Project Proposal Final Version 1st Quarter 2008

Computer Systems Project Proposal - September 2008

Performance and Media Management in C++ by **Michael LeGore**

1. Purpose and scope of the research project

To develop an application capable of managing the communication between many different media resources, and also able to play those media back in the form of events on a timeline. In terms of scope, I would like to make an IDE using Glade interface designer and GTK+ builder class to facilitating the construction of dynamic performances/presentation. I would also like to have support for USB devices to control hardware by writing a small USB protocol for communicating with those devices.

2. Background and review of current literature/research in this area.

Currently, there exist a few programs designed for mixing audio in real time, with separate programs for audio visualization, video playback, and hardware interfacing. Currently, there are no applications that combine these features into one comprehesive program. The purpose of this project is to fill that void.

3. Procedure and Methodology.

I will be using C++ and object oriented design to allow for easy extension of the program, if I or anyone else thinks of features to add on later. To verify the performance of my project, one could test out using the program in it's intended environment. I plan on working with the Drama department in order to organize the possible use of this application in a small play. How I plan to take on designing the application, is that I will write use cases, describing features that a

use might use. In terms of hardware, I will be writing a specification for communication, where a device will identify its methods and variables, and expose them for use by the application. For communicating between resources, I will have a C++ class that manages all of the objects being used by the program, and gives them references to the other resources, for example an audio source has access to an audio buffer. I will be writing in C++, with the possibility of using an embedded ruby interpreter for having ruby script resources as well.

  4. Expected Results & Applications, value to others

I expect that by the end of this project, I will have an application capable of playback of at least sound, and hardware interfacing. I hope, however, that by at the end of the project, I will have a program capable of accessing USB hardware, playing sound back, and a having a GUI for editing. If all goes well the program will be able to save and load the project files, and play them back with or without cueing. I will also be working with the drama department at TJ in order to look into use in a small play, and I have already talked to Kevin Deitz about interfacing with the sound board.

An example of a simple protocol for transmission of data and commands to USB hardware devices. This type of system could be extended and implemented by many different types of microcontrollers and would be readily usable by the media application.

```
//Spec for the performance hardware serial interface
psudocode
// << means write to serial
// >> means read from serial
// all commands end with ;

Establishing contact:

while >> != contact
<< polling;
end
//Identify the device
>> devicetype=<devicetype>;numproperties=<numproperties>;
[property1=<property1>,property2=...]>;]
[numactions=<numberofroutines>;]
[action1([param1,param2,param3]):
[retvalue,retvalue,retvalue];action2...]

//Run routine action with params 4 5 6
<< action1(4,5,6);
>>

//Manually set the variable value
<< set property1=5;
>> 0|1


<< get property1;
>> <property1>;

>> update property1=<property1>;
```