Dublin University Rifle Club

Rifle Club Management System

Design Brief

Mark Dennehy

May 22, 2001

i

Contents

List of Figures

Executive Summary

1 Introduction

The Club at present suffers from an excessive amount of paperwork, duplication and a lack of support information. The nature of the information the Club maintains makes it particularly suitable to computerisation. Hence, this design was created for a system to maintain Club data and provide operational and decision-making support information. It has been named R.C.M.S., for Rifle Club Management System.

2 General Architecture

The Club gathers a large amount of information on its member's progress during the year. The best means for maintaining a store of this data is to place it in a database. SQL databases have had thousands of man-years of work invested in their design and implementation. Capitalising on this is an obvious choice. Hence, an SQL database called PostgreSQL has been selected and installed on the club webserver, durc.tcd.ie; libraries to use this database from several languages have also been installed.

This gives a single repository for all Club data. A single program could concievably be written to produce all required reports for Club use, but this approach has many disadvantages.

1. A large program will take longer to write

2. A large complex program will have more problems and bugs

3. A large program may not encompass all the reports needed, and adding to it to generate further reports will require the programmer to become familiar with the program thus making adding reports a difficult task.

A single program can often be easier for novice users as they only have to learn to use one program. However, this is outweighed by the inflexibility of a monolithic approach. Further, most users will be Range Officers who will only have to use the system to generate Range Officer Reports and that can be handled by a single program which can call other programs for assistance.

Thus the architecture is as follows;

- All information stored in an SQL database

- Several small programs, each of which produces a single report, and which can be used by other programs.

- Output should be in LaTeX format since this allows for output in printed format, acrobat or web-based formats, and only requires a simple text file to do so.

- A single program should be available to produce an RO report with a great deal of work done on the user interface. However, it should be possible to use this program from a standard dumb terminal so the interface should have two modes; text (curses) based and graphical.

Thus we have a large group of utility programs and a single large RO Report program.

3 Data

The data is stored in a single SQL database with the following tables.

This is the main record. Every other record is linked to this one via the **durc_id**

members		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT PRIMARY KEY
surname	VARCHAR(100)	NOT NULL DEFAULT ''
name	VARCHAR(100)	NOT NULL DEFAULT ''
gender	CHAR(1)	
date_of_birth	DATE	
course	VARCHAR(255)	
year	ENUM	'JF'
		'SF'
		'JS'
		'SS'
		'PG'
		'Staff'
		'Faculty'
		'Alumni'
		'Honorary'
		'Other'
		'unknown'
eye_dominance	CHAR(1)	
handedness	CHAR(1)	
pref_air_rifle	VARCHAR(5)	
pref_free_rifle	VARCHAR(12)	
pref_sporter_rifle	VARCHAR(5)	
pref_jacket	TINYINT(2)	
pref_sling	TINYINT(2)	
sling_stop	DECIMAL(32)	
handstop	DECIMAL(32)	
air_class	CHAR(1)	
free_class	CHAR(1)	

INDEX name (surname name)
INDEX (year)

members_phone		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
phone_type	SET	'Term'
		'Home'
		'Mobile'
		'Work'
		'other'
phone_number	VARCHAR(255)	NOT NULL

INDEX phone_number (phone_number)

members_address		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
address_type	SET	'Home'
		'Term'
		'Work'
		'other'
street_address	TEXT	
city	TEXT	
county	TEXT	
country	TEXT	

Note that we don't keep the actual photos in the database, just their location. This way we can copy & modify the pictures with standard programs and the database performance isn't reduced.

members_photo	
durc_id	INTEGER UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
photo_filename	TEXT

One of these tables per discipline tracked to prevent having to use an ENUM with all the various disciplines. This way we can add a club discipline and not have to ALTER TABLEs. Note that we don't track ten-shot series shot at competitions. There's probably a better way to track scores : ideas would be appreciated. Anyone got any better ideas 'bout storing sporter scores ? Has anyone got a better idea about actually keeping them ? What if we start attending competitions ?

members_scores		
members_scores_air_10m		
members_scores_free_25yd_indoor		
members_scores_free_50m_outdoor		
members_scores_sporter_25yd_indoor_4P		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
date	DATE	
witness_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL
score	SMALLINT	UNSIGNED NOT NULL
notes	TEXT	
		KEY score (score)
		KEY witness (witness_RO)

members_no_shows		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
date	DATE	NOT NULL
detail_no	TINYINT	UNSIGNED ZEROFILL NOT NULL
witness_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL
		KEY date (date)
		KEY witness (witness_RO)

members_ids		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
id_type	ENUM	'Student'
		'DUCAC'
		'Honorary'
		'Other'
		NOT NULL
id_number	TINYTEXT	
witness_RO	INTEGER	UNSIGNED ZEROFILL
witness_date	DATE	

As for members_scores, several tables track different ammo types.

members_ammo_usage		
members_ammo_usage_air_sport		
members_ammo_usage_air_finalematch		
members_ammo_usage_winchester		
members_ammo_usage_eley_standard		
members_ammo_usage_eley_club		
members_ammo_usage_eley_match		
members_ammo_usage_eley_tenex		
members_ammo_usage_lapua_super_club		
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL PRIMARY KEY
date	DATE	NOT NULL
detail_no	TINYINT	UNSIGNED ZEROFILL NOT NULL
amount_issued	SMALLINT	UNSIGNED ZEROFILL NOT NULL
amount_used	SMALLINT	UNSIGNED ZEROFILL NOT NULL
issued_by_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL
		KEY issued_by (issued_by)
		KEY date (date)

		duty_roster
date	DATE	NOT NULL
discipline	ENUM	"Air"
		"Free"
		"Sporter"
		"Event"
		"Competition"
		"No Shooting"
		NOT NULL
primary_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL
secondary_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL
TRO	INTEGER	UNSIGNED ZEROFILL NOT NULL
alternate_RO	INTEGER	UNSIGNED ZEROFILL NOT NULL

INDEX (date)
INDEX (primary_RO)
INDEX (secondary_RO)
INDEX (TRO)
INDEX (alternate_RO)
INDEX (discipline)

		detail
date	DATE	NOT NULL
detail_no	TINYINT	NOT NULL
firing_point	TINYINT	NOT NULL
durc_id	INTEGER	UNSIGNED ZEROFILL NOT NULL
confirmed	CHAR(1)	NOT NULL DEFAULT 'N'

INDEX (date)
*INDEX (**durc_id**)*
INDEX (confirmed)

	event
date	DATE NOT NULL
event	TEXT

INDEX (date)

		competition
date	DATE	NOT NULL
competition_name	TEXT	NOT NULL
competition_discipline	TEXT	
firing_points_per_detail	TINYINT	
detail_length	TIME	
start_time	TIME	