



**Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom**

TELEPHONE:
INTERNATIONAL:
FAX:
E-MAIL:

**Cambridge (01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk**

Training

ANSAwise - Exercise Briefings: Sports Event Administration System

Mark Madsen

Abstract

These are the briefing notes for the sports event administration system exercise

[It was originally developed for a specific customer.]

APM.1738.01.01

Draft
Briefing Note

9th July 1996

Distribution:
Supersedes:
Superseded by:

**ANSAwise - Exercise Briefings: Sports Event Administration
System**



**ANSAwise - Exercise Briefings: Sports Event Administration
System**

Mark Madsen

APM.1738.01.01

9th July 1996

The material in this Report has been developed as part of the ANSA Architecture for Open Distributed Systems. ANSA is a collaborative initiative, managed by APM Limited on behalf of the companies sponsoring the ANSA Workprogramme.

The ANSA initiative is open to all companies and organisations. Further information on the ANSA Workprogramme, the material in this report, and on other reports can be obtained from the address below.

The authors acknowledge the help and assistance of their colleagues, in sponsoring companies and the ANSA team in Cambridge in the preparation of this report.

APM Limited

Poseidon House
Castle Park
CAMBRIDGE
CB3 0RD
United Kingdom

TELEPHONE UK
INTERNATIONAL
FAX
E-MAIL

(01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk

**Copyright „ 1996 Architecture Projects Management Limited
The copyright is held on behalf of the sponsors for the time being of the ANSA
Workprogramme.**

APM Limited takes no responsibility for the consequences of errors or omissions in this Report, nor for any damages resulting from the application of the ideas expressed herein.

Contents

1	1	Sports Event Administration Exercise Participant Briefing
1	1.1	Introduction to the problem
1	1.2	Requirements of the Exercise
2	1.2.1	Define the objects
2	1.2.2	Define the associations
2	1.2.3	Start of season scenario construction
2	1.2.4	Meeting scenario
2	1.2.5	Error scenarios
3	2	Sports Event Administration Exercise Participant Summary
3	2.1	Concepts Illustrated
3	2.1.1	Identification of roles of entities
3	2.1.2	Authorisation levels
3	2.1.3	Error detection and correction
3	2.1.4	Implementation-oriented solution
3	2.2	Ideal solution components
3	2.2.1	Object relations and state
3	2.2.2	Error detecting mechanisms
4	2.2.3	Authorisation levels
5	3	Sports Event Administration Exercise Instructor Briefing
5	3.1	Introduction to the problem
6	3.2	Requirements of the Exercise
6	3.2.1	Define the objects
6	3.2.2	Define the associations
6	3.2.3	Start of season scenario construction
6	3.2.4	Meeting scenario
6	3.2.5	Error scenarios
7	3.3	Concepts Illustrated
7	3.3.1	Identification of roles of entities
7	3.3.2	Authorisation levels
7	3.3.3	Error detection and correction
7	3.3.4	Implementation-oriented solution
7	3.4	Ideal solution components
7	3.4.1	Object relations and state
7	3.4.2	Error detecting mechanisms
7	3.4.3	Authorisation levels

1 Sports Event Administration Exercise Participant Briefing

1.1 Introduction to the problem

You have volunteered to create a computerised scoring system for the benefit of a local children's swimming league. Teams get together for competitions called meets during which the children perform in two types of events: figures and routines; your system is concerned only with routines.

Children must provide their names, ages, addresses, and team names to register prior to the meet. To simplify scoring, each contestant is assigned a number.

During a meet, figure events are held simultaneously at several stations that are set up around a swimming pool. There are volunteer judges and scorekeepers. Scorekeepers tend to become burned out, so there is a fair amount of turnover in their ranks. Several judges and scorekeepers are assigned to each station during a meet. Over the course of a season, each judge and scorekeeper may serve several stations. For scoring uniformity, each figure is held at exactly one station with the same judges. A station may process several figure events in the course of a meet.

Contestants are split into groups, with each group starting at a different station. When a child is finished at one station, he or she proceeds to another station for another event. When everyone has been processed at a station for a given event, the station switches to the next event assigned to it.

Each competitor gets one try at each event, called a trial. Just before a trial, the child's number is announced to the child and to the scorekeepers. Sometimes the children get out of order or the scorekeepers get confused and the station stops while things get straightened out. Each judge indicates a raw score for each observed trial by holding up numbered cards. The raw scores are read to the scorekeepers, who record them and compute a net score for the trial. The highest and lowest raw scores are discarded and the average of the remaining is multiplied by a difficulty factor for the figure.

At the beginning of a season, all swimmers will be entered into the system and a season schedule will be prepared, including deciding which figures will be judged at which meets. Prior to a meet, the system will be used to process registrations. During a meet, it will record scores and determine winners.

1.2 Requirements of the Exercise

Your system will be used to store all information needed for scheduling, registration, and scoring. It will be based around the use of WWW-type HTML Forms for data entry and retrieval.

At each stage of the exercise, specify the processing functions needed to support the actions taken by objects.

All data access is subject to the appropriate authorisation.

1.2.1 Define the objects

Make a list of all the objects that need to be present, and their associated data.

1.2.2 Define the associations

Draw a diagram showing how the objects are associated with each other and how they interact. Take into account the different kinds of authorisations needed for data to be modified.

1.2.3 Start of season scenario construction

Construct a scenario for setting up the scoring system at the start of the season.

1. Information on teams, competitors, and judges must be entered.
2. What forms are needed?
3. What fields are needed on each form?

1.2.4 Meeting scenario

Prepare a scenario for scoring on the day of a meeting.

1. Limit the scenario to
 - (i) 2 teams
 - (ii) 4 competitors
 - (iii) 2 stations
 - (iv) 6 judges
 - (v) 4 events
2. Include
 - (i) scoring during events
 - (ii) determination of winners at the end

1.2.5 Error scenarios

Identify as many errors as you can.

1. Which of these lead to
 - (i) errors in results
 - (ii) deadlocks in the meeting

2 Sports Event Administration Exercise Participant Summary

From this exercise one is supposed to gain a reinforced belief in object-oriented analysis, and to experience visualising the solution within a particular implementation framework.

2.1 Concepts Illustrated

Specific concepts this exercise is intended to illustrate are

2.1.1 Identification of roles of entities

The roles that entities take can vary from situation to situation, as when the system is used for processing registrations before a meeting.

2.1.2 Authorisation levels

Not every entity can store data in the system, or alter data that is already stored in the system

2.1.3 Error detection and correction

In an error-critical application such as this, errors must be detected as soon as they occur, and the situation must be corrected back to before the error happened.

2.1.4 Implementation-oriented solution

The idea is to implement this system as a WWW-based database with a front end consisting of a set of HTML Forms for authorisation, data entry, and data updates.

2.2 Ideal solution components

An ideal solution to the exercise should include work on all of the following components

2.2.1 Object relations and state

Each object should have a set of relations to other objects and a clear view of what state it supports- alternatively, what data it is described by.

2.2.2 Error detecting mechanisms

Each object needs to have a method for detecting errors and taking action to correct them.

2.2.3 Authorisation levels

Every object needs to have knowledge of its authorisation to perform data accesses of particular types, and ways of proving these authorisations to other objects when necessary.

3 Sports Event Administration Exercise Instructor Briefing

This is the exercise as presented to the participants

3.1 Introduction to the problem

You have volunteered to create a computerised scoring system for the benefit of a local children's swimming league. Teams get together for competitions called meets during which the children perform in two types of events: figures and routines; your system is concerned only with routines.

Children must provide their names, ages, addresses, and team names to register prior to the meet. To simplify scoring, each contestant is assigned a number.

During a meet, figure events are held simultaneously at several stations that are set up around a swimming pool. There are volunteer judges and scorekeepers. Scorekeepers tend to become burned out, so there is a fair amount of turnover in their ranks. Several judges and scorekeepers are assigned to each station during a meet. Over the course of a season, each judge and scorekeeper may serve several stations. For scoring uniformity, each figure is held at exactly one station with the same judges. A station may process several figure events in the course of a meet.

Contestants are split into groups, with each group starting at a different station. When a child is finished at one station, he or she proceeds to another station for another event. When everyone has been processed at a station for a given event, the station switches to the next event assigned to it.

Each competitor gets one try at each event, called a trial. Just before a trial, the child's number is announced to the child and to the scorekeepers. Sometimes the children get out of order or the scorekeepers get confused and the station stops while things get straightened out. Each judge indicates a raw score for each observed trial by holding up numbered cards. The raw scores are read to the scorekeepers, who record them and compute a net score for the trial. The highest and lowest raw scores are discarded and the average of the remaining is multiplied by a difficulty factor for the figure.

At the beginning of a season, all swimmers will be entered into the system and a season schedule will be prepared, including deciding which figures will be judged at which meets. Prior to a meet, the system will be used to process registrations. During a meet, it will record scores and determine winners.

3.2 Requirements of the Exercise

Your system will be used to store all information needed for scheduling, registration, and scoring. It will be based around the use of WWW-type HTML Forms for data entry and retrieval.

At each stage of the exercise, specify the processing functions needed to support the actions taken by objects.

All data access is subject to the appropriate authorisation.

3.2.1 Define the objects

Make a list of all the objects that need to be present, and their associated data.

3.2.2 Define the associations

Draw a diagram showing how the objects are associated with each other and how they interact. Take into account the different kinds of authorisations needed for data to be modified.

3.2.3 Start of season scenario construction

Construct a scenario for setting up the scoring system at the start of the season.

1. Information on teams, competitors, and judges must be entered.
2. What forms are needed?
3. What fields are needed on each form?

3.2.4 Meeting scenario

Prepare a scenario for scoring on the day of a meeting.

1. Limit the scenario to
 - (i) 2 teams
 - (ii) 4 competitors
 - (iii) 2 stations
 - (iv) 6 judges
 - (v) 4 events
2. Include
 - (i) scoring during events
 - (ii) determination of winners at the end

3.2.5 Error scenarios

Identify as many errors as you can.

1. Which of these lead to
 - (i) errors in results
 - (ii) deadlocks in the meeting

This is the summary as presented to the participants.

From this exercise one is supposed to gain a reinforced belief in object-oriented analysis, and to experience visualising the solution within a particular implementation framework.

3.3 Concepts Illustrated

Specific concepts this exercise is intended to illustrate are

3.3.1 Identification of roles of entities

The roles that entities take can vary from situation to situation, as when the system is used for processing registrations before a meeting.

3.3.2 Authorisation levels

Not every entity can store data in the system, or alter data that is already stored in the system

3.3.3 Error detection and correction

In an error-critical application such as this, errors must be detected as soon as they occur, and the situation must be corrected back to before the error happened.

3.3.4 Implementation-oriented solution

The idea is to implement this system as a WWW-based database with a front end consisting of a set of HTML Forms for authorisation, data entry, and data updates.

3.4 Ideal solution components

An ideal solution to the exercise should include work on all of the following components

3.4.1 Object relations and state

Each object should have a set of relations to other objects and a clear view of what state it supports- alternatively, what data it is described by.

3.4.2 Error detecting mechanisms

Each object needs to have a method for detecting errors and taking action to correct them.

3.4.3 Authorisation levels

Every object needs to have knowledge of its authorisation to perform data accesses of particular types, and ways of proving these authorisations to other objects when necessary.

