INTERACTION METHODOLOGIES FOR MULTIMODAL COMMUNICATION

PATRICK J. MURRAY

Department of Industrial Engineering
North Carolina State University
Raleigh, North Carolina U.S.A.

ABSTRACT

A multi-modal communications strategy has been under development at North Carolina State University. This strategy is intended to meet the demands of faster and more effective communication between the human user and the computer. However, one aspect in the development of this strategy has not been fully addressed, the interface between the Applications Developer (AD) and the multi-modal interface.

The interface resides between the human user and the AD's application on a host computer. This interface allows knowledge transfer, from the user, to occur in natural modes, e.g. graphics, touch, voice, and keyboard. Although this is indeed beneficial to the user, the key element with this interface is that it appears to the AD as one device, or one input/output source. Therefore the AD can request information from within his application, with the same procedural call format, regardless of the source of this information, e.g. voice, keyboard, or mouse.

The issue we are addressing is the method by which the AD will access this interface. Three methods are currently under investigation: a direct procedural call, a global structure call, and a discrete structure call. A direct procedure call would consist of one call per value required. A global structure call would only require one call, and return values in a structure from which the AD would extract the necessary information. A discrete structure call would also require one call, but would be defined for the number of values required.

We have developed three separate interfaces for investigation, one for each of these methods. Different levels of programmers, assuming the role of the AD, will be used to represent the different programming skills of AD's, novice through professional. A standard coding task will be developed which will have the subjects code input routines and manipulate the returned values based on a given flowchart. The flowchart will provide intersubject consistency.

Several issues are be addressed in the investigation. At what point is flexibility lost by utilizing a simplified format? Will coding or design time be a factor for a method's acceptability? What role will error potential or occurrence play in a method's acceptance? Perhaps the most crucial question is whether this interface actually facilitates the AD's task as well as it does the user's.

The measures that will be collected will include, but will not be limited to, coding and design time, the number and types of coding and design errors, complexity of manipulation based on the returned information, and the consistency between the different levels of programmers.