MEMIS - MHEG ENVIRONMENT FOR MULTIMEDIA INFORMATION AND SIMULATION

Alan Gauton, Tariq Durrani

Signal Processing Division University of Strathclyde
George Street Glasgow G1 1XW Scotland UK
e-mail : ag@spd.eee.strath.ac.uk, durrani@strath.ac.uk

ABSTRACT

MHEG represents a new multimedia and hypermedia standard proposed by ISO/IEC. This paper presents a new software authoring environment based around MHEG-5 that offers users a vehicle for creating multimedia applications that can interact with external programs which involve intense computational tasks. MEMIS provides a linkage between a multimedia front-end and externally available computational processes.

The paper provides a background to the development of the environment, by identifying the facilities offered by MHEG, discusses the efficacy offered by MHEG Vs JAVA for multimedia development; and then covers the development process and includes specific exemplars of the environment for managing multimedia applications which include (a) real-time signal processing embedded within a LabVIEW kernel, (b) ATM, (c) Set Top Box, and (d) Kiosk for Internet commerce that utilises MATLAB type calls. The results includes system level architecture for multimedia implementation, and the timing requirements for such applications.

1.   INTRODUCTION

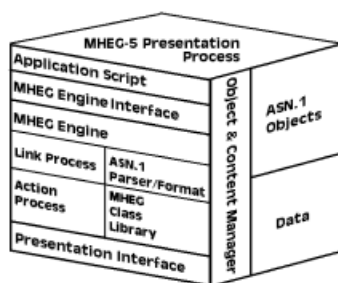### 1.1 MHEG - a new paradigm for multimedia presentations

Figure 1 - the MHEG-5 Presentation Layer

MHEG is a standard proposed by the ISO/IEC and others for hypermedia and multimedia presentations. MHEG stands for **M**ultimedia and **H**ypermedia Information Coding **E**xpert **G**roup [5], and it comprises several constituent parts. These parts cover different levels of platform, different target areas for applications, ensuring correctness of scripts, etc.

The MEMIS project is based around Part 5 of the MHEG standard [4], which is intended primarily for use in lightweight platforms such as set-top boxes; and as with all MHEG parts, requires the use of a script applied to an engine to be executed. As with other systems, MHEG requires authoring tools for the successful and efficient creation of applications. At present a number of authoring systems have been written for different versions and platforms supporting MHEG [1, 3], but MEMIS is the first of these to truly attempt linking applications together in a way which allows both local and external processes to be driven within a common framework. Figure 1 shows the relationship of the different activities within the MHEG-5 presentation process, starting from the scripts generated by these tools, and how they interact.

1.2 Current Status of MHEG

There are several different parts to the overall MHEG standard at varying stages of becoming full international standards. Parts 1, 3, 4, and 5 have been fully published [5], while Parts 6, and 7 are in progress. Part 2 has been included within other parts of the standard, and related topics, and as such is of little relevance. The topics covered by the different parts are as follows:

1. MHEG Object Representation Base Notation (ASN.1).
2. Not officially included
3. MHEG Script Interchange Representation
4. MHEG Registration Procedure
5. Support for Base Level Interactive Applications
6. Support for Enhanced Interactive Applications (CD Level)
7. Interoperability and Conformance Testing for Part 5 (Preliminary)

These different levels of MHEG are suited to different machine platforms, and are based on different types of architecture. For example, the power required to create applications using Part 1 is quite enormous in comparison to that of Part 5. In general terms, support for Part 5 is currently available for use on PC's and UNIX workstations. Part 6 provides a similar level of performance as Part 5, but has increased reliability on the Java Virtual Machine, while the aim of Part 7 is to provide information and testing facilities for applications encoded through Part 5.

1.3 MHEG vs. Java

This section provides a brief discussion on the similarities, and more importantly, the differences offered by MHEG as against

the ubiquitous Java environment for developing multimedia applications.

As with many aspects of the current trend towards distributed computing, and computing through the Internet, there are several different types of technology available for providing information, including HTML, Java, cgi-scripts, etc.

In many ways the development of MHEG is similar to that of Java, in that it's aim is to allow processing and applications through networks. However, while Java is a programming language, MHEG is more intended as a method of providing a way of creating and executing pure multimedia systems. In addition to this, Java's performance relies on more than the power of the machine it runs on - there are currently several different Java Virtual Machines, and these depend on implementation methods to determine the performance of the applications.

MHEG is a scripted system, and as such requires an interpreted format, though the actual script can be compiled into a more compact form. This script is intended to be portable, as Java is, but while Java is diversified throughout many different possible fields, be it animation, games, spreadsheets, MHEG is pure multimedia. To this end, the simplicity of MHEG is also it's power. While Java is required to spend time evaluating the complex command set attached to a full programming language, MHEG has a far simpler command set, which improves the ability of the MHEG engine to perform it's tasks, and includes the ability to include pre-compiled modules should they be required.

2. MEMIS - APPLICATION

2.1 MEMIS Architecture

The MEMIS environment covered in this paper is not simply an MHEG authoring tool. As mentioned above, the environment is also intended to provide a method of interacting with external applications through a standard interface, thus minimising the effort of learning the different command structures within those applications. In effect, MEMIS drives the applications remotely.

To achieve this level of interaction, MEMIS exists as several different modules. The first of these modules is the main MEMIS environment which allows the user to develop their MHEG based applications, largely through a click and point, or drag'n'drop interface. To allow increased levels of control over the application additional facilities are provided through textual and menu type interfaces. This offers full control of the MHEG functionality, without overcomplicating the interface. This module is the core of the suite, and is subsequently the more complex module. Before an MHEG application can be executed, the MHEG script requires to be generated, and this is the second main module. Figure 2 gives a perspective of the MEMIS Architecture.

The third module is the MHEG engine, which executes the application script, and provides the means of turning a script into the view with which the author and additional users can interact. The engine therefore controls the main multimedia

presentations aspects of MEMIS, while also providing the link to external applications. However, in its aim of being multi-platform, it will allow external applications to be called on the host platform, much as a "C" program can call a system command. This call can be made to a module acting as an interface to external applications, and through this module the different applications can be called, such as MATLAB and LabVIEW, thus allowing control and driving the applications from within MEMIS.
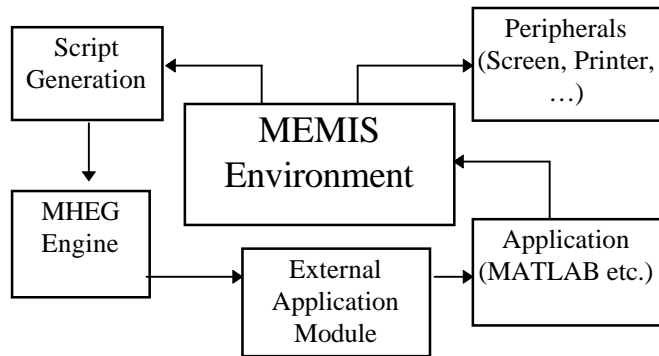

Figure 2 - MEMIS Architectural overview

The level of control available within the environment depends on the methods that can be used within the interface. At present the methods available for use with most modern Microsoft Windows™ applications revolve around OLE (Object Linking and Embedding) and DDE (Dynamic Data Exchange). These systems allow a certain amount of remote control to be applied in terms of being able to call commands within the applications, and also in passing data streams to the applications for processing directly within these applications. MEMIS is intended to tap these technologies and drive applications through this control, before returning their results back to the MEMIS environment.

Finally, the MEMIS architecture is designed to allow users access to a range or peripherals, and this final level of control is provided within the main MEMIS environment. This is an essential item, as at present many current Java applications cannot be displayed in hard copy as the Java virtual machine host cannot provide the required service.

2.2 Advantages of Using the MEMIS Environment

The MEMIS environment is designed to be a simple method for producing multimedia applications through a visual format, where the application is built up by adding visual objects. While this is very practical in terms of adding items such as graphical objects, and textual objects, this causes problems when dealing with items such as audio objects, with no visual form. This is not a real problem however, as the audio objects can appear in a separate list of available objects, used for the final script generation, amongst other uses.

This simplicity also makes the package easy to use, as the method of building up applications is very intuitive, in a similar way to that of working with a web browser. The graphical format is easy to use, and in theory should speed up the development times of the application. This is also an elegant method of working, since the number of times the user has to resort to traditional text based programming is minimal, as choices are largely selected through selection methods of hotspots, menus, or links, and these can largely be handled through menu operations.

Another advantage of MHEG is that it is intended to be a cross-platform system, therefore basic applications created by MEMIS will work on all MHEG engines supporting the scripting method implemented.

The final major advantage of working with MEMIS is that it is based on MHEG, which is the first true multimedia system to achieve ISO standard status. This status means that with the support of bodies such as DAVIC (**D**igital **A**udio **VI**sual **C**ouncil)**[2]**, representing many of the world's leading technology companies, there is a good chance that MHEG will be universally exploited as more set-top box and network computing facilities become available.

3. KEY APPLICATIONS OF MEMIS

3.1 Key Application Developments

To illustrate the power and facilities offered by MHEG, several applications are being developed, including background applications for computationally complex sub-processes. These applications demonstrate the flexibility and capabilities of the MEMIS environment, operating on different formats available at different locations.
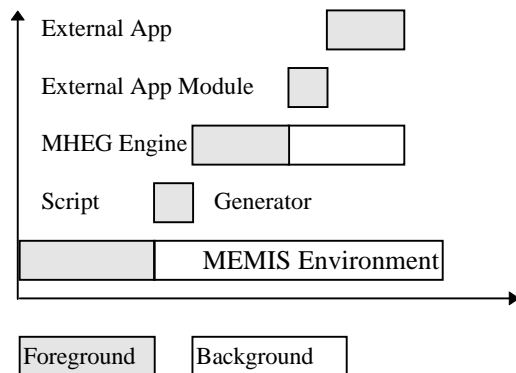

Figure 3 - Overview of Application Timing

Figure 3 gives an overview of the timing cycle which may be required for many of the applications for which MEMIS is suitable. As can be seen, the MEMIS environment is also available in the background, controlling the cycle of Script Generation, Engine calls, as required within the applications developed.

3.1.1 Vibration Analysis

The MEMIS project provides a method of generating applications beyond the scope of traditional MHEG. To facilitate real-time signal processing for Vibration Analysis the MHEG script is enhanced, to allow MEMIS to interact with LabVIEW as a background process, providing analysis of information supplied within the environment. The output from the LabVIEW process is returned to the MEMIS application, providing results in a similar method to any other application included within the MEMIS project. Figure 4 is a snapshot of this application, illustrating a sample input and output display.
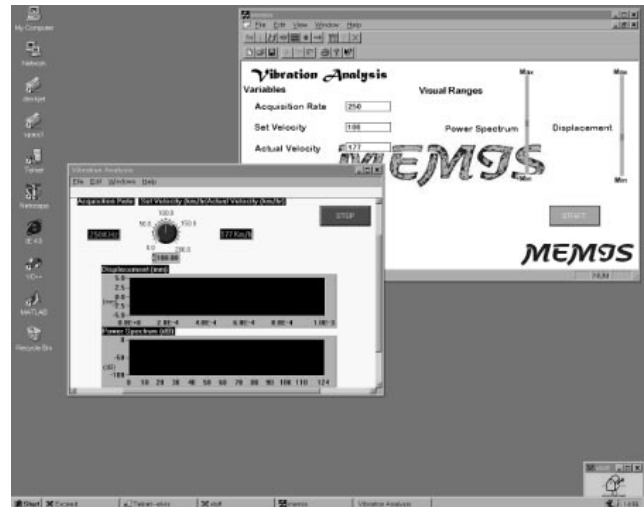

Figure 4 - Sample Vibration Analysis Tool

3.1.2 Advanced ATM

The Auto-Telling Machines (ATMs) currently in use are limited in the facilities they offer to users such as withdrawal of cash, retrieval of mini-statements, and ordering of simple items, such as new cheque-books or information. An advanced ATM using MEMIS can offer many more facilities

An important part of the ATM network is that it should operate in a secure network, so that access to account information is only possible to the client, and the branch network. To this extent, an encryption system is required at the client and server ends of the system. This aspect of security requires a complex algorithm for encryption, which can be implemented using MEMIS' interaction with external applications such as MATLAB. Additional processing tasks related to security can include additional security measures such as either face or retina recognition.

Additional support for MHEG within an ATM increases the methods available for use in publishing additional information services, as MHEG is capable of including data streams such as MPEG, and is capable of presenting information in a more interactive format.

3.1.3 Set Top Box

The Signal Processing Division's first experience of MHEG came from the DIAMOND project [3] with a set-top box that would require an interface to allow the user to interact with a Video on Demand server (VoD), and MHEG was intended as the target system. The Set Top Box can take the core set of functionality designed for use within the ATM as a basis, and can offer additional functions and services such as Share Shop and analysis, bill payments, balance enquiries, financial services, ordering and arranging holidays, and home shopping;

3.1.4 Kiosks & Internet Commerce

Kiosks providing Internet access are gradually becoming available throughout the world, and with this technology more commercial sites are being developed to allow users to purchase goods and services remotely. However, users are still wary of working this way, as they are unsure of the safety of purchasing this way, and revealing card details this way.

Figure 5 - Sample Internet Commerce Application

Given this, the MEMIS project can provide a solution. The MHEG facilities provided are ideally suited for providing an attractive user interface, and by allowing access to external applications such as MATLAB, complex security and database systems can be included, therefore removing a large part of the fears of users.

An example of this type of system is movie ticket ordering, where the user can access a movie theatre database, which will allow users to see previews of a movie, select which movie they wish to see, while the system transfers the correct money from their account, and prints their tickets at the kiosk. Figure 5 provides an illustration of how the preview facility can be presented, allowing users to preview a movie trailer before purchasing tickets.

The above applications are all suitable for development within MEMIS, and are being developed to show the potential of the environment.

4.   DISCUSSION

4.1 Presentation at ICASSP '98

A comprehensive presentation of MEMIS will be made at ICASSP to illustrate the facilities offered by the environment. The presentation will follow the format of the paper in providing an overview of MHEG part 5, and also of the MEMIS environment, and will include an active demonstration of the types of application to which MEMIS will be used. These fall into two distinct categories: those using no external applications, and those which do.

MHEG only applications: An example system using MHEG will be presented, showing the methods used both in creating and executing an application. This will be based round a simple interface, aimed at showing how simple the MEMIS interface is, as well as showing the full range of facilities within MHEG.

MEMIS Enhanced Applications: The second system to be demonstrated will be based on an Internet Commerce system, using local facilities to provide a simulated remote access system. The scenario is that of allowing the user to purchase tickets, and will use the MHEG interface to provide users with a login facility, leading to a sub-system allowing the preview of movie clips before the user selects which movie they would like to see. This system will cover the main aspects of the MEMIS environment, providing an MHEG based user interface, and requiring access to a database system for provision of login and account details, as well as providing access to the movie clips and ordering facilities. This database system, will be linked in as an external application, as well as using additional resources for password encryption and ticket handling as required. Provision of many of the external facilities required are possible using an application such as MATLAB, and this will therefore be used to provide these functions.

5.   ACKNOWLEDGEMENTS

6.   REFERENCES

[1] CSELT "MediaTouch Authoring Tool": http://www.cselt.stet.it/ufv/

[2] DAVIC Web Site: http://www.davic.org/

[3] DIAMOND Project: http://www.octacon.co.uk/proj/diamond

[4] ISO/IEC 13522-5, "Information Technology - Coding of Multimedia and Hypermedia Information", 1997

[5] ISO/IEC JTC1/SC29/WG12 MH Encodings Web Site: http://www.demon.co.uk/tcasey/wg12 (Password Protected Site, carrying official MHEG related information)