necessarily remain constant over time. For example, in the case of a water level - flow relationship, the river bed at the site could be subject to scour, and this could affect what flow corresponds to a given water level. Thus, we can have a Rating Series that applies slightly different transformations over different parts of the time span of the data.

Some rating changes occur instantaneously such as when a water level recorder is relocated, in which case the new Rating Curve is filed with the time at which it changed. Sometimes a rating change is gradual, from one stable form to another. During the transition period, the rating used lies somewhere between the new Rating Curve and its previous form. This situation is allowed for by associating two times with the new Rating Curve, thereby creating a smoothed change. The earlier time, called the "start" time, is the one at which the change starts; the later time, called the "effective" time, is the one at which the change is considered to be complete. In the overlap interval between the Rating Curves, a linear interpolation is applied. Given that *Y*old corresponds to *X* in the old Rating Curve, and *Y*new corresponds to *X* in the new Rating Curve, then the value of *Y* is given by:

Y = *Y*old + (*Y*new-*Y*old) * (time - start-time)/(effective-time - start-time)

BATCHES

A Batch is the storage space allocated when data are captured. Extra space may be allocated initially and remain available to be filled later. Thus for a Continuous Series captured frequently, such as 3 hourly telemetry of 1/4 hour data, the Batch might be made large enough to hold a month's data and take 240 sessions to fill up.

On the other hand Tideda does not allow the size of Batches allocated by some capture methods to be set by the user, and automatically allocates just enough space. Thus results from that session, including any mistakes, are isolated in a single batch to aid subsequent quality control.

A batch is a useful unit of data when managing data capture, such as checking and correcting mislabeling, i.e. assignment of the wrong Site properties. We can list the properties of the batches of data in a file in the order they were created. We can copy a batch and at the same time change its properties. We can recover data that were previously filed, and overwritten, by deleting the later batch, and we can reverse this by undeleting.

Each file can contain up to two gigabytes, and a batch can be as big as two gigabytes. Perhaps a more important limit is the maximum number of batches per file; and at present this is 32,000.

Each batch contains a Continuous Series, or a Gauging series, or a Rating Curve.

Gaps

The Gap property of a Continuous batch is either true or false, and determines whether the preceding period of time is either a Gap, or can be interpolated across. If there is not a preceding batch then interpolation is impossible and automatically Gap=true. Gaugings are never interpolated and Ratings are always interpolated so the Gap property is not available for these batches.

Text presentations show Gap periods with a symbol (eg. ?). Quantities interpolated over time periods that are partly gap, are indicated by this symbol after the number. A parameter can be set to less than 100% in some presentation processes, and then gaps are not indicated when the record contains at least this percentage of the time.

Time series graphs show gaps as blank spaces delimited by diamond symbols.

Gap and Filler values

Gap values within a data element have been adopted by a special convention for use in multi-item series where a gap is not in all items at the same time.

- 2147000001 marks a gap, i.e. a time when the value is not known.
- 2147000002 marks a filler, i.e. a time when there was no observation but interpolation between the previous and following non-filler values is valid.

*** WARNING ***

Most Tideda processes are aware of this convention and interpret the values correctly. The exceptions to this are the user defined PSIM procedures where the above item values are returned and must be handled by the PSIM procedure. For this reason it is advisable to avoid the use of this feature in data that will be filed in an archive for future use. It is best to restrict the use of gap and filler values to working files where the extra logic required to handle them in PSIM procedures is preferable to splitting the data into multiple sites.

Use of **Gap and Filler values** is never necessary in an archive because an item with them within a multi-item series can be taken out to become a single item series in which the gaps are represented in the normal way using the batch **Gap** property.

The **GRAPH AND ZOOM** process and the **SELECT** process are unaware of the convention, and are convenient tools to find these very large values.

Gap values are used when capturing multi-item series from devices which use a similar convention, and sometimes in data prepared for presentation. The programs which do this are not standard Tideda processes. This manual describes the convention so that users:

- Who have an application for this convention know about it.
- Who come across the special values in a file and need to know why they are there and what to do about them.

Entry of new data

Entry of new data for a site within the existing time range on file requires creation of a new batch and, if it is Continuous, requires consideration of the following batch's Gap properties.

- When the new data is a filed sequence in a Continuous Series between two times, file a new Batch that covers that range of time, i.e. from time-on to time-off. The times in the new batch need not match existing times in the destination Site.
- If the time-on is within an existing Gap in the destination Site then the new batch is assigned Gap=true, and otherwise Gap=false.

Likewise, if the time-off is within an existing Gap in the destination Site then the following batch is assigned Gap=true, and otherwise it is left as Gap=false.

Entry into an empty space is a simple case of this general process.

To replace a single filed value in a Continuous, Gauging or Rating Series simply enter a new value at the same time.

Release of storage space

The **Release** process releases storage space occupied by series data that have been superseded or deleted. It also makes a single batch out of consecutive batches when the later batch has Gap=false, thereby releasing space in the file directory. It should only be used when necessary because it destroys evidence about how the data were captured.

RELATED FILES

Attribute file

An Attribute file contains Attributes which describe Sites and the quantities measured there. They include fields for the Site name, the name and units of each Item recorded, the name and units of each rated Item, map references and recorder information. They also include divisors for converting values from the file units to presentation units (e.g. 1000 to convert litres/s to m3/s).

The **Site number** property of an attribute object is the key to its Site.

Attributes are optional (but highly recommended), and when supplied are used as labels in the user interface and on data presentations. Data presentations often use floating point numbers, which allow decimal digits. A divisor that turns our file units into units for presentation can be entered directly through the user interface, but is better supplied automatically as an Attribute. This use of Attributes, to label units and supply a divisor, will make it much easier to work with your data.

The file units of time series data should always be smaller than the resolution of the recording instrument, because quantities are represented by 32 bit integers (in the range 0 to ±2147,000,000 [i.e. $\pm(2^{31}$ - 483648)]), with no digits after the decimal point. When captured the data are scaled from its original units to preserve accuracy, for example by multiplying by

1000. Consistent use of a factor of 1000 where appropriate is a helpful assumption when mistakes in labeling require diagnosis.

Although we must adopt particular units of measurement, Tideda treats our data as numbers, and does not to force a system of units on us. We can use metric units, foot-pound units, or whatever else suits.

Extended directory file

An Extended Directory file is a special kind of Tideda file that accesses data spread across a number of other Tideda files. It allows us to view several such files as if they were one, permitting new data to be viewed in conjunction with existing data. In this way recently acquired data on magnetic disk can be associated with earlier data archived on write-once storage such as Cdrom, for input in a seamless way to an application. It appears to Tideda as a normal Tideda file but does not initially contain any data, and is only a directory which points to where the data may be found in other Tideda files.

If we write data to an Extended directory file, it will store that data, and it can be immediately used as an input from that Extended directory.

However if we write data to one of the files joined into an Extended directory, that directory will not "see" the new data, and we will have to create a new Extended directory that can see the new data.

All Tideda files have a directory and every batch written into the file corresponds to an entry in that directory. When part of an earlier batch is replaced by new data, up to three new entries may be written into the directory, one for the new batch, and the others to describe the parts of the earlier batches that have been retained. An important feature of Tideda is that when a new batch supersedes data already in the file, the original data are not removed from the file. This means that we can later retrieve the original data if required.

There is a directory entry called a **deleting batch** written by the Delete process and if we subsequently delete a deleting batch, the originally deleted data are retrieved. This feature is only available until the next time process Release is used on the file.

A general rule with the Tideda directories is that later entered batches supersede earlier entered batches with the same keys. This applies to each series component at each Site ie. Continuous Series, Gaugings and Ratings respectively. When creating an Extended directory the order in which files are joined is taken into account, and batches in each file as it is joined supersede batches in any file previously joined with the same keys.

List file

When data are entered manually using a keyboard, they are first written as text to the List file where they can be edited using a text editor. This allows us to check our entries and make any alterations before the data are sent to a Tideda file. We can choose any name for the List file.

Any data in a Tideda file can be selected, written to the List file, edited there and then sent back to the Tideda file.

Batch file

The name of the Batch file is always **batch.dat**.

When values are entered into Tideda directly from a digitiser or a digital recorder, they are first written to the Batch file. Processing digital data became important during the 1970s and 1980s with the advent of inexpensive microprocessor based equipment, but more recent equipment usually includes software that converts the output to text. Unlike the List file which is a text file, the Batch file cannot be edited directly. We must copy the Batch file's contents to the List file if we wish to check or edit it, then copy the edited contents back to the Batch file for further processing.

Parameter file

The name of the Tideda parameter file is always **timouse.dat**.

Tideda saves session parameters (option settings, times, files names) in this file, and then starts its next session with these settings. If we delete this file the next session starts with default settings.

Comment file

Text comments, keyed by Site then time, can be stored in a Comment file. They can be recalled and noted on graph and table presentations either by a mark to simply show that a comment exists, or by the full text.

Chart definition file

The name of the chart definition file is always **charts.def**.

Details about the mechanism of a chart recorder determine how we interpret the traces it draws on paper. We enter the dimensions and other details about the mechanism using the **ENTRY** menu's **CHART DEFINITIONS** dialog (pages 8.15 and 8.20). The information about each type of recorder is given a name and stored for reuse in the chart.definition file. This information is then used by the **ENTRY** menu's two **TRANSLATE** processes (pages 8.18 and 8.21) and the **GRAPH** menu's **CIRCULAR CHART** process (page 5.27).

Script files

The file type extension of Script file names is usually .TSF. These are text files containing commands which execute a user defined set of Tideda processes (see Chapter 11 – TSF Language).

They are written by process **RECORD SCRIPT** or a text editor and read by process **RUN SCRIPT**.

Sim files

The subscript of Sim file names is conventionally .SIM. They contain the source code for user defined procedures that are executed by process **PSIM**.

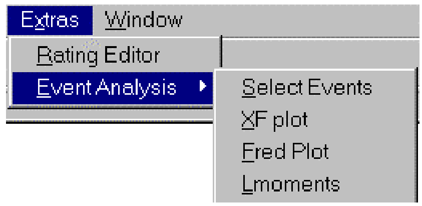
They are written by a text editor and read by process **PSIM**.

CHAPTER 15 EVENT ANALYSIS

This chapter describes four processes which analyse the frequency distribution of maxima or minima in a time series.

These extrema represent "events" and hence the label event analysis. This guide is written for hydrologists so the events in mind are floods and droughts and the applications are such things as the risk a floodway will be overtopped, a reservoir inadequate or that instream habitat or the quality of natural water will be damaged. The time series may be a record of flow in a river and the events droughts, or rain and the events rain storms, or whatever.

The **EXTRAS** menu contains an **EVENT ANALYSIS** submenu with the 4 processes.

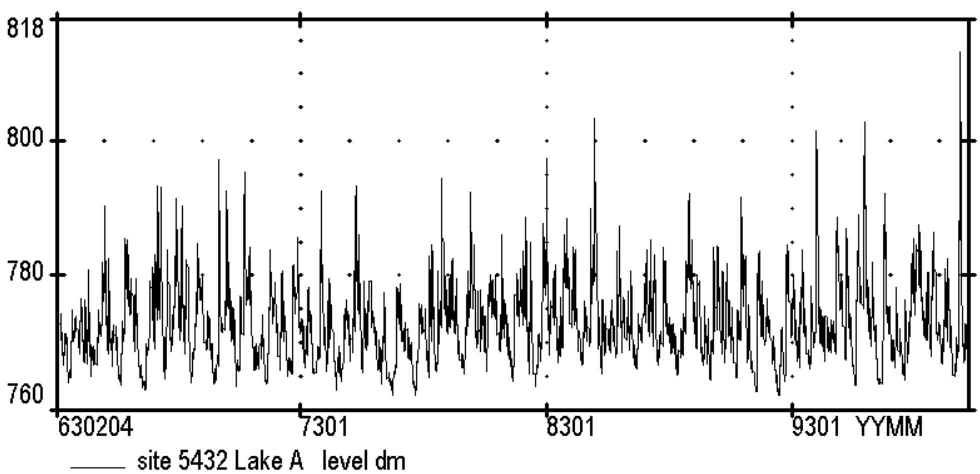


Annual exceedance probability

A frequency distribution is a relationship between the risk and the size of extrema. When the extrema are maxima then Tideda quantifies the risk as the annual probability that the size will be greater, and this quantity is often called the annual exceedance probability, shortened to Aep. On the other hand when the extrema are minima, the risk is the annual probability that the size will be smaller. The convenient notation Aep will be used to describe this kind of risk as well because we extend usage of the word exceedance to include exceedingly small. Similarly we use the word extrema for a set of maxima **<u>or</u>** minima. The return period is directly related to the Aep and both are functions of size that graph as monotonic curves.

$$\text{Return\_period(size)} = 1/\text{Aep(size)} \qquad (1)$$

The following graph shows the time series used to illustrate this chapter.



Other statistics of hydrological events

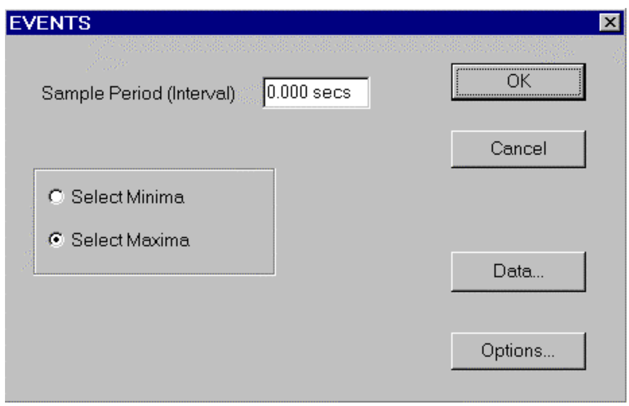
While the Aep is an important statistic of events it is not the only one. For example other statistics include:

1. cross-correlations with other records at the time of the events e.g. if extreme drought flows correlate with extremes in the predictable Southern Oscillation of barometric pressure differences across the Pacific Ocean, then droughts can be predicted;
2. serial correlation between successive events e.g. between floods large enough to damage instream biota a second time before they have recovered from the first flood;
3. use of a sequence of known risk e.g. a drought flow sequence in a simulation of the immediate future of an electricity supply system (that has both hydro storage and thermal components) to determine the thermal generation now that will ensure a specified reliability of power supply;
4. seasonal differences e.g. the best season for work in a river bed is when floods are least likely.

SELECT EVENTS

The first step in Tideda's Aep analysis is to select a set of extrema by sampling a continuous time series record. To complete an analysis many sampling parameters must be arbitrarily chosen. Fortunately with a Tideda archive we can repeatedly sample in various ways changing the sampling parameters each time until the optimum sample is obtained. This repetition is possible when we have copious recorded evidence of past events; and do not face the more common requirement with environmental statistics of minimising the cost of gathering just sufficient data.

The first step is simple.
Click **EXTRAS**,
click **EVENT ANALYSIS**,
click **SELECT EVENTS** to get this dialog >



Set **Sample Period** = 0 for instantaneous extremes. It could be a longer time, even several years, in which case the extremes are averages over that period.

Select **Minima** or **Maxima**.

Click **DATA** to get the Data dialog and specify the series for analysis. To reproduce the illustrations below specify Site 5432 from 4 Feb'63 to 14 Mar'00.

Click **OPTIONS** to get the Table Options dialog and specify the print appearance. A small Fixedsys or Courier font is recommended.

Click **OK** and a file called Events.dat is written in the **Working Directory**.specified in the **FILE** menu's **PREFERENCES** dialog.

Events.dat file

The Events.dat file is text, beginning with 4 header lines followed by a data line for every calendar month in the series, as follows:.

```
Site     5432 Dates   630203 240000 to  1000314 240000
MAXIMUM level   dm      Lake A
Interval            0 item  1 rating 0 format 8.2 divisor     100.0
        level       date   time   p.begins
       277599      630227 240000   630203
       277602      630303 240000   630301
       277327      630401 240000   630401
       277144      630530 240000   630501
       277148      630621 240000   630601
       276863      630701 240000   630701
       276977      630830 240000   630801
       277495      630917 240000   630901
        . . . . . . . . . . . . . . . . .
```

The 1st line and 1st three items on the 3rd line are from the **DATA** dialog which specified the series, and the remaining contents in the heading lines are attributes of that series from the Attribute file, except the lablel **p.begins**.

P.begins indentifies the time Partition in which each selected extreme value begins. In the Events.dat file there is a value for each calendar month; so that its time Partition is calendar months. The value may be an average that extends over several months, and by convention it is identified by the date and time when it begins..

XFPLOT

This process selects a sample of data from the Events.dat file, ranks them in order (Rank =-1 for the most extreme), assigns Aeps to each value by equations (2) and (3), and graphs the values versus Aep.

$$\text{frequency} = (\text{Rank} - \text{Alpha}) \,/\, (N + 1 - 2\ \text{Alpha}) \qquad (2)$$

$$\text{Aep} = 1 - (1 - \text{frequency})12 \,/\, \text{Partition} \qquad (3)$$

Equation (2) is called the plotting position formula, and it expresses the frequency in Partition time units, and equation (3) converts this frequency to annual time units. Recall that Aep has the argument "size" in equation (1), and this is represented by the argument Rank in equation (2); each observation in the sample has both a Rank and a size.

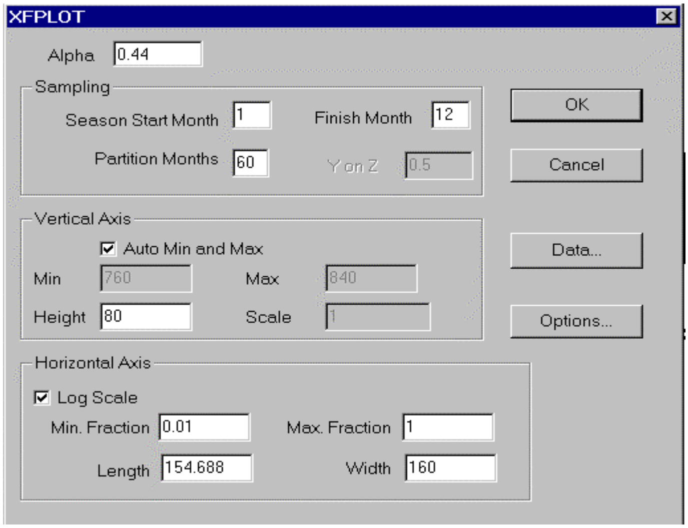
N is the number of extrema in the sample.

Alpha is called the plotting position parameter and for unbiased plotting positions different values are required depending on the distribution of the data. For example when the distribution is Gumbel or Normal, set Alpha = 0.44 or 0.375 respectively.

Click **XFPLOT** to get this dialog >

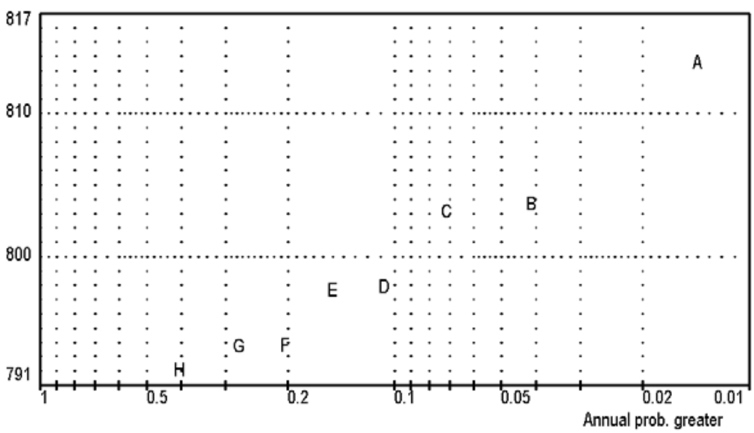
Set the parameters as shown. **Alpha** and **Partition_Months** are defined in equations (2) and (3).

Description of the other **Sampling** parameters **Season_Start_Month**



Season_Finish_Month and **YonZ** is in a later section called "Seasonal Analysis".

Click **OK** and the following graph appears.



A-H site 5432 Lake A 6302 thru 10003 level dm m=60

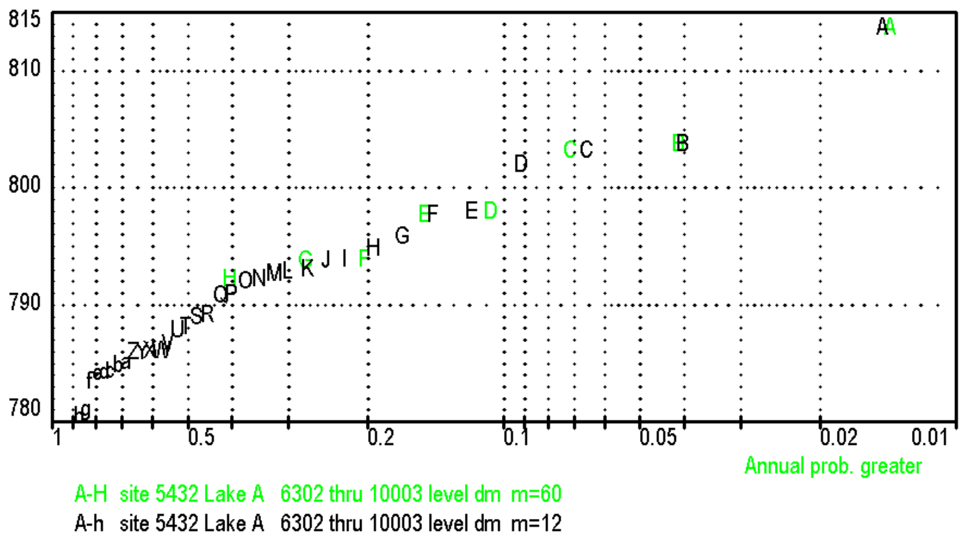
Click **VIEW**, click **TEXT** to see a table of the plotted data.

```
Reading data from C:\ALAKE\EVENTS.DAT
From  630203 240000 to 1000314 240000
 60 mth     Recorded  maximum    Alpha=0.440
 Partition    Value   measured   ann.   ret.
 starts        at     level      Prob.  per
 yymm  yymmdd:hhmmss    dm        1/y    y
 9802  991117:240000   813.30  A 0.014   70
 8302  841222:240000   803.38  B 0.042   24
 9302  951214:240000   802.82  C 0.073   14
 7802  830113:240000   797.58  D 0.109    9
```

6802	690913:240000	797.31	E 0.152	7
6302	670311:240000	793.47	F 0.206	5
7302	750402:240000	793.37	G 0.281	4
8802	901215:240000	791.76	H 0.414	2
	Mean =	799.12		

The 37 year series is prepesented by just 8 maxima, the first 7 from 60 month Partitions, and the last from the remaining 25 months.

We can overplot another set of extrema from a 12 month Partition as follows.
Click **EXTRAS**, click **EVENT ANALYSIS**, click **XFPLOT**, deselect **Auto_Min_Max**, set **Vertical_Axis_Min**=783 & **Vertical_Axis_Max** =815 so there will be more space at the bottom of the graph, click **OK** to repeat the previous graph.
Click **GRAPH**, click **PLOT TYPE**, click **OVERPLOT**
Click **EXTRAS**, click **EVENT ANALYSIS**, click **XFPLOT**, set **Partition** = 12, click **OK** to get the



following graph and table.

12 mth Partition starts yymm	Recorded Value at yymmdd:hhmmss	maximum measured level dm	Alpha=0.440 ann. Prob. 1/y	ret. per y
9902	991117:240000	813.30	A 0.015	68
8402	841222:240000	803.38	B 0.041	24
9502	951214:240000	802.82	C 0.067	15
9302	940110:240000	801.59	D 0.093	11
8202	830113:240000	797.58	E 0.120	8
6902	690913:240000	797.31	F 0.146	7
7002	700919:240000	795.47	G 0.172	6
7802	781015:240000	794.42	H 0.198	5
6702	670311:240000	793.47	I 0.225	4
7502	750402:240000	793.37	J 0.251	4

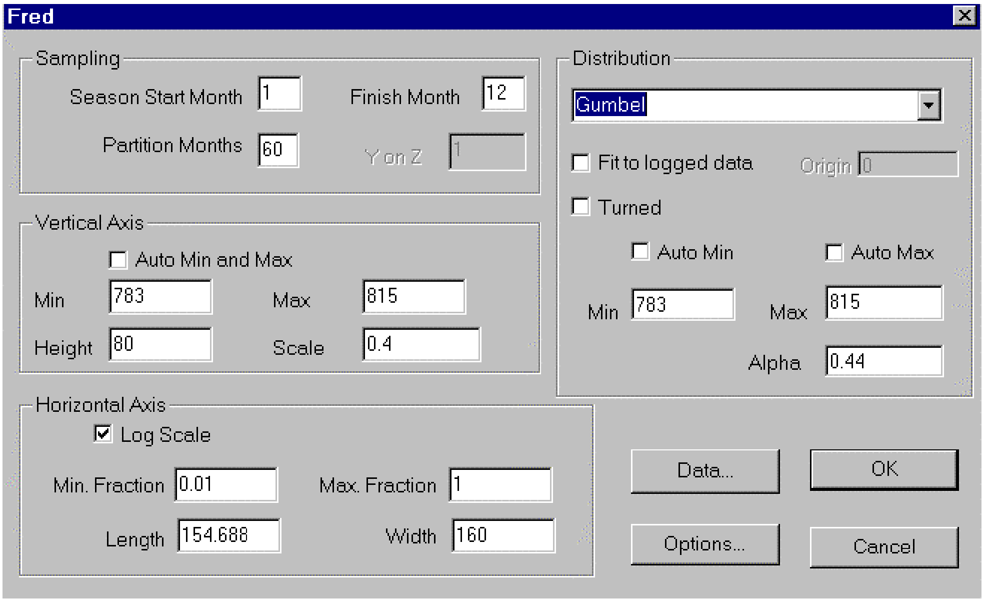
. . . .

Using Partition = 60 had censored all but 1 of the 27 smallest annual maxima but retained 7 of the 10 largest Thus both samples almost equally well represent the larger maxima.

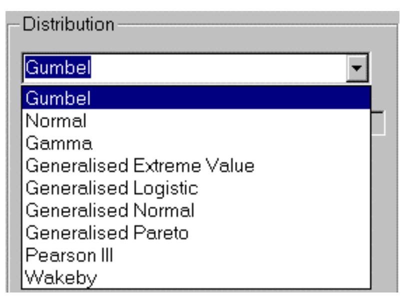
FRED

FRED stands for FREquency Distribution. This process selects a sample of data from the Events.dat file, computes L-moments of the sample and then the parameters in a specified distribution formula so that it has the same lower order moments, then graphs and tabulates that distribution.

Click **FRED** to get this dialog.



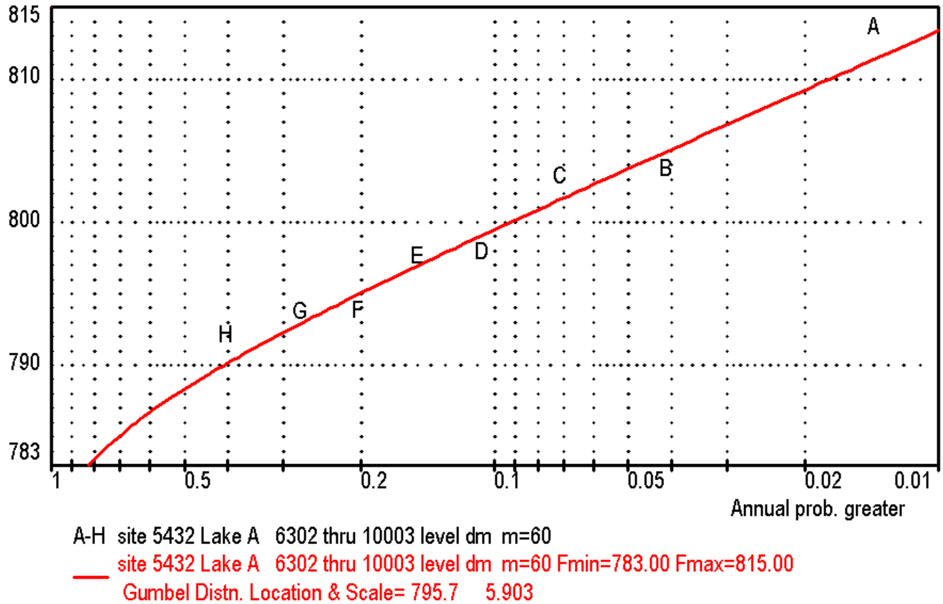
The **Sampling, Vertical Axis** and **Horizontal Axis** parameters are the same as for the **XFPLOT** process. To prepare the following illustration we must first run **XFPLOT** and choose **PLOT TYPE OVERPLOT**.



Set **Distribution = Gumbel**. **FRED** offers 9 distribution formulas to choose from >
To fit **Gumbel** and **Normal** requires calculation of only 2 parameters, and the others require more, thus these two should be preferred. **Gumbel** has +ve skew, **Normal** has zero skew and **Gumbel Turned** has –ve skew.

Set the **Distribution Min** & **Max** the same as the **Vertical Axis Min** & **Max**. L-moments of the sample are calculated using only the values that are in this range which can differ from the range of the graph. Thus dodgy data at either the top or the bottom can be excluded when calculating moments but remain on the graph.

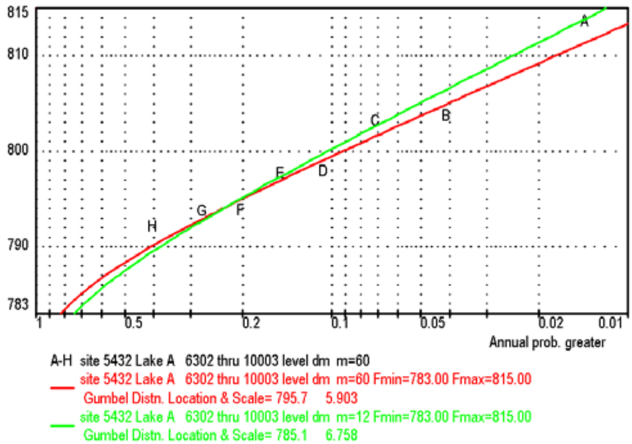
Click **OK** to get the following graph and table.




```
Reading data from C:\ALAKE\EVENTS.DAT
Site      5432 Lake A
From   630203 240000 to 1000314 240000
Moments L1=  799.12     L2=    4.09     T3= 0.300       T4= 0.150
Location  =  795.72 Scale =    5.90   100yr/2.33yr = 1.030
60 mth    Recorded  maximum        -- Gumbel Distribution --
Partition   value   measured                1.96   ann. ret.
starts        at    level          level    std.   prob. per
yymm   yymmdd:hhmmss  dm             dm      dev.   1/y    y
                                  826.99   20.18 0.001 1000
                                  813.37   11.62 0.010  100
9802   991117:240000   813.30 A   813.30   11.58 0.010   99
                                  809.25    9.17 0.020   50
                                  803.75    6.22 0.050   20
8302   841222:240000   803.38 B   803.38    6.05 0.053   19
9302   951214:240000   802.82 C   802.82    5.80 0.058   17
                                  799.50    4.63 0.100   10
7802   830113:240000   797.58 D   797.58    4.33 0.136    7
6802   690913:240000   797.31 E   797.31    4.31 0.142    7
                                  795.07    4.47 0.200    5
6302   670311:240000   793.47 F   793.47    4.85 0.254    4
7302   750402:240000   793.37 G   793.37    4.88 0.257    4
8802   901215:240000   791.76 H   791.76    5.46 0.324    3
              Mean =   799.12
```

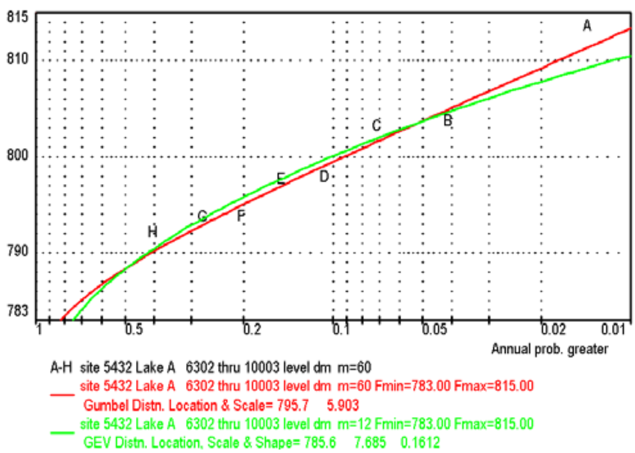
Note that the return period of the maximum on page 16-4 (70 yr) is much less than the return period of the same event in this table (99 yr). The former estimate is based only on the fact that this was the largest observation in 37 years, while the latter estimate is based on the size of the largest observations in 8 different periods of 5 years. The latter figure is more reliable because it is based on more data.

However using more data does not necessarily give us a more reliable answer. For example if we: Click **EXTRAS**, click **EVENT ANALYSIS**, click **FRED**, set **Partition** = 12, click **OK** we get:



The revised distribution curve now passes through the maximum (A) and assigns it a return period of 66 yr. This change is brought about by including all the 22 values that plot below 793.37 (G) and are above 783 when fitting the distribution. (On p 16-5 is a graph and on p 16-15 a table of these values.) It can be argued that we should not change the distribution curve by this much above G soley to fit observations less than G.

Fitting a 3 parameter GEV distribution is not a good solution to this problem, as shown in the following graph, because the GEV seems to deviate too far the other way at the top.

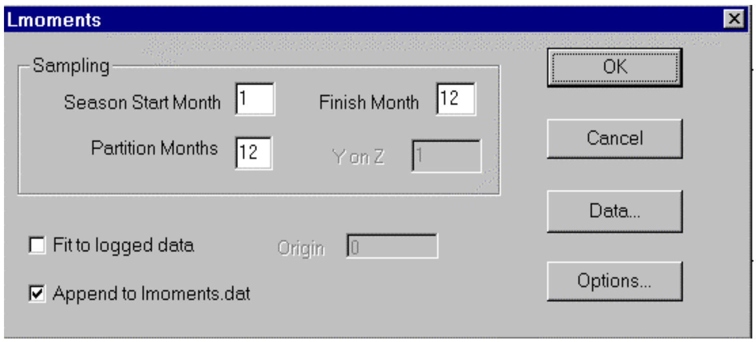


Unfortunately there is no generally accepted rule for assigning a return period to the most extreme observation, or for the more important task of extrapolating the frequency distribution to estimate the size of events with even longer return periods. Thus process **FRED** offers a choice of alternative methods so that we can compare alternative distributions, just as process **XFPLOT** offers a choice of alternative sampling methods.

LMOM

This process selects a sample of data from the Events.dat file, computes the first 4 L-moments and prints them. They are used to assist the choices identified in the previous paragraph.

Click **EXTRAS**, click **EVENT ANALYSIS**, click **LMOM** to get this dialog >



The **Sampling** specification is the same as for processes **XFPLOT** and **FRED**.

Click **OK** to get the following in the Text window

```
Reading data from C:\ALAKE\EVENTS.DAT
From  630203 240000 to 1000314 240000
 12  1 12 1.0    *   0 10003 9911  788.95    4.68 0.070 0.202  5432 1 0
max level   dm      Lake A
```

Select **Append to Lmoments.dat** and the results are appended to a text file called Lmoments.dat in the **Working Directory** specified in the **FILE** menu's **PREFERENCES** dialog. For example after we have run process **LMOM** twice this file contains:

```
part seas YonZ orig interv first last  L1     L2    T3     T4    site item rate
12    1 12 1.0    *      0    10003 9911 788.95 4.68 0.070 0.202 5432  1    0
60    1 12 1.0    *      0     9012 9911 799.12 4.09 0.300 0.150 5432  1    0
```

"First" and "last" are the partitions that contain the smallest and largest values in the set. Note that the smallest value in the Partition=12 sample is in the most recent partition "10003" for the spurious reason that this partition was actually only 14 days while the other partitions are 12 months long. However when fitting a distribution we censored the values less than 783 so this error did not affect our results.

Theory in the following paper defines L-moments and explains why they are appropriate for our purpose:

> Hosking J R M (1990) "L-moments: Analysis and Estinmation of Distributions using Linear Combinations of Order Statistics", J. Royal Statistical Soc B, v52(1) pp.105-124.

Source code supplied with the following report was used to calculate L-moments and to use them to fit the distributions available in process **FRED**:

> Hosking J R M (1990) "Fortran routines for use with the method of L-moments, ver. 2", IBM mathematics research report, T J Watson Resaearch Center, NY 10598, 117 pp.

The following book explains how to combine observations from several locations to make the choice about which distribution to use when extrapolating a frequency distribution to estimate the size of events with even longer return periods than the most extreme observation.

> Hosking J R M, Wallis J R (1997) "Regional Frequency Analysis: an Approach Based on L-Moments", Cambridge University Press, 224pp.

The following report explains in an elementary way how different sampling options affect the calculated L-moments. The writer intends to revise and publish the generally relevant part of this report.

> Thompson S M Henderson R D and Pearson C P (1992) "Frequency of Drought Flows affecting ECNZ", NIWA report to ECNZ, 42pp.

Seasonal Analysis

We now examine the result of selecting a sample from a season of the year with a duration that is a fraction of a year (Y/12) where

$$Y = 1 + (12+\text{Season\_Finish\_Month} - \text{Season\_Start\_Month}) \text{ modulo } 12 \qquad (4)$$

Extrema that have occurred in the remaining fraction (1-Y/12) of each year are not considered available for selection.

We begin this discussion using plotting positions because then we have algebraic formulae in which the Rank, sample size N and Partition are all explicit, and there is no fitting process. When we choose Y=12 we get the same sample as we have discussed before and the Aep is calculated using equations (2) and (3) which we combine in equation (5).

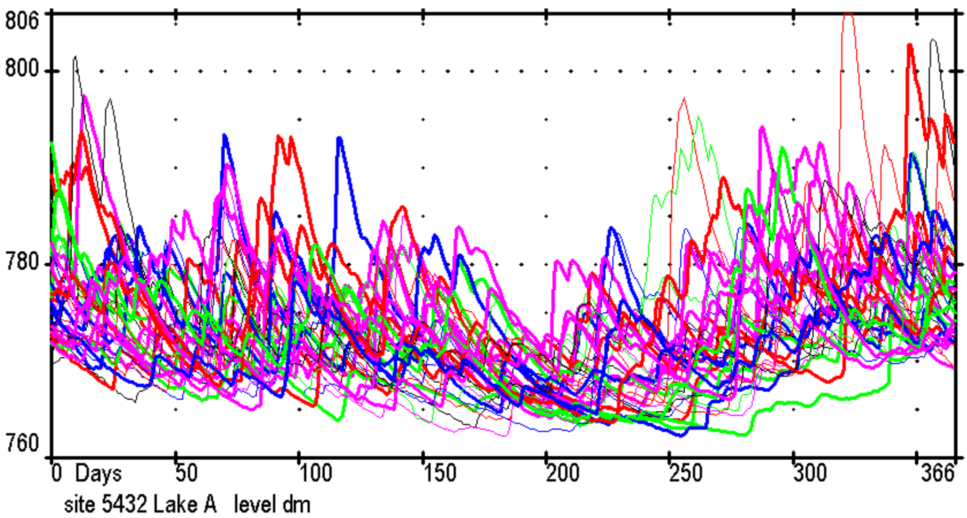
$$\text{Aep} = 1 - [1 - (\text{Rank} - \text{Alpha})/(N + 1 - 2\ \text{Alpha})]^{12/\text{Partition}} \qquad (5)$$

Partition/12 is the number of years from which each minima is selected. However when Y < 12 each Partition includes less time (i.e. (Y/12) Partition/12 years), and so the Aep is greater and return period less:

$$\text{Aep} = 1 - [1 - (\text{Rank} - \text{Alpha})/(N + 1 - 2\ \text{Alpha})]^{12/(\text{Partition}\ (Y/12))} \qquad (6)$$

When the distribution of extrema is the same throughout the year this adjustment produces the correct Aep for any value of Y given a large Partition and large N. However there is no reason to take a seasonal sample when the distribution is the same throughout the year, so equation (6) is not useful and is not implemented in this form in process **FRED**.

When the distribution of extrema differs in different seasons of the year then a more complex analysis is required to estimate the Aep. For example consider the following graph in which 37 calendar years beginning on 1st January have been overplotted.



When we run process **XFPLOT** with a nonseasonal sample and **Partition** = 12 and we examine the table we find the number of annual maxima in each month are:

Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
6	4	4	2					3	4	8	6

The maxima are all in the months September through April (which we will call Z), and none are in the months May through August.

For clarity of the following argument we start by assuming that all the maxima outside of Z, i.e. in the months May through April, are less than the least maximum in Z, although this is not strictly true. If we specify a Season Y that overlaps part of Z, then the effective sample duration is only the fraction of Y that overlaps (which we will call **YonZ**) To get the same distribution as we get when we sample the whole year:

$$\text{Aep} = 1 - [1 - (\text{Rank} - \text{Alpha})/(\text{N} + 1 - 2\ \text{Alpha})]^{12/(\text{Partition YonZ})} \quad (6)$$

Process **FRED** Implements equation (6). Equation (5) is just the particular case of (6) when Z=12. The argument leading to (6) and the paramter **YonZ**, may be easy to understand but they are not so easy to use.

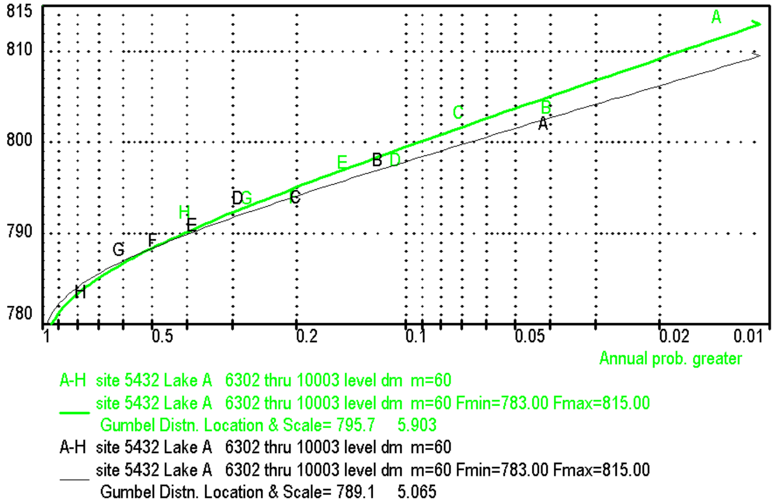
When a distribution F(size) is fitted by L-moments to a sample of extrema using the Fortran code referenced on p 16-10, then the seasonal Aep is defined by equation (7).

$$\text{Aep} = 1 - [1 - \text{F(size)}]^{12/(\text{Partition YonZ})} \quad (7)$$

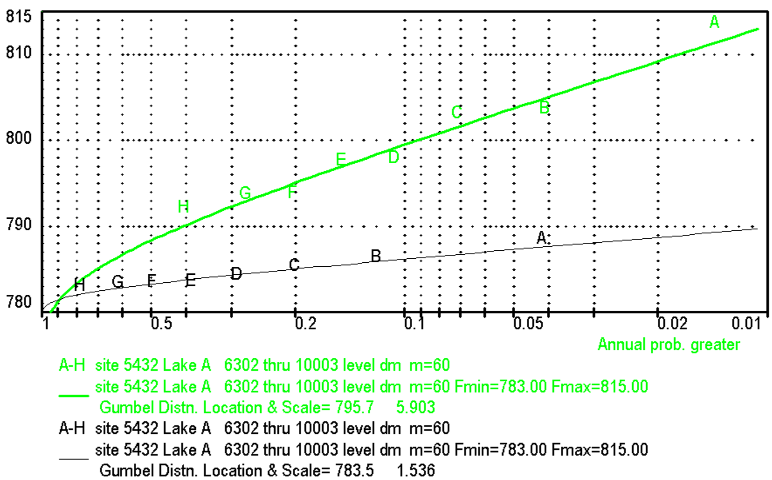
The following example illustrates how to use (6) and (7). Five comparisons are graphed, and all use the same series as the previous examples (Site 5432 from 4 Feb'63 to 14 Mar'00), and all except the last use Partition = 60 months.

1st we compare Gumbel distributions fitted to: (a) a nonseasonal sample; (b) a sample from months 1 to 4 with YonZ = 0.33. The result is reasonably consistent with the argument

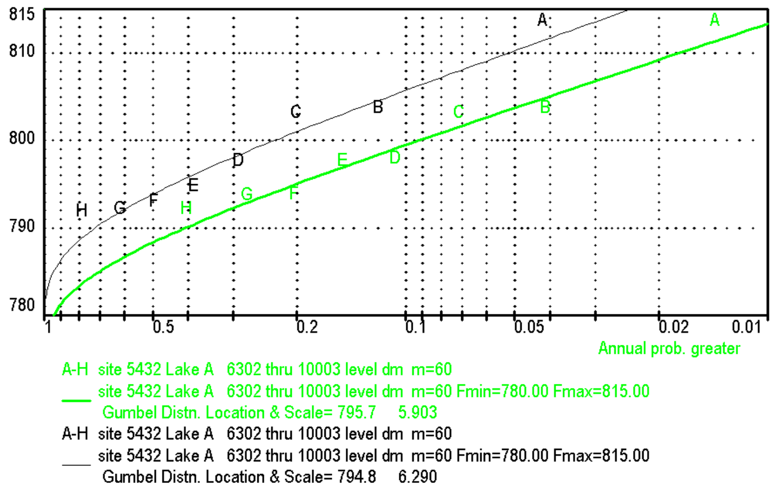
2nd we compare Gumbel distributions fitted to: (a) a nonseasonal sample; (b) a sample from



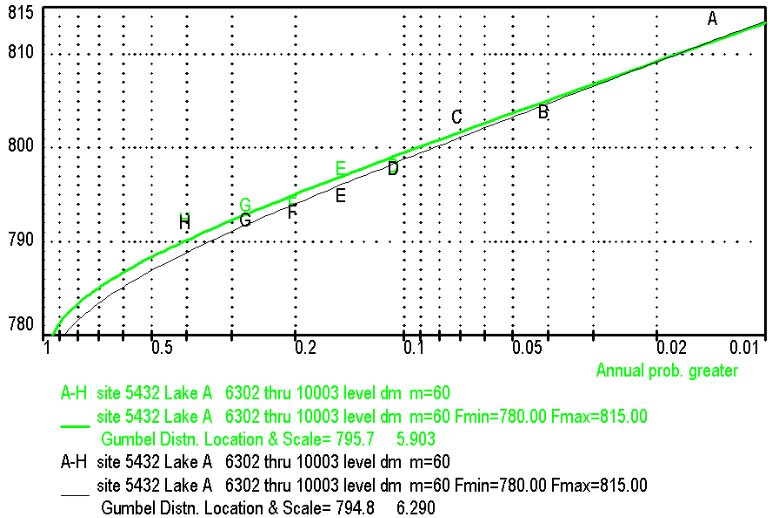
months 5 to 8 with YonZ = 0.33. The result is clear evidence for a different distribution of extrema in this season consistent with the lack of annual maxima shown on p 16-11.



3rd we compare Gumbel distributions fitted to: (a) a nonseasonal sample; (b) a sample from months 9 to 12 with YonZ = 0.33. The result exposes a problem because a seasonal curve of maxima obviously cannot plot above the nonseasonal curve representing the same series.



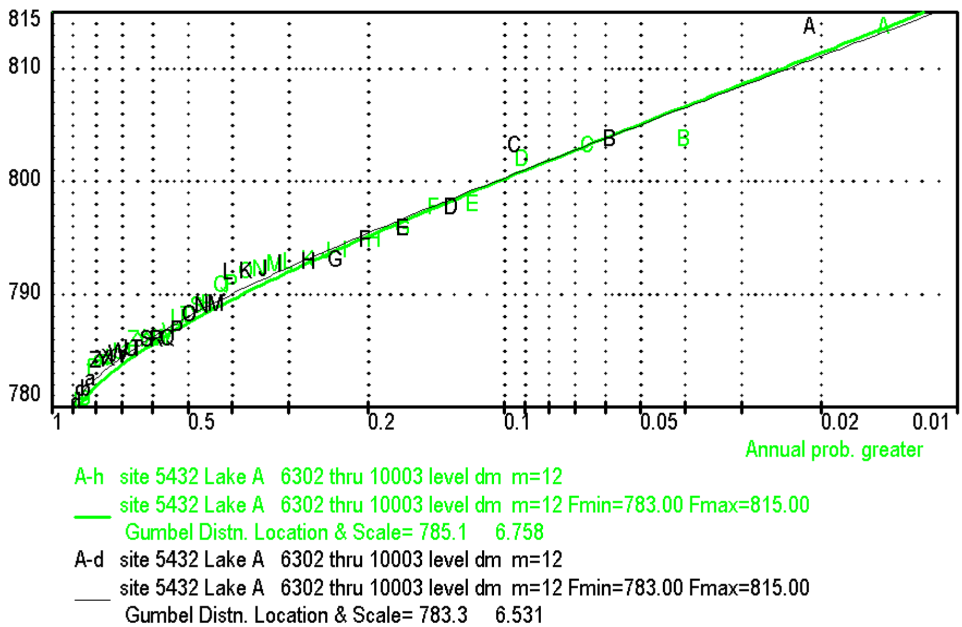
4th we make the same comparison as the 3rd case except we set YonZ = 1, and this corrects the problem.



For a seasonal sample the correct value of YonZ can be anywhere between Y/12 and 1.

1. YonZ = Y/12 will be correct when a distribution in a seasonal sample is the same (e.g. 1st case), or less extreme (e.g. 2nd case), than in a nonseasonal sample. In the 2nd case the distribution will accurately represent the risk in the season of less extreme extrema.
2. 1 >= YonZ > Y/12 will be correct (e.g. 4th case) when the time series has many values in each partition close to the exteme value. Then a more accurate estimate of the distribution can be obtained by reducing the partition interval so as to increase the sample size N.

To illustrate the last statement we reduce the Partition interval from 60 to 12 months, and in a 5th graph compare Gumbel distributions fitted to (a) a nonseasonal sample (light colour); (b) a sample from months 9 to 12 with YonZ = 0.7 (black). The excellent agreement confirms the statement.



A complete table of the annual extrema is on the next page. The column “1.96 std. dev.” in the table is the range of values either side of the tabulated value for 95% confidence. Reducing the partition interval from 60 to 12 months has reduced the confidence range as follows:

```
page 16-7,  Partiton=60, 100 year value = 813.37 +- 11.62 = 801.75 to 824.99
page 16-15, Partiton=12, 100 year value = 816.14 +-  8.72 = 807.42 to 824.86
```

There is a trade-off between two contradictory effects, both desireable
1. decrease in confidence range with decreasing Partition interval, and
2. increase in the average size of the sample maxima with increasing Partition interval.

```
Reading data from C:\ALAKE\EVENTS.DAT
```

Site 5432 Lake A
From 630203 240000 to 1000314 240000
Moments L1= 788.95 L2= 4.68 T3= 0.070 T4= 0.202
Location = 785.05 Scale = 6.76 100yr/2.33yr = 1.034
Values smaller than 783.0 not used when fitting
Values larger than 815.0 not used when fitting

12 mth partition starts yymm	Recorded value at yymmdd:hhmmss	maximum measured level dm		-- Gumbel Distribution -- level dm	1.96 std. dev.	ann. prob. 1/y	ret. per y
				831.73	13.12	0.001	1000
				816.14	8.72	0.010	100
9902	991117:240000	813.30	A	813.30	7.94	0.015	66
				811.42	7.42	0.020	50
				805.13	5.71	0.050	20
8402	841222:240000	803.38	B	803.38	5.25	0.064	16
9502	951214:240000	802.82	C	802.82	5.10	0.070	14
9302	940110:240000	801.59	D	801.59	4.79	0.083	12
				800.26	4.45	0.100	10
8202	830113:240000	797.58	E	797.58	3.80	0.145	7
6902	690913:240000	797.31	F	797.31	3.74	0.150	7
7002	700919:240000	795.47	G	795.47	3.33	0.193	5
				795.19	3.27	0.200	5
7802	781015:240000	794.42	H	794.42	3.11	0.221	5
6702	670311:240000	793.47	I	793.47	2.92	0.250	4
7502	750402:240000	793.37	J	793.37	2.90	0.253	4
7302	731107:240000	792.65	K	792.65	2.77	0.277	4
7902	791203:240000	792.49	L	792.49	2.74	0.283	4
9602	961022:240000	792.24	M	792.24	2.70	0.292	3
9002	901215:240000	791.76	N	791.76	2.63	0.310	3
8802	881030:240000	791.63	O	791.63	2.61	0.315	3
6402	650109:240000	790.50	P	790.50	2.45	0.360	3
6802	680311:240000	790.42	Q	790.42	2.44	0.364	3
				788.95	2.29	0.430	2.33
9402	941109:240000	788.70	R	788.70	2.28	0.442	2
8302	831119:240000	788.50	S	788.50	2.26	0.451	2
9802	980310:240000	787.67	T	787.67	2.22	0.493	2
8502	860103:240000	787.42	U	787.42	2.22	0.506	2
8102	810310:240000	786.19	V	786.19	2.22	0.570	2
9702	980223:240000	785.64	W	785.64	2.24	0.600	2
7202	721114:240000	785.63	X	785.63	2.24	0.601	2
6502	651118:240000	785.58	Y	785.58	2.24	0.603	2
8702	870401:240000	785.50	Z	785.50	2.25	0.608	2
9202	921115:240000	784.67	a	784.67	2.30	0.653	2
8902	891216:240000	784.45	b	784.45	2.32	0.665	2
8602	870205:240000	783.95	c	783.95	2.37	0.692	1
7102	711006:240000	783.90	d	783.90	2.37	0.695	1
9102	910914:240000	783.75	e	783.75	2.39	0.703	1
6602	670126:240000	783.06	f	783.06	2.47	0.739	1
7402	750225:240000	780.71	g	780.63	2.85	0.854	1
8002	801220:240000	780.07	h	779.96	2.97	0.880	1
7602	770203:240000	778.83	i	779.22	3.12	0.907	1
6302	640123:240000	776.63	j	778.34	3.30	0.933	1
7702	771113:240000	776.00	k	777.20	3.55	0.959	1
0002	1000301:240000	768.96	l	775.32	3.99	0.985	1
	Mean =	788.95					

CHAPTER 16 ENTRY VIA EXCEL

Scope of this use of EXCEL

This chapter is about using the Microsoft EXCEL program for reading text data in a proprietary layout and writing it as text in a layout for entry to a Tideda file using on of the **ENTRY** menu's processes **FULLY SPECIFIED**, **SERIES VALUES** or **TIME STAMPS**.

EXCEL is widely used and has some excellent features for processing time series. Tideda complements it by providing an archiving capability for data that is unequally spaced in time, requires interpolation, and is more extensive than can be conveniently stored on a set of spreadsheets. We suggest that you use both programs.

The following assumes you already have a copy of EXCEL and some experience with it. If not then you will need to make progress on those two things before reading further. You require expertise at using macros in Excel, and preferably but not necessarily at writing macros. If someone adapts the code supplied with Tideda for the proprietary text layouts produced by your equipment then you can use them without further change.

We focus narrowly on one kind of EXCEL application. We present a set of 4 examples which illustrate 4 methods. You need to recognise which method is relevant, and then adapt the example we provide of that method to process your particular proprietary text. You can run the examples using the following files supplied with Tideda:

Tlist.xls,

Time_stamps_example.txt,

Series_values_example.txt,

Fully_specified_example.txt

Tlist.xls contains the Visual Basic code of the 4 macros and one of these will be appropriate as a starting point for your task. It is unlikely that your application will be straight forward.

For example, when the text from the recorder contains unexpected characters, the macro will stop processing and go into its debug mode. A Code window appears with the code highlighted that was in use when it stopped processing. An Immediate window also appears where we type: ?row <enter>   and EXCEL responds with the row number of the offending data. We then open the spreadsheet containing the text, go to that row and edit the text to fix the problem. Then we rerun the macro.

You need to know how to use the debug options in this way to get past the problems you will encounter. This requires some learning and probably some instruction. We suggest you do not start out with an attempt to know all about Visual Basic macros, it is a very big subject. The following notes describe the small part of this subject that you need to know.

EXCEL's date format

EXCEL assumes that the format of dates will correspond to the operating system's regional setting. To check and if necessary reset this:

Click Start (button at bottom left on a Windows 95 and a Windows NT screen)

click Settings

click Control Panel

click Regional Settings icon

click Regional Settings tab

select: English (New Zealand)

(and only in NT) select the button: Set as system default locale.

click Apply

If you change this setting you have to restart the computer to adopt it. Then the date 2/4/98, for instance, will be interpreted as 2 April 1998. If your computer has the American setting it will interpret this date as February 4, 1998.

OPEN TLIST.XLS

After launching either EXCEL 7.0 or the EXCEL 97, open the Workbook Tlist.xls supplied with Tideda:

Click File, click Open,

double click the folder where Tlist is located,

double click Tlist.xls

We can use the macros in Tlist.xls in either of these versions of EXCEL.

Importing text into EXCEL

Open a proprietary text file and respond to the questions about delimiters. For example, we suggest you begin by opening one of the three .txt files listed above.

The delimiters in these files are: comma, tab and | respectively.

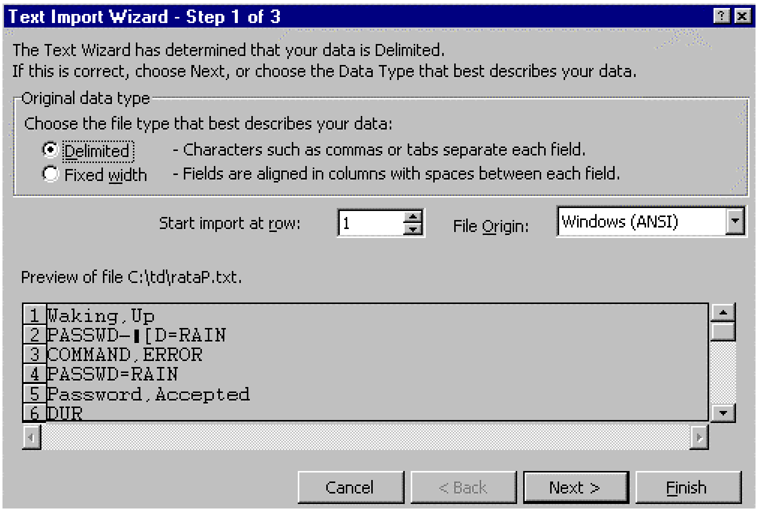
The following illustration uses Time_stamps_example.txt.

Click File, click Open,

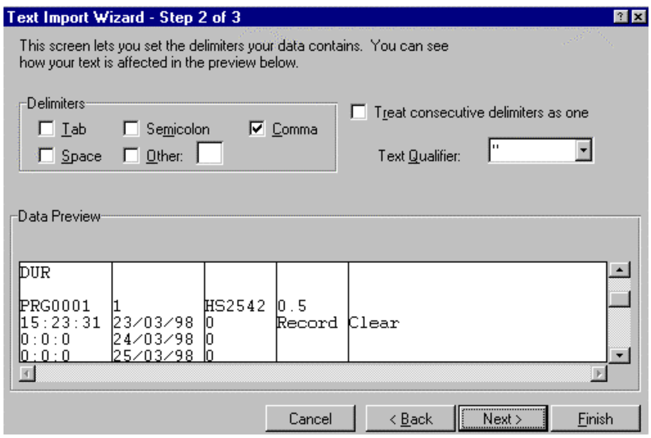
double click the folder containing the text file,

select Files of type: Text files,

double click Time_stamps_example.txt



Click Next



Select Delimiters Comma, Click Finish.

The data will be in “Workbooks(2).Worksheets(1)” since this is the second Workbook you have opened, and it has only one Worksheet.

RUN A MACRO

To run the relevant macro in Tlist.xls (which you have already opened):

click Tools, click Macro,

(and only in EXCEL 97) click Macros,

select Time_stamps.

click Run

Then close the text file in Workbooks(2) **without saving** so that the original text file is not changed, which ensures you can easily process it again:

click File, click Close, click **No.**

You can examine the result of this process by opening the List file with a text editor, such as Notepad.

TIME STAMPS example

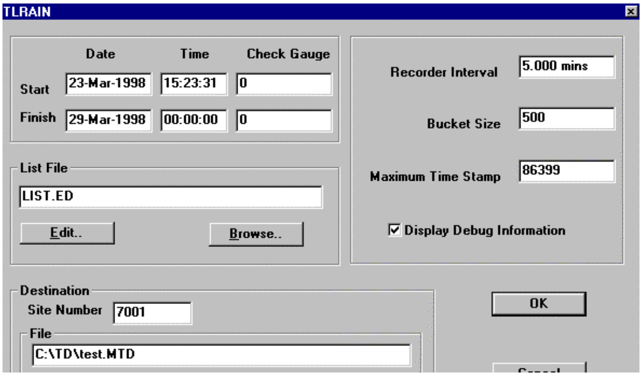
With the following Time_stamps_example.txt as input:

```
Waking,Up
PASSWD-_[D=RAIN
COMMAND,ERROR
PASSWD=RAIN
Password,Accepted
DUR
PRG0001,1,HS2542,0.5
15:23:31,23/03/98,0,Clear
0:0:0,24/03/98,0
0:0:0,25/03/98,0
23:13:23
23:38:55
0:0:0,26/03/98,2
0:5:6
0:0:0,27/03/98,1
0:0:0,28/03/98,0
Record,End,
CLR
Please,Wait...
OK
BYE
```

The Time_stamps macro will rearrange this data thus:

```
TIMES 19980323 152331 19980328 0
55411  0     0     83603  85135  0     306    0     0      1
```

Click **ENTRY**, click **LIST TO TIDEDA**, click **TIME STAMPS** to get this dialog >



Click **OK** to enter the data into the **Destination File**.

SERIES VALUES example

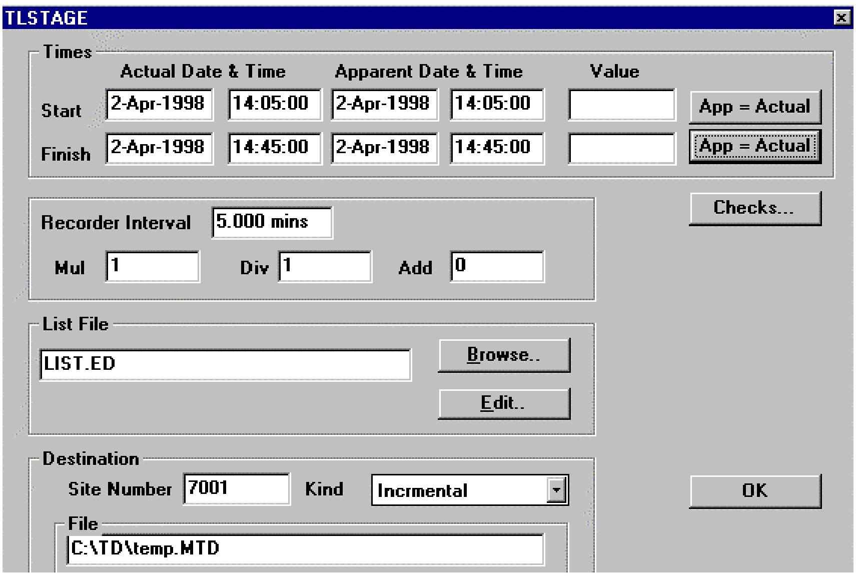
Use the following Series_values_example.txt as input:

```
DATE    TIME FLO   LEV   VEL
dd/mm/yy     hh:mm LPS   cm.   MPS
2/04/98      14:05 6.656 5.359 0.4542
2/04/98      14:10 0     2.616 0
2/04/98      14:15 10.64 6.147 0.5822
2/04/98      14:20 4.867 4.29  0.4724
2/04/98      14:25 0     2.718 0
2/04/98      14:30 2.358 3.023 0.3962
2/04/98      14:35 8.095 5.436 0.5395
2/04/98      14:40 7.817 5.359 0.5334
2/04/98      14:45 3.33  3.632 0.4176
```

The Series Values macro will rearrange the data thus::

```
TIMES 19980402 140500 19980402 144500
6656      0 10640   4867      0   2358   8095   7817   3330
```

Click **ENTRY**, click **LIST TO TIDEDA**, click **SERIES VALUES** to get this dialog >



Click **OK** to enter the data into the **Destination File**.

FULLY SPECIFIED example

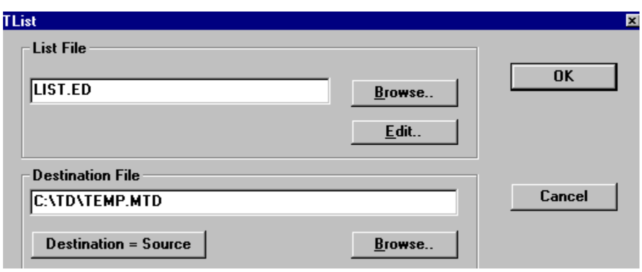
Use the Fully_specified_example.txt as input:

```
Reynolds Bach••
Flow Logger:  266
Location: 4
   Pipe shape:  Circular
   Cross Sectional Area:    0.552 Sq Metres
 Major (Vertical) Dimension:     0.838 Metres
   Minor Dimension:  0.838 Metres
    Printing from:   1/7/97 0:0:0.
     |              |         |        |   MEAN     |   MEAN    |
    DATE   TIME     |  DEPTH  |VELOCITY|   FLOW     |  HOURLY   | HOURLY
                    |         |        |            |   DEPTH   |FLOW
   ---------------+---------+--------+----------+---------+------
                    |   (m)   |  (m/s) |  (l/s)     |   (m)     |(l/s) •-
Rate:   6 MIN
Date: 01/07/97
      1/ 7   0. 0 |  0.144  |  0.71  |    44.1    |           |
             0. 6 |  0.144  |  0.68  |    42.5    |           |
             0.12 |  0.147  |  0.70  |    45.2    |           |
             0.18 |  0.147  |  0.70  |    45.2    |           |
             0.24 |  0.140  |  0.69  |    41.4    |           |
             0.30 |  0.147  |  0.70  |    45.2    |           |
             0.36 |  0.144  |  0.74  |    46.3    |           |
             0.42 |  0.145  |  0.69  |    43.6    |           |
             0.48 |  0.147  |  0.66  |    42.5    |           |
             0.54 |  0.138  |  0.67  |    39.2    |   0.144   |  43.5
      1/ 7   1. 0 |  0.136  |  0.65  |    37.0    |           |
             1. 6 |  0.136  |  0.68  |    39.2    |           |
             1.12 |  0.135  |  0.65  |    37.0    |           |
```

The Fully_specified macro will rearrange this data thus:

```
333 3 instant
 144           710          44100          970701      0
 144           680          42500          970701      600
 147           700          45200          970701      1200
 147           700          45200          970701      1800
 140           690          41400          970701      2400
 147           700          45200          970701      3000
 144           740          46300          970701      3600
 145           690          43600          970701      4200
 147           660          42500          970701      4800
 138           670          39200          970701      5400
 136           650          37000          970701      10000
 136           680          39200          970701      10600
 135           650          37000          970701      11200
```

Click **ENTRY**, click **LIST TO TIDEDA**, click **FULLY SPECIFIED** to get this dialog >

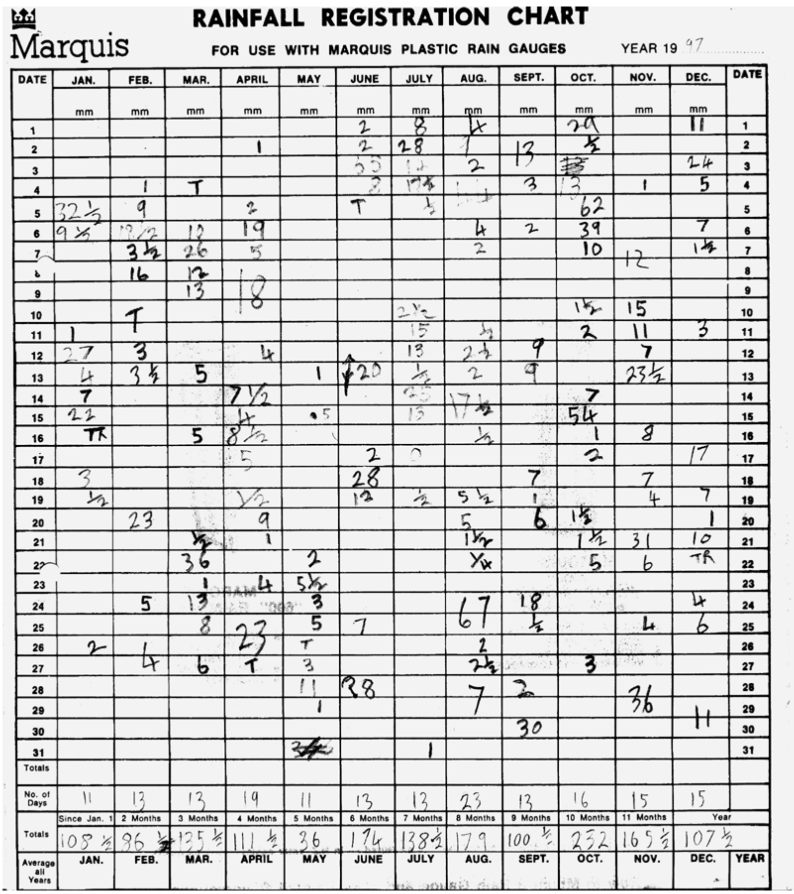


Click **OK** to enter the data into the **Destination File**.

DAILY example

When handwritten data must be typed into a computer it is usually best for checking if the handwritten layout is repeated in type. Thus the handwritten daily rain record below was typed into Sheet1 of Tlist.xls, which is also shown below. Where the gauge was not read until after two or more days of rain, the mm are distributed exactly evenly over those days. This total will appear as a single number in the Tideda file. Each trace of rain has been entered here as 0.1 mm.

Other series of daily numbers could be typed into Sheet1 of Tlist.xls, after deleting what may already be in that spreadsheet, so that in a sense this method is a general purpose tool. Formulas to Count the rain days and Sum the columns can be put in the spreadsheet to check the entered data against the manual summary at the bottom of the handwritten sheet, but are not a necessary part of the entry processing.



	1	2	3	4	5	6	7	8	9	10	11	12
1						2	8	4		29		11
2				1		2	28	9	6.5	0.5		
3						65	14	2	6.5			24
4		1	0.1			8	17.5	22	3	13	1	5
5	32.5	9		2		0.1	0.5	22		62		
6	9.5	18.5	10	19				4	2	39		7
7		3.5	26	5				2		10	6	1.5
8		16	12	6							6	
9			13	6								
10		0.1		6			2.5			1.5	15	
11	1	0.1					15	0.5		2	11	3
12	27	3		4		6.7	13	2.5	9		7	
13	4	3.5	5		1	6.7	0.5	2	9		23.5	
14	7			7.5		6.7	25	8.8		7		
15	22			4	0.5		13	8.8		54		
16	0.1		5	8.5				0.5		1	8	
17				5		2				2		17
18	3					28			7		7	
19	0.5			0.5		12	0.5	5.5	1		4	7
20		23		9				5	6	1.5		1
21			0.5	1				1.5		1.5	31	10
22			36		2			0.3		5	6	0.1
23			1	4	5.5							
24		5	13		3			33.5	18			4
25			8	11.5	5	7		33.5	0.5		4	6
26	2	2		11.5	0.1			2				
27		2	6	0.1	3			2.5		3		
28					11	28		3.5	2		18	
29					1			3.5			18	5.5
30									30			5.5
31					4		1					
32	1997	1	1	12	31							

To use the Daily macro to pre-process this data, you **must** also type into row 32 as shown above:

col.1) the year,

col.2) start month,

col.3) start day,

col.4) finish month

col.5) finish day.

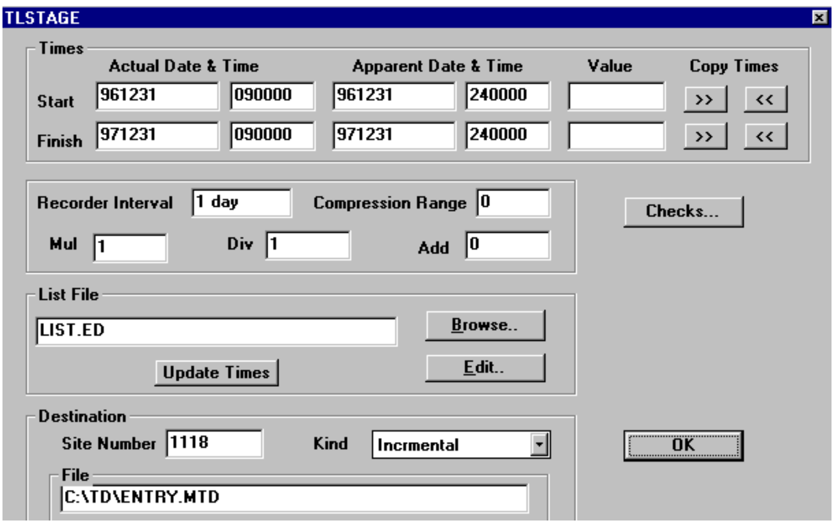
When only part of a year is entered, the last 4 of these numbers identify which part.

The text input is actually in Worksheet(1), and we have referred to it as Sheet1, because that is its usual name. The Daily macro reads Workbooks(1).Worksheets(1) and after it is run, the List.ed file contains the following text:

```
TIMES 19970101 0 19980101 0
 0       0       0       0       0       32500   9500    0       0       0
 0       1000    27000   4000    7000    22000   100     0       3000    500
 0       0       0       0       0       0       2000    0       0       0
 0       0       0       0       0       1000    9000    18500   3500    16000
 0       100     100     3000    3500    0       0       0       0       0
 0       23000   0       0       0       5000    0       2000    2000    0
 0       0       0       100     0       10000   26000   12000   13000   0
 0       0       5000    0       0       5000    0       0       0       0
 500     36000   1000    13000   8000    0       6000    0       0       0
 0       0       1000    0       0       2000    19000   5000    6000    6000
 6000    0       4000    0       7500    4000    8500    5000    0       500
```

9000	1000	0	4000	0	11500	11500	100	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1000	0	500	0	0	0	0
0	0	2000	5500	3000	5000	100	3000	11000	1000
0	4000	2000	2000	65000	8000	100	0	0	0
0	0	0	6700	6700	6700	0	0	2000	28000
12000	0	0	0	0	0	7000	0	0	28000
0	0	8000	28000	14000	17500	500	0	0	0
0	2500	15000	13000	500	25000	13000	0	0	0
500	0	0	0	0	0	0	0	0	0
0	0	1000	4000	9000	2000	22000	22000	4000	2000
0	0	0	500	2500	2000	8800	8800	500	0
0	5500	5000	1500	300	0	33500	33500	2000	2500
3500	3500	0	0	0	6500	6500	3000	0	2000
0	0	0	0	0	9000	9000	0	0	0
0	7000	1000	6000	0	0	0	18000	500	0
0	2000	0	30000	29000	500	0	13000	62000	39000
10000	0	0	1500	2000	0	0	7000	54000	1000
2000	0	0	1500	1500	5000	0	0	0	0
3000	0	0	0	0	0	0	0	1000	0
0	6000	6000	0	15000	11000	7000	23500	0	0
8000	0	7000	4000	0	31000	6000	0	0	4000
0	0	18000	18000	0	11000	0	24000	5000	0
7000	1500	0	0	0	3000	0	0	0	0
0	17000	0	7000	1000	10000	100	0	4000	6000
0	0	0	5500	5500	0				

Click **ENTRY**, click **LIST TO TIDEDA**, click **SERIES VALUES** to get this dialog where the time of day of the readings can be entered; in this case as 9 am >.



Click **OK** to enter the data into the **Destination File**.

Editing macro code

MACRO CODE

The Visual Basic macro code in Tlist.xls can be used by both the 7.0 and 97 versions of EXCEL, but the procedures for editing and running it differ. To see the code:

in EXCEL 7.0

Click the Module1 tab at the bottom of the EXCEL window

in EXCEL 97

Click Tools, click Macro, click Visual Basic editor, click Modules, click Module1

<u>or</u> press Alt+F11.

When editing the macro code to match the requirements of your particular text, you can easily switch from viewing the text to viewing the code, and back again, by using just two buttons:

in EXCEL 7.0 they are items

in the Window menu at the top of the screen (*.txt) & (Tlist.xls)

in EXCEL 97 they are tabs

at the bottom of the screen (Microsoft Excel) & (Microsoft Visual Basic).

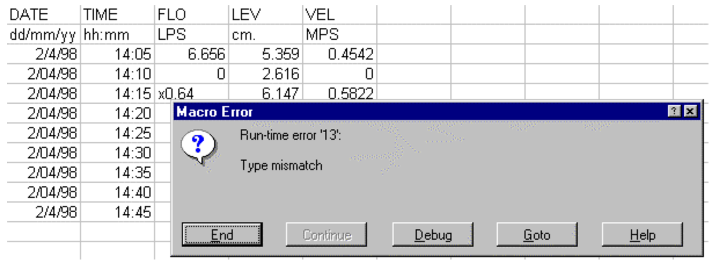
You can process several proprietary texts in the same EXCEL session provided you assign different names to the List file; ie List.ed becomes List.ed1, List.ed2, etc. By closing each proprietary text file before you open the next, they can all be read by the macro as Workbooks(2).Worksheets(1).

Code to write the header can be more than half the code, but if it causes problems it can be omitted altogether because you can type the times into a LIST TO TIDEDA process dialog later on. In the Time_stamps and Series macros, which have less processing of dates and times than the other macros especially if the header is omitted, the few parameters that will most likely be changed when adapting to a new proprietary text are underlined. One of these 2 macros is probably the best to begin with.

In the code listing at the end of this chapter, the 4 principal macros are marked XXXXXXXXXX.

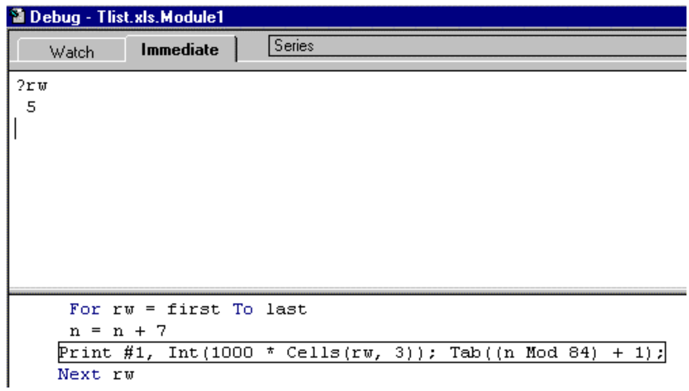
MACRO DEBUG PROCEDURE

The following illustrates the debug procedure. (The first 3 pictures are the Excel 7.0 versions.) We start by doctoring a FLO value in the Worksheet created when Series_values_example.txt is opened in EXCEL. In particular we change the value 10.64 to x0.64.



We then run the Series macro and the error we introduced causes the program to stop showing the Macro Error dialog>

In the Macro Error dialog click the Debug button to open the Debug dialog >



The error occurred in the boxed statement because it could not multiply the contents of the cell containing x. To find out which cell, put the cursor in the Immediate window above the code and type: ?rw <enter>. The response will be the current value of rw, which is the number of a row in the Worksheet.

We now go to the worksheet:

- in EXCEL 7.0 click Window, click Series_values_example.txt
- in EXCEL 97 click Microsoft Excel tab at bottom of window

We can easily find the offending x in row 5 and replace it with a number.

We now go back to the Visual Basic editor, turn off the debug and restart the macro:

- In EXCEL 7.0 click Window, click Tlist.xls

Click Run in the program banner >

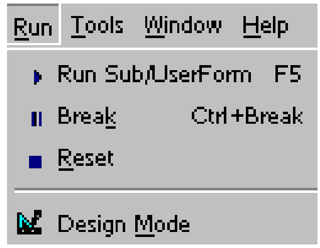
Click Reset in the drop down list >
Click Start in the revised drop down list.

Do not use the Continue option without first clicking Reset unless you are sure about what you are doing.

- In EXCEL 97 click Visual Basic tab at bottom of window

Click Run in the program banner >

Click Reset in the drop down list >
Click Run Sub in the drop down list >



MACRO CODE listed

```
'OPEN THIS WORKBOOK FIRST THEN
'OPEN THE DATA IN A SECOND WORKBOOK

Dim first, last, xltime As Double
        'THESE 3 VARIABLES ARE PASSED TO FUNCTIONS

Sub Time_stamps()   'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
            'OPEN STAMPS_EXAMPLE DATA WITH DELIMITER = COMMA
            'TRANSLATE USING OPTION "TIME STAMPS"
Dim cel As String
            'ACTIVATE THE WORKSHEET WITH THE PROPRIETARY TEXT
Workbooks(2).Worksheets(1).Activate
            'FIND FIRST AND LAST ROWS OF THE TIME SERIES
For rw = 1 To 64000
    cel = Cells(rw, 1)
    If Left(cel, 3) = "PRG" Then first = rw + 1
    If Left(cel, 3) = "Rec" Then last = rw - 1: Exit For
Next rw
If first > 0 And last > first Then
            'OPEN THE LIST FILE
    Close #1
    Open "c:\td\list.ed" For Output As #1
    Range(Cells(first, 1), Cells(last, 1)).NumberFormat _
        = "hr:m:s"
            'HEADER CAN BE OMITTED HERE AND PUT IN LATER
    Print #1, header_time_stamps
            'PRINT THE SERIES
    For rw = first To last
     n = n + 7
     Print #1, Int(86400 * Cells(rw, 1)); Tab((n Mod 84) + 1);
    Next rw
    If Cells(last, 1) = 0 Then Print #1, "1"
    Close #1
Else
    MsgBox "no data found"
End If
End Sub
Function header_time_stamps() As String
Dim a As Double
Cells(first, 2).NumberFormat = "d/m/yy"
xltime = Cells(first, 1) + Cells(first, 2)
header_stamps = "TIMES" & ymd_hms
Cells(last, 2).NumberFormat = "d/m/yy"
xltime = Cells(last, 1) + Cells(last, 2)
header_stamps = header_stamps & ymd_hms
End Function


Sub Series_values()   'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
            'OPEN SERIES_VALUES_EXAMPLE DATA WITH DELIMITER = TAB
            'TRANSLATE USING OPTION "SERIES"
Dim cel As String
            'ACTIVATE THE WORKSHEET WITH THE PROPRIETARY TEXT
Workbooks(2).Worksheets(1).Activate
            'FIND FIRST AND LAST ROWS OF THE TIME SERIES
For rw = 1 To 64000
```

```
    cel = Cells(rw, 1)
    If Left(cel, 3) = "dd/" Then first = rw + 1
    If Left(cel, 3) = "" Then last = rw - 1: Exit For
Next rw
If first > 0 And last > first Then
            'OPEN THE LIST FILE
    Close #1
    Open "c:\td\list.ed" For Output As #1
            'HEADER CAN BE OMITTED HERE AND PUT IN LATER
    Print #1, header_series_values
            'PRINT THE SERIES
     For rw = first To last
     n = n + 7
    Print #1, Int(1000 * Cells(rw, 3)); Tab((n Mod 84) + 1);
    Next rw
    Print #1,
    Close #1
Else
    MsgBox "no data found"
End If
End Sub
Function header_series_values() As String
Cells(first, 1).NumberFormat = "d/m/yy"
Cells(first, 2).NumberFormat = "hh:mm"
xltime = Cells(first, 1) + Cells(first, 2)
header_series = "TIMES" & ymd_hms
Cells(last, 1).NumberFormat = "d/m/yy"
Cells(last, 2).NumberFormat = "hh:mm"
xltime = Cells(last, 1) + Cells(last, 2)
header_series = header_series & ymd_hms
End Function


Sub Fully_specified() 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    'OPEN REYNOLDS_BACH DATA WITH DELIMITER = |
    'TRANSLATE USING OPTION "FULLY SPECIFIED"
Dim xltime As Double, seq1 As Double, seq2 As Double
Workbooks(2).Worksheets(1).Activate
            'SPECIFY YEAR
yr = 1997
            'OPEN LIST.ED AND PRINT SITE, NUMB & KIND
Close #1
Open "c:\td\list.ed" For Output As #1
Print #1, "333 3 instant"
        'LOOP THROUGH DATA
For rw = 1 To 16280
        'IN ROWS WITH DATA BOTH COL.1 & COL.2 ARE NUMBERS
If Val(Cells(rw, 1)) <> 0 And Val(Cells(rw, 2)) <> 0 Then
            'PUT DATE & TIME IN TXT1
    txt1 = Cells(rw, 1).Text
    leng = Len(Trim(txt1))
            'FIND DATE, IF ANY, & PUT IT IN YMD
    If leng > 9 Then
        For i = 1 To leng
            If Mid(txt1, i, 1) = "/" Then
                txt2 = Trim(Mid(txt1, i - 2, 5)) & "/" _
                        & Right(Str(yr), 2)
                xltime = CDate(txt2)
                ymd = Year(xltime) * 10000 _
                        + Month(xltime) * 100 _
                        + Day(xltime) - 19000000
                txt1 = Right(txt1, 7)
                Exit For
            End If
        Next i
    End If
    'FIND TIME, INSERT ZERO IF NECESSARY, & PUT IT IN HMS
    For i = 1 To Len(txt1) - 1
        If Mid(txt1, i, 2) = ". " Then
            Mid(txt1, i, 2) = ".0"
            Exit For
        End If
    Next i
```

```
        hms = Val(txt1) * 10000
                'PRINT SERIES, ONE TIME TO A LINE
        Print #1, Cells(rw, 2) * 1000, Cells(rw, 3) * 1000 _
                , Cells(rw, 4) * 1000, ymd, hms
                'STOP WITH MESSAGE IF TIME OUT OF SEQUENCE
        seq2 = ymd + hms / 100000
        If seq2 <= seq1 Then MsgBox Str(rw) & _
            Str(seq1) & Str(seq2): Exit For
        seq1 = seq2
        n = 0
    Else
        'STOP AT END OF DATA, IE. AFTER 35 ROWS WITHOUT DATA
        n = n + 1
        If n > 35 Then Exit For
    End If
    Next rw
                'CLOSE LIST.ED SO THAT TIDEDA CAN READ IT
    Close #1
    End Sub


    Sub Daily()  'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
            'TYPE DATA INTO THE WORKSHEET IN THIS WORKBOOK,
            'A MONTH OF UP TO 31 DAYS PER COLUMN
            'IN ROW 32 TYPE: YR, MTHON, DAYON, MTHOFF, DAYOFF
            'EG. 1997 1 1 12 31
            'TRANSLATE USING LIST TO TIDEDA OPTION "SERIES"
    Workbooks(1).Worksheets(1).Activate
    yr = Cells(32, 1)
    cl = Cells(32, 2)
    rw = Cells(32, 3)
    last_cl = Cells(32, 4)
    last_rw = Cells(32, 5)
    xltime = DateValue(Str(last_rw) & "/" _
                    & Str(last_cl) & "/" & Str(yr)) + 1
    header_daily = ymd_hms
    xltime = DateValue(Str(rw) & "/" & Str(cl) & "/" & Str(yr))
    header_daily = "TIMES" & ymd_hms & header_daily
    mth = Month(xltime)
            'OPEN LIST.ED
    Close #1
    Open "c:\td\list.ed" For Output As #1
    Print #1, header_daily
    Print #1, " 0"; Tab(9);
    n = 8
    Do
        n = n + 8
        Print #1, Int(Cells(rw, cl) * 1000); Tab(n Mod 80 + 1);
        If cl = last_cl And rw = last_rw Then Exit Do
        xltime = xltime + 1
        new_mth = Month(xltime)
        If new_mth <> mth Then
            mth = new_mth
            rw = 0
            cl = cl + 1
        End If
        rw = rw + 1
    Loop
    Print #1,
    Close #1
    End Sub


    Function ymd_hms() As String
        'THIS FUNCTION IS USED IN ALL THE SUBROUTINES ABOVE
    ymd = Year(xltime) * 10000 + Month(xltime) * 100 _
        + Day(xltime)
    hms = Hour(xltime) * 10000 + Minute(xltime) * 100 _
        + Second(xltime)
    ymd_hms = Str(ymd) & Str(hms)
    End Function
```

INDEX