

## Automated Commentaries for Simulated Soccer

### Comment Class

Audience	All
Author	Justin Hogg
Scope	
Date Created	15/02/2007
Version Number	0.1
Version History	0.1 – Original Document (JH)
Reviewed	Yes
Last saved by Justin Hogg, 18/03/2007	

## **Sign-off sheet**

**Date: 18/03/2007**

**Document Author Signature: JH\***

**Document Author Name: Justin Hogg**

**Quality Assurance Signature: AS\***

**Quality Assurance Name: Akbar Sherwani**

**Project Manager Signature: AM\***

**Project Manager Name: Ahsan Musssa**

\* By signing this document you approve that the entire contents of the deliverable has been reviewed and is in line with the objectives of the project.

## **1. Introduction**

As the project concerns audio commentary using Text-To-Speech (TTS), it was necessary to represent a comment as an object that stores the textual comment parts and other variables that allow it to be used effectively by the rest of the system.

### **1.1 Overview**

This is a simple class created to encapsulate all attributes relating to a comment and its use in the system. It provides methods that return these attributes for processing.

### **1.2 Comment Object**

Each Comment object has at least one string part that represents the text to be passed to the speech engine. Some comments can be stored using a single string as no variables are required – “kick in” or “he has wandered offside” are complete comments.

More informative comments require variables - “red 5 shoots from 18 yards” - for example, requires the player number and distance the ball has travelled. These types of comment are split into string parts at the points where a variable is necessary. A comment object has getter methods to return these unchanging parts in the order they make up the complete comment, thus allowing these parts to be merged into a complete string that contains the comment and its variables. The maximum number of string parts a comment can have is four.

The comment object also has a priority rating, stored as an int, that determines its importance on a scale of 1 to 12, 1 being the most important. A getter method exists to allow the system to prioritise the selection of comments to be spoken, based on the most important and interesting events.

## **2. Evaluation**

This class works well as a template for a comment. It offers the flexibility to hold a complete comment in a single string, or hold component parts of a comment (that do not change), that can be accessed for mark-up with variables.