



REGATTA

Scoring and Race Management

The Regatta Scoring and Race Management program is a fully menu driven and automated regatta management system. It keeps track of up to 999,999 concurrent regattas, with up to 999 divisions per regatta, and up to 99 races per division for each regatta. The program maintains a permanent database of boats, classes, and yacht clubs. The number of regattas maintained is limited only by hard disk space available.

Results are automatically printed:

- Individual race results in 5 different menu selectable formats.
- Regatta results in a format appropriate to the number of races in the regatta.
- Fully formatted and indexed race results are written to disk in web page ready HTML files. Results can be posted on the Club's web page in minutes simply by uploading these HTML files.

The program can run on the committee boat using a laptop. Be sure the laptop has the capacity required. We have found that calling in race results on a cell phone is marginally satisfactory. The cellular transmission line quality often tends to be unacceptable, and the connection frequently drops out. Too many transcription errors result.

The program is written in ANSI standard C++ code (to the extent that ANSI has a C++ standard), utilizing a proprietary library. The program is compiled into 16 bit executables utilizing 386 protocol, thus requiring 32 bit processors to run. The 16 bit compiler causes memory constraints as will be apparent during use. The pro-

gram is being converted to compile into 32 bit executables, which will alleviate the memory problems. Either way, the program requires a computer system which is 100% 32 bit IBM compatible (Intel 386 DX or better processor) with at least 4 Mb memory, and a non interlaced VGA color display. (There has been trouble with the Compaq Desk Pro models. These computers do not use industry standard VGA protocol). The program has been tested using MS-DOS 6.2. It might well work using DOS 5.1, but we strongly recommend using DOS 6.2 or higher.

There must be 575kb of memory available to run the present (16 bit) program. That means that when you execute the mem command (or chkdisk for older versions of DOS), the free memory available to run programs should be at least 575 Kilo Bytes. We run it with 575kb program memory, and attempts to run it on less than 550kb are usually not productive.

Almost any 32 bit desktop or laptop clone can be set up to run with this much memory available. Just don't load it down with all sorts of TSR's and other cute memory usurping non productive programs. Don't make the computer a network file server, or network print server. Better yet, don't hook up to the network if you have memory problems. You can always boot from a floppy, and make sure the *dos=high* command is in your config.sys file. Don't put anything cute or clever in your autexec.bat or config.sys file. The KISS concept definitely applies (Keep It Simple Stupid!) You might consider installing a memory manager (qemm, 386 max, or memmaker in DOS 6.2). That will help your overall performance anyway.

If you use Windows 95, you may not be able to run out of a plain dos window, depending on how you are configured. It may be necessary to modify the "msdos.sys" file so that the "bootgui=1" line reads "bootgui=0". This suppresses the gui part of windows 95. Then use your standard dos memory manager (dos 6.2 has one, or 286max (that is what I use), or whatever).

The executable program itself requires more than 6.5 Mb of hard disk space. Data files require about 2 Mb per regatta, depending on the number of entries and races each entry sails. As the number of boats on file increases, it is well possible to require a disk partition of 50 Mb. The hard disk should have a 50 Mb partition reserved for the regatta system. With time, one should be prepared to increase the allotted disk space upward.

The program installs on any desired hard disk partition (such as drive *d:*). A regatta menu is then placed in a directory *d:\regatta*, and the program and data files are placed in the *d:\regatta\rg* directory. Be certain these directories are not already in use, as they will be over-written.

The installation procedure is fully automatic.

- Install distribution floppy 1 in a the appropriate floppy drive. For example:

drive b:

- Log on to that drive by typing:

b: <cr>

- Start the installation procedure by typing:

go <cr>

- Answer the questions on the screen, when prompted.

The registration floppy may be installed during program installation, or later by itself. Again, install the registration floppy into the appropriate floppy drive, log on to it, and type

go <cr>

To start the program at a later time, type

regatta <cr>

from the root directory of the *c: drive* or from the root directory of the drive with the program files

in it. (*drive d:* in the example above). Any existing files by that name will be overwritten.

There is no difference between the unregistered program and the registered program, except that the registered version:

- Has a different opening screen.
- Prints the name of the registered yacht club on all printouts.

Aside from the usual scoring reports, the program prints envelopes for any skipper in a regatta, or for the owner of any boat on file, simplifying the distribution of race instructions and results to contestants, as well as mailing race announcements for next year's regatta.

The program utilizes a leveled division structure. This permits scoring at multiple levels, and in multiple groupings. Thus classes, groups, categories, scoring methods, etc., can all be combined for any regatta.

These division levels also provide a convenient platform to compare different rating systems and different time correction methods, as division levels can be used for the different scoring options.

See the sample regatta comparing Time on Time, Time on Distance, and ORCA. These were real results, for a real race. Study the mix of the top 15 boats carefully. Compare the time spread of the bulk of the fleet.

The program accommodates 15 different ratings concurrently. Different rating systems can therefore be compared, by using different ratings for different division levels.

The major data files of the scoring program are:

1. Master files:
 - File of Regattas
 - File of One Design Classes
 - File of Yacht Clubs
 - File of Boats
2. Regatta Specific Files
 - Regatta Divisions
 - Regatta Entries

- Race Information
 - Time Cards
3. Index Files
- Permanent For Master Files
 - Volatile For Active Regatta

The Regatta File contains the description of every regatta entered, so long as it has not been purged. Old regattas can be deleted when it is certain that its information is no longer of interest. Keep in mind that the following year, it is nice to pull up the old regatta, and use it as a template for the new regatta. The old regatta can be copied as a model for the new regatta. When regattas are deleted, various levels of deletion may be selected so that, for example, only the data needed for a template is retained. The reason for deleting old regattas is to reduce hard disk clutter. Since the individual regatta information on the regatta master file is quite minimal, the significant disk space requirement is for the entries and time cards. Since the active regatta data files are separate for each individual regatta, the presence of a large number of files for other regattas does not in any way degrade the performance of the system. That is, if there is sufficient hard disk space.

The Class File contains information on all one design classes which have been entered in any regatta. The class file contains ratings for each class, so that when a new boat is entered of an existing class, all class information is automatically entered. (So long as the new boat is a *strict one design*).

The Yacht Club file contains names and abbreviations for yacht clubs. This information is used for each entry as an advisory only to dress up the printed results.

The Boat File contains information on every boat which has been entered in any regatta. The information includes the name and address of the owner, the yacht club represented by the owner, the class of boat (if any), the class ratings, as well as the ratings of the individual boat. The latter may be different from the class ratings when the boat is not a strict one design boat, although she belongs to a one design class.

The boat file may be automatically updated from the class file. If at the beginning of the year ratings are revised, one need only to enter the ratings in the class file. Ratings in the boat file will be automatically updated, for all boats which are marked as strict one design.

Data files are not maintained in any particular order, but are indexed by a number of attributes, using ordered index files. Because these index files may get out of whack, reindexing of data files should be done on occasion. Whenever the accuracy of results is in doubt, or when data records appear to be missing, reindex the data files.

Master files relate to every regatta, and their index files are maintained continually.

The regatta specific files deal with one individual regatta, and their index files are maintained only while the regatta is the active regatta. When a regatta becomes the active regatta, its index files are automatically generated. That is how it is supposed to be, but re-index manually if in doubt.

The active regatta is defined by a number of divisions, entries, starting times, course lengths, finish times, rating methods, scoring methods, etc. The index files for these operations are numerous, and are maintained only while the regatta is active. The good news of this is that disk space is freed up, the bad news is that the active index files must be generated every time a regatta is activated. The good seems to outweigh the bad.

Every regatta has at least one division, and can have up to 999 divisions. Divisions are ranked into levels:

1. Race Level
2. Scoring Level
3. Scoring Level
4. Scoring Level
5. Scoring Level

The highest division ID number cannot exceed 999. When a division is deleted, its number does not become available. The program internally assigns division numbers, and this is transparent to the user. What this means is that you cannot enter more than 999 new divisions in a regatta,

because the division number indexes every time a new division is entered.

Every race is sailed at division level 1, the race level. Every boat must be assigned to a division at division level 1. Guns are fired only for race level divisions.

Division levels 2, 3, 4, and 5, are scoring levels only, and need not be used in any one particular race. Any boat in any race level division, may be assigned to any one division in each of the higher division levels. Thus every boat can be assigned to one division in each division level. However, every boat must be assigned to a level 1 division, otherwise she cannot sail the race. Level 0 is the all entries combined division if selected.

Any boat which signs up for a regatta is an entry. A boat is entered by selecting its sail number, checking to see if her skipper is the same as the owner. It is quickest to enter classes and boats in the master files, but it can be done on the fly during participant entry.

Since boats sail under different rating systems, etc., most boats have more than one rating depending on the rating system which will be used for the regatta. Quite often a regatta will sail different divisions under different rating systems, and then use a third for overall. The program allows every boat to have 15 different ratings for different rating systems. Because each boat usually has a rating type under which it prefers to sail, each boat has a default rating type (which is automatically copied from the class information, but which can be readily changed for the individual boat).

Once boats are entered, they can be printed out by sail number, or by rating. The default rating has as purpose to separate the boats by rating type at this point. These lists are helpful to the race committee while dividing entries into the required classes.

Entries are assigned to divisions through Division Assignment. Select the target division (by level if appropriate), and enter the sail number. All boats are now entered, and assigned to divisions, at all levels. Although index files are automatically generated whenever a regatta is activated and

whenever changes warranting reindexing are made, it is prudent to make sure all index files have been generated and are current. For this purpose, menu selections are often available to force certain files to be reindexed.

Race packets are now printed, either a race at a time, or all races at once. These packets contain the information which simplify life for the "on the water" Race Committee. Starting lists, starting order, duplicate sail number warnings, finishing sheets, etc. Race Committees are pleased to receive these packets one starting sequence per manila envelope, and with the race number boldly marked.

The second part of the race is actually running it.

Once all boats are entered, the next step is to verify the time cards.

Each race sailed has a race information card assigned to her by the program. 5 races for 3 divisions means 15 race information cards.

It is important to record the starting time of each individual race. This is a requirement of the system. If a race does not start, it cannot finish.

Each entry has a time cards assigned to her by the program for each scheduled race in the regatta. 20 entries in a 7 race Olympic event makes 140 time cards.

Boats starting or ranking as starter must be recorded. It is important to note the different kinds of finishing status (maybe it should be called starting status). These are:

- DNC Not showing up at all at the starting area
- DNS Sailing around in the vicinity of the starting line after the 5 minute signal
- DNF Actually crossing the starting line after the starting signal

The program allows to automatically record any boat as having started when entering a finish time, (it nevertheless checks to see that the boat is actually entered), or it can require that a boat is recorded as a starter, or that she has actually started, before it will accept a finishing time for that boat.

This prevents a boat from counting as a finisher if she has not first been recorded as a starter. The automatic starting recording is by far the most convenient, but there are races where strict finishing status recording is required.

Once finish times have been entered, all data entry is completed for printing the results.

Finishing entry can be performed by finishing position for one design classes if desired. This is accomplished by simply entering the sail numbers in order of finish.

Normally finishing data is entered by finishing time. This is also available for one design racing. The easiest way is to enter the finishing times in order of finish. This is done by entering the sail number and finishing time in order of finish. Once the first finishing time and date has been entered, all further times are entered by sail number and minutes and seconds only. Whenever the number of minutes decreases, the hours index.

This is very quick, and makes fewer errors than entering the hour as well for every finisher. There is an option which suppresses the automatic hour indexing, but this is quite a bit slower. It should, however be used for making corrections to existing finishing times, so that the hours do not inadvertently increase.

Actual races are computed through the race result section of the program. Select the division and the race number and tell it to proceed. One of the options is to force a recalc of everything.

Multiple races per division can be readily calculated. Just select the race number and the division.

To make the program work the way which is most convenient (not to mention work at all) please write down all improvements you would like to see incorporated in the program. Write down the keystrokes how to get there.

Some Idiosyncrasies Of The Computer Program

The following are some, but by no means all, of the idiosyncrasies of the Regatta program.

Esc usually get you out of trouble without destroying anything. No promises, but we tried.

Selection boxes respond to the number of the selection, Page Up, Page Down, Home, End, and the curser arrow keys. Space bar doubles as a 0 selection.

Side bar selection lists respond similarly, but also to typing the sail number or class name and then hitting Enter.

Fill in screens respond to Curser Arrow keys, Tab and Shift Tab keys, Ctrl Home, Ctrl End, Page Up, Page Down. In addition, Esc removes the light bar the first time it is hit. If you then wish to continue data entry, just hit enter again (or any key for that matter) and the light bar re-appears. If the light bar is suppressed, and Esc is hit a second time, then one exits the fill in screen. The light bar also disappears when Enter is hit on the last field of a fill in screen.

One exits a fill in screen by hitting Esc when the light bar is not visible. To make the light bar invisible, hit Esc when it is visible. This means that Esc Esc exits from a fill in screen. An alternate means of exiting a fill in screen is to hit Enter in less than two hundred milliseconds when the light bar is visible. In other words, a quick double tap on the Enter key.

When filling in data on a light bar active data field, the data does not become permanent, until you hit enter, or somehow else exit the field (Cursor Arrows, etc.). If you just hit Esc, then the original value in the data field will reappear, negating your efforts to enter a new value. So: first hit Enter, then Esc.

Drop down lists are plain selection boxes embedded in a selection screen. They are activated by

punching F2 or F10. One of these days we are going to make either of these keys accept the screen as is (i.e., to replace the <cr> Esc Esc sequence). Stay tuned for that change.

For any Yes/No question, the program looks for an “n” or an “N”, or a <cr>. Anything else is a *yes*. Only the “n” and “N” are *no*. Thus the space bar and zero and h are all *yes*. “Certainly” and “Affirmative” are *yes*, but “Naturally” is *no*. If the question is *Yes or no*, then <cr> is *yes*. If the question is *yes or No*, then <cr> is *no*. Whatever is capitalized is the default for <cr>.

Using the space bar for yes is very nice and quick, but since it also says yes to a print request, it can cause an inordinate amount of paper to be spewed from the printer. Not all print activity can be aborted with Esc, although it is supposed to work that way.

On text fill in fields, there is usually some sort of automatic formatting. This is both good and bad. To defeat the formatting hit F4, or type over the auto-formatted text, but don't change the spelling. Change only upper and lower case.

The Insert key works on fill in fields, but there is no way to tell if it is toggled on or off other than by trying. Default is over-type. Every new field resets the toggle to over-type.

Home and End get you to the beginning and the end of a fill in field.

Ctrl Home and Ctrl End get you to the first and last fill in field of a fill in screen.

Ctrl <backspace> clears the entire field.

Enter all dates as dd mm yy or dd mm yyyy, where any delimiter will suffice.

Use the 24 hour clock.

Otherwise races spanning the '12' end up funny.