

Development of a Computerized NAR Contest Manager

Research & Development project
NARAM-41

Chris Kidwell
NAR 45225

Abstract

The continuing development of a standalone computer application, ContestManager, is presented. The application is designed to ease the tedious process of data entry and tabulation during and after a contest. Setting up a contest is simply a matter of selecting the events to be flown from a list of choices. All calculations for data reduction and points tabulation are handled internally. The program has been tested at various Open and Regional Meets, as well as NARAM-40 and provided accurate results in real time, rather than waiting for hours after the contest for the results to be computed.

Introduction

Running an NAR contest involves a great deal of data processing and record keeping that the contestants rarely see. The Contest Director has to go through many repetitive calculations for such tasks as averaging timing data, reducing altitudes, averaging scores, and awarding points. The calculations themselves are not difficult, but the sheer number of calculations makes the job tedious and time-consuming. Computers, however, are ideally suited for these types of tasks, so the development of a computerized data-entry system for NAR contests would be extremely beneficial to the hobby.

Several computerized solutions already exist, but are not widely used. A FileMaker Pro database has been used for data-entry at the past 3 NARAMs. This system is extremely difficult to setup for a contest, essentially requiring the author to custom-design the database each time it is used. Operation is also difficult and requires extensive training, which limits the number of users who can do data entry. Spreadsheets have also been used in the past, notably by Glenn Feveryear at RAMTEC-5 and RAMTEC-6. Although the operation of such spreadsheets is easier than the FileMaker Pro database, the spreadsheets must still be custom-designed for each contest.

This project describes the continuing development of ContestManager, which is a standalone application for data entry and processing for NAR contests. ContestManager was originally developed based on Glenn Feveryear's spreadsheet first used at RAMTEC-5. The goal of the project was to develop a user-friendly, standalone application for recording contest flight data and processing NAR points. Preliminary versions of ContestManager were tested at ECRM-25 and RAMTEC-6 regional meets, and presented for Research & Development at NARAM-40. Many positive comments were received at NARAM-40 and have since been incorporated into the final version. ContestManager was also used as a comparison against the FileMaker Pro database for NARAM-40. Several errors in results processing were identified, including one case where 4th place was awarded for a flight which should have only received flight points due to a lack of a return.

Development has continued throughout the past year with extensive testing at meets hosted by NARHAMS, SPAAR, and PARA sections. New features added since NARAM-40 include: support for redundant tracking stations, automatic identification of new records, display of standings by event or meet, printing of results and standings, online Pink Book, support for pre-NARAM points to determine year-end standings, and display of section point totals.

Methods

The ContestManager program is written entirely in the C++ programming language, using Microsoft's Visual C++ Version 5.0 and the Microsoft Foundation Class (MFC) library. Development was on a 180 MHz PentiumPro desktop computer. Field testing utilized a 150 MHz Pentium notebook computer. No personal money was spent on development.

One of the major benefits to the C++ language is the concept of inheritance, in which one object definition can be derived from another. For example, the generic object Shape defines a set of functions for displaying, moving, or saving shapes. One can then derive objects Circle and Square from Shape and be able to use those same functions without redefining them. New functions specific to each object class can also be added, such as calculation of area for Circle and Square. In this example, Circle and Square are said to be inherited from the base class Shape.

This concept of inheritance is used extensively to handle all of the different types of NAR events. Figure 1 shows the class hierarchy for the events. Every event is derived from CFlight, which handles the most basic functions such as formatting results and points for display, which are common to all events. CDuration and CAltitude are both derived from CFlight to provide functions for calculating average times and altitudes, respectively.

A summary of the data hierarchy is shown in pseudo code in Figure 2. CContestManagerDoc, derived from the MFC class CDocument, manages all of the data for a single document representing one contest. The contestType field specifies whether the contest is a Section, Local, Open, Regional,

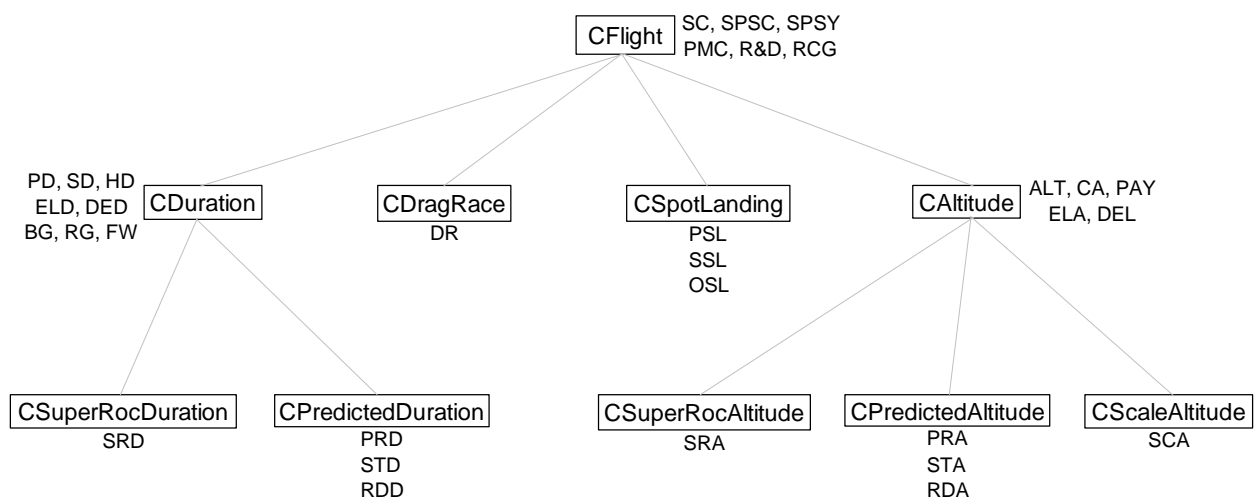


Figure 1. Class hierarchy for events

or National Meet. The Contest Factor, used for calculating contest points, is read from a table for all but National Meets, which are calculated internally based on the events at the meet. The formula for this calculation is shown in Appendix A.

For each event, there is a CEventInfo object, which defines information common to the entire event, such as baseline for altitude events and target altitude or duration for precision events. Each contestant is handled by a CContestant object, which is stored as an array in CContestManagerDoc and sorted by the contestant's name. The NAR number, age division, section number, and section name are also stored for each contestant. The total points awarded to the contestant at this meet are also stored here, along with pre-NARAM points for National Meets.

Flight data is stored for each contestant in an array of CEventHeader objects, one for each event. The CEventHeader object has a pointer back to the CEventInfo object for that event to reference data common to the entire event. The total score for the event and NAR points awarded are also stored here. Each flight is a CFlight-derived object for the particular event and is stored as an array within CEventHeader.

CContestManagerDoc		CEventInfo	
string	contestName	integer	eventID
integer	contestType	integer	weightingFactor
integer	altitudeReductionMethod	float	baseline
boolean	useRedundantTrackers	integer	target
string	trackingStation1Label	boolean	isMultiround
string	trackingStation2Label	integer	multiroundMax
CEventInfo[]	events		
CContestant[]	contestants		
CContestant		CEventHeader	
string	contestantName	CEventInfo*	eventInfo
string	NARnumber	integer	score
integer	division	integer	points
integer	sectionNumber	integer	place
string	sectionName	CFlight[]	flightData
integer	totalPoints		
integer	preNaramPoints	CFlight	
integer	place	CEventHeader*	eventHeader
CEventHeader[]	events	integer	score
		boolean	returned
		boolean	disqualified
		integer	comment

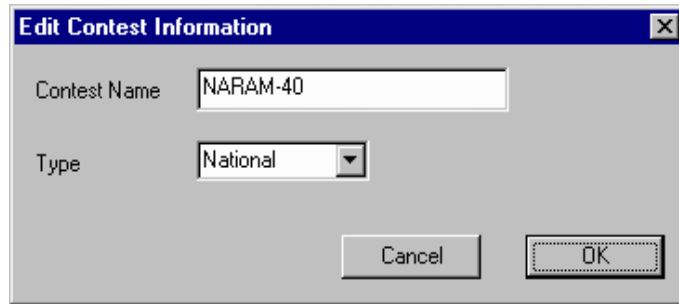
Figure 2. Data hierarchy

At smaller meets, it is often necessary to combine age divisions so that there are at least two entries per combined division per event. The current age divisions are A (7 - 13 years), B (14 - 18 years), C (19+ years), and T (teams). When there is only one entry in a given division for an event, that division must be combined with one or more adjacent divisions according to age. T Division is defined according to the age of the oldest team member, which nearly always makes it equivalent to C Division. ContestManager makes the assumption that T Division is always equivalent to C Division. In this manner, divisions can be combined in various ways such as A/B/CT, AB/CT, A/C/BT, ABC/T, etc.

Prior to updating the standings for each event, ContestManager counts the number of entries per division for that event and identifies which divisions have fewer than two entries. It then proceeds through a long series of logic steps to produce a valid set of combined divisions. To maximize the total points awarded, the number of entries combined is minimized. For example, with 2 A Division, 1 B Division, 3 C Division, and 3 T Division entries, A and B Divisions would be combined. However, with 3 A Division, 1 B Division, 3 C Division, and 2 T Division entries, B and T Divisions would be combined. These combinations are reevaluated for each event, so it is possible to have different division combinations for different events. While this may be somewhat confusing, and therefore rarely done when scoring events manually, it does have the advantage of awarding the maximum number of points overall. Contest Board Chairman Tom Lyon has also given his approval for this method of combining divisions.

Results

When the application is first run, it prompts the user for a name and type for the new contest (Figure 3). After the contest is named, the user specifies events for the contest (Figure 4). The input fields for motor selection, multiround, baseline, and target altitude or duration update automatically

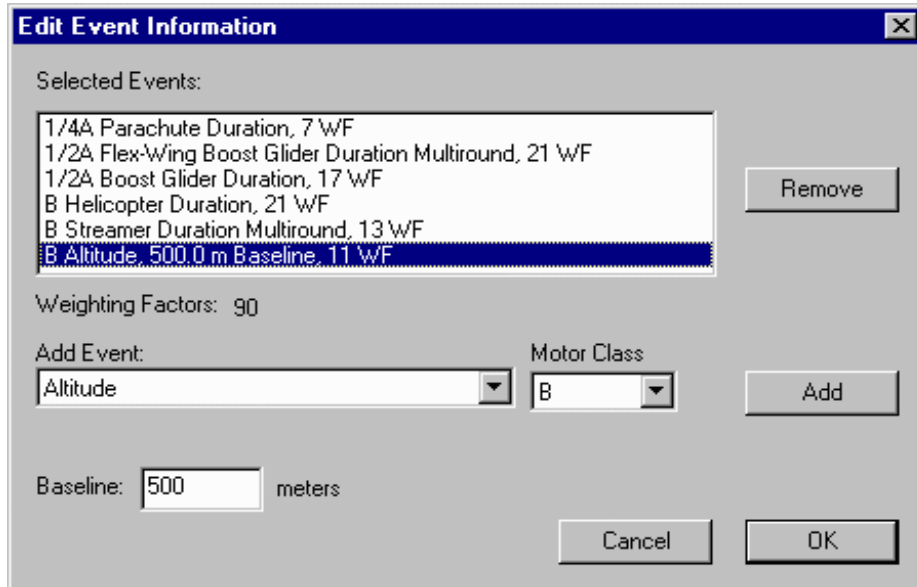


Edit Contest Information

Contest Name:

Type:

Figure 3. Contest Information dialog box



Edit Event Information

Selected Events:

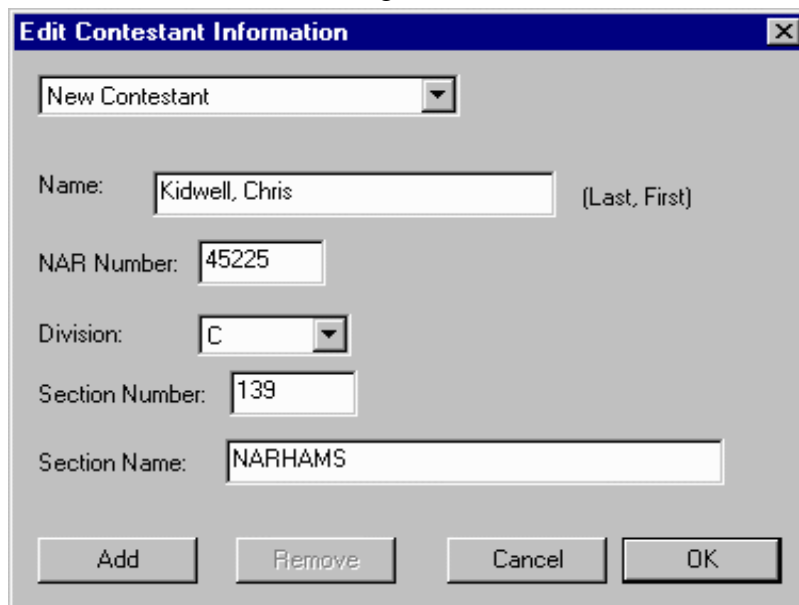
- 1/4A Parachute Duration, 7 WF
- 1/2A Flex-Wing Boost Glider Duration Multiround, 21 WF
- 1/2A Boost Glider Duration, 17 WF
- B Helicopter Duration, 21 WF
- B Streamer Duration Multiround, 13 WF
- B Altitude, 500.0 m Baseline, 11 WF**

Weighting Factors: 90

Add Event: Motor Class:

Baseline: meters

Figure 4. Event Information dialog box



Edit Contestant Information

Name: (Last, First)

NAR Number:

Division:

Section Number:

Section Name:

Figure 5. Contestant Information dialog box

Edit Flight Data [X]

Name: Ash-Poole, Jennifer [v] NAR#: 61415

Event: B Rocket Glider Duration [v]

Flight Number: Flight 1 [v]

Timer 1: 79.38 seconds

Timer 2: 78.07 seconds

Average: 79 seconds

Returned Comment: Official Flight [v]

Disqualified

Add Remove Cancel OK

Figure 6. Flight Data dialog box for duration event

Edit Flight Data [X]

Name: O'Neal, Dave [v] NAR#: 16856

Event: 1/2A Super-Roc Altitude [v]

Flight Number: Flight 2 [v]

Length: 100.0 cm

Azimuth: East 63.0 West 46.0

Elevation: 41.0 34.0

Altitude: 129.3 meters

Closure: 2.9 %

Average: 129 meters

Returned Comment: Official Flight [v]

Disqualified POSSIBLE NEW RECORD

Add Remove Cancel OK

Figure 7. Flight Data dialog box for altitude event

depending on the event selected. The total weighting factors selected for the meet are also displayed, and a warning message is given if the total is greater than the maximum allowed for the type of contest specified. After selecting the events, contestants may be added (Figure 5). The NAR section name is filled in automatically based on the section number. For National Meets, pre-NARAM points are also input so that the national champion standings may be calculated at the conclusion of the meet. Contestants may be added, removed, or changed at any time, so the user need not enter them all at once.

Flights are recorded through the Edit Flight Data dialog box. Examples for duration and altitude events are shown in Figures 6 and 7, respectively. The user selects the contestant and event and enters the information for the new flight. Alternatively, the NAR number of the contestant may be typed in, and that contestant is selected. All necessary calculations such as averaging of timer results, altitude data reduction, and rounding of scores are done automatically upon adding the new flight. There are also boxes to check for return or disqualification, as well as a list of comments, which is normally used as an explanation for a disqualified flight.

Appendix B contains printouts of the various views using the NARAM-40 data file. A summary of all of the flight results in a spreadsheet-like format is shown in Appendix B1. This view provides a fast and convenient way to see all of the results at once. Results shown in red indicate that the contestant has not fulfilled the return requirements for that event, and can only receive flight points. To make data entry easier, double-clicking on a cell automatically brings up the dialog to edit that field. The user can also switch from showing flight results to showing NAR points for each event, as well as totals for each contestant (Appendix B2). This view is particularly useful for the Contest Director to fill in the points form for the Contest Board since the display is nearly identical to that form.

The standings view (Appendix B3) is used to show the places for each entry in each event. This view may be printed in its entirety, or one event at a time, which is useful for posting results for contestants to view at the meet. The meet champions view (Appendix B4) shows the contestants sorted by the points awarded at that meet. The NARAM points view (Appendix B5) is only available for National Meets and shows each contestant's pre-NARAM, NARAM, and total contest year points, sorted alphabetically by the contestant names. Finally, the National Champions view (Appendix B6) displays the same information, but sorted by total contest year points.

The version history for the development of the project is shown in Appendix C. Since the 1.0b4 version was presented at NARAM-40, a great deal of work has been done in adding features, testing, and debugging. ContestManager has been used by NARHAMS, SPAAR, and PARA sections at various Section, Open, and Regional Meets in the past year. As an example of the usefulness of ContestManager, one may compare the post-meet data processing time following RAMTEC-6 and RAMTEC-7. At RAMTEC-6, with 32 competitors, Glenn Feveryear's spreadsheet was the primary results processor, and it took over an hour to finalize the results. At RAMTEC-7, there were 37 competitors, but ContestManager was used instead, and results were finalized in approximately 15 minutes.

Although work will continue to keep ContestManager up-to-date with the latest rules changes, there are very few remaining features to add. Notable additions that are planned for the near future include: automatic identification of maximum times for multiround events, additional checks for valid lengths in super-roc events, support for radio-controlled glider event. Work will also begin soon on a Macintosh version of ContestManager.

Conclusions

ContestManager has proven to be a great success in all respects. It is a standalone application, not requiring the use of Excel, Access, or Filemaker to run. It has been tested at many different types of meets with a wide variety of events. ContestManager provides fast and accurate flight results with unprecedented simplicity and ease of use.

Acknowledgments

I am very grateful to Glenn Feveryear for providing the original idea and many suggestions for improvements to the project. Also thanks to Glenn and Rita Feveryear for putting up with me constantly borrowing flight cards at RAMTEC-6. Jennifer Ash-Poole also provided much assistance in data entry at RAMTEC-6. Thank you to Tom Lyon for donning the Pink Book Lawyer hat to clarify several rules and provide the formula for calculating the NARAM contest factor. Rod Schafer, Dick Freed, and Dan Wolf also provided many useful suggestions. Finally, thanks go to Jim Filler, John McCoy, and Alan Williams for allowing me to test the program at their respective contests.

Appendix A

National Meet Contest Factor Calculation

The contest factor for a National Meet is calculated by dividing the maximum number of flight points possible in a year by the number of weighting factors at the National Meet. The maximum flight points are achieved by flying 4 Regional Meets, with 80 weighting factors each, for a total of 960 points. The contest factor is rounded to 2 decimal places.

$$\text{NARAM CF} = (\text{max FP}) / (\text{NARAM WF})$$

$$\text{max FP} = (80 \text{ WF} / \text{Regional Meet}) * (3 \text{ CF}) * (4 \text{ Regional Meets} / \text{year})$$

For the 1998 contest year,

$$\begin{aligned} \text{max FP} &= 80 * 3 * 4 \\ &= 960 \end{aligned}$$

$$\begin{aligned} \text{NARAM CF} &= 960 / 181 \\ &= 5.30 \end{aligned}$$

Appendix B

ContestManager Printouts from NARAM-40

Appendix C

Version History

1.0b1 May 15, 1998

first major test at ECRM-25, May 16-17, 1998

- Bugs: Eggloft events scored as sum of two instead of best single flight (fixed 1.0b2)
- Scale Altitude puts altitude in score edit field (fixed 1.0b2)
- Scale Altitude should still display points even for TL (fixed 1.0b2)
- Altitude reduction does not check for all 0 (i.e., both stations TL) (fixed 1.0b2)
- Altitude reduction: closure should be absolute value (fixed 1.0b2)
- All places should be awarded, even in the case of ties, per Tom Lyon (fixed 1.0b2)

1.0b2 June 12, 1998

tested at OPOSSUM-2, June 13, 1998

- New: new method for combining divisions. ugly, but it works
- Bug: points need to be recalculated after adding a contestant (fixed 1.0b3)
- Feature Requests:
 - show total points for each contestant (added 1.0b3)
 - output data in text and html formats (added 1.0b4)

1.0b3 June 18, 1998

tested at RAMTEC-6, June 19-20, 1998

- Bugs: Multiround is +4 WF, not +2 (fixed 1.0b4)
 - no return needed for multiround events. up to the CD to ensure only 2 models used
 - scroll bar only scrolls by 5 lines, not 1 page (fixed 1.0b4)
 - page up/page down keys not supported (fixed 1.0b4)
 - average time does not handle time lost by 1 timer (i.e., 0 time for 1 timer) (fixed 1.0b4)
- Feature Requests:
 - show total points by section
 - fill in section names automatically when adding contestants (added 1.0b4)
 - import contestant database for NARAM
 - draw points in red if no returns for a required return event (added 1.0b4)

1.0b4 August 6, 1998

presented as R&D project for NARAM-40. won 3rd place

- New: fixed memory leaks for flight data
- cleaned up flight dialog
- NARAM contest factor calculated correctly
- added geodesic altitude reduction

1.0b5 October 9, 1998

tested at MATTHEW-2, October 10, 1998

- New: NAR number displayed in edit flight dialog
- Entering NAR number in edit flight dialog loads that contestant times in minutes, e.g. 1:30.00, converted to seconds automatically

added redundant tracking by 4 stations
altitude reduction/redundant tracker prefs now stored with each document
cleaned up data structures to remove redundancy. file size reduced by 25%
added check for new records
display standings
printing support added
R&D and drag race scores are places (eg score of 1 = 1st place)
added various warning messages
fixed several bugs with Precision Duration
added online Pink Book

Bugs: changing the name of a contestant after added really screws up ordering
DR does not need a return

1.0b6 November 12, 1998

New: "Score" label changed to "Place" for DR and R&D flight data dialog
fixed a bug in HTML output
fixed a bug with points being awarded for DQs in single-flight events
added a check to make sure everyone has an NAR or Team number
added preference for tracking station names

1.0b7 November 20, 1998

beta tested by Jennifer Ash-Poole

New: fixed bug with weighting factors not displayed properly in Edit Events dialog
fixed bug with "add flight" error messages on changing flights
fixed bug with displaying results of spot landing in cm

Bugs: non-numbers allowed for certain edit fields in flight dialog (fixed 1.0b8)
altitudes calculated by vertical midpoint never close (fixed 1.0b8)
points calculated incorrectly for DQs in events where minimum score wins (fixed 1.0b8)

1.0b8 January 25, 1999

first public release

Bugs: OSL result > 50 m can still place (fixed 1.0b9)
OSL no distance input treated as track lost (fixed 1.0b9)
Standings view wraps around for large contests in Win95 (fixed 1.0b9)

1.0b9 February 25, 1999

New: improved error checking in flight dialog
added field for pre-NARAM points
display meet champions, national champions
display section point totals
events can now be changed after contestants are added
section list moved to external dat file
standings printed 1 event per page
added option to print standings in only one event at a time

1.0b10 May 14, 1999

tested at ECRM-26, May 15-16, 1999

- New: fixed bug which prevented changes to contestant information after entering
fixed bug where section points were not updated when a contestant changed sections
fixed bug where the event info was not displayed correctly on initial update
baseline can now be changed after event is added
target for set/random altitude/duration can now be changed after event is added
fixed bug which caused error message when removing contestant
- Bugs: SRD/SRA only accept integer values for length (fixed 1.0b11)
B division combined with A on some events, T on others (not a bug)
missing time averaged as 0 (fixed 1.0b11)

1.0b11 May 17, 1999

- New: IND listed in section point totals
SRD/SRA lengths now accept decimal values
when only B division needs to be combined, go with fewer entries in A or T divisions
better checking for missing timer/tracking entries

1.0b12 May 20, 1999

- New: fixed bug with initializing of US Records
cluster, payload, eggloft flights automatically marked NR if not returned
cluster, payload, eggloft flights get 0 points if not returned
fixed bug where target/baseline do not update properly after removing an event

1.0b13 June 15, 1999

- New: added NAR/Team numbers to results and points views
added option to print standings as displayed on screen or one event per page
added margins to results and points printouts
added drop down list of section numbers to Add Contestant dialog box

1.0 July 27, 1999

- New: removed close boxes from dialogs
added checks for altitude and timer data for track lost or malfunctioning timer
events with static points count the first "flight" as static, rest as total points
added tracking data calculator for non-contest flights
fixed bug in rounding scores for super-roc
fixed bug in updating pre-NARAM section points in add contestant dialog
added automatic checking for multiround max
fixed bug in displaying section standings
fixed bug which required score for OSL, even if DQ
fixed bug when changing contestant's name

1.1 July 27, 1999

- fixed bug which caused crash on adding contestants