1. Introduction

Interactivity is very prominently placed in current MPEG-4 activity. However, a well-defined interaction model (which affects several other design decisions) is not specified. Existing standards such as DSM-CC and RTSP support traditional vcr-type interactivity to re-position a media stream during playback. As MPEG-4 intends to support new forms of application specific interactivity, a more complete interactivity model is necessary. The interaction model discussed in this document tries to capture the interactivity requirements and behavior of MPEG-4 presentations.

The current specification just assumes a back channel is available to convey user interaction but it does not say what is carried over the channel. An interaction model is necessary to have predictable uniform behavior in an MPEG-4 presentation. In the absence of an interaction model, we cannot expect a uniform response to user actions on different systems. However, defining an exhaustive list of user interactions and system’s responses is impossible. So a generalized model that captures the most complete interactive behavior should be defined.

Figure 1 shows the exchange of messages between an MPEG-4 client (player) and a primary server. In case of local playback, the player manages the interactions. In the following discussion, server refers to both local playback from a hard disk and remote playback over a network. Explicit distinction will be made where necessary. Remote playback has additional constraints because of the network traffic and QoS issues.

The interactivity MPEG-4 expects to support can be defined as the system’s response to user input. System’s response should be defined in order to have a uniform predictable behavior in an MPEG-4
presentation. User interaction can cause two types of actions: 1) local actions taken by the terminal as defined by the specification. 2) remote actions taken by the server in response to user input.

2. Local Interactivity

Certain user actions cause a change in the behavior of objects in the presentation. The changes are local and no messages are exchanged with the servers. To keep the behavior uniform across all platforms, all local interactivity should be handled via locally generated BIFS updates. The application translates user events (mouse clicks etc.) to BIFS events (change x parameter of object etc.). The translation is up to the application, and not specified by MPEG. The only thing that the terminal needs to know is how to handle BIFS updates, either coming from the server, or locally from the user. This ensures that interaction can affect any aspect of the scene that the application desires.

Such a model implies that an MPEG-4 player should be able to decode BIFS data from more than one channel. The primary BIFS coming from an MPEG-4 file (stream) and the BIFS updates generated locally as a result of user interaction. MPEG-4 specifications should make the support of at least one external BIFS information channel normative.

3. Server-side Interactivity

Certain user actions cause a message to be sent to the server, which cause it to modify the object stream it is delivering by adding an additional object, removing an object etc. The user actions may include clicking an object, inputting a text string, etc.

MPEG cannot possibly define the interactive behavior completely in terms of user events, as it is application specific. However, to support complete application dependent interactivity, CGI-like approach to interactivity should be adopted. This implies the only forms of user input allowed are clicks and data entries. A user input shall cause an application-specific command is sent to the server (if this type of event is specified). The server can then respond by sending a BIFS update command (that changes the view, adds a new object etc.). This allows total freedom for supporting complete interactivity.

To support such an approach, BIFS nodes should have an attribute (s) that give enough information for DMIF to contact the server and send appropriate messages. In case of remote-server interactivity, all the information necessary for DMIF (what is necessary to establish a DMIF session? or to send a message in over an existing session?) to establish a session should be included. This model requires an object’s associated link modifies over time. Allowing objects to have different response to user interaction at different times is a very important feature to support interactive applications. This can be supported by allowing BIFS updates to modify the link information. The nodes that are foreseen to have link attributes are the visual nodes and grouping nodes (if a group of visual objects have the same behavior). To keep the node structure uniform, link attributes may be part of all the nodes and indicated by a flag.

4. Remote-Server Interactivity

In this class of interactions, the MPEG-4 player is interacting with remote servers (secondary servers). All servers other than the primary server fall into this category. For example, accessing HTML page as a result of a user interaction falls into this category. This is differentiated from the server side interactivity in that there is no guaranty that the requested objects can be retrieved or played in time. This is because, when an MPEG-4 session is opened, resources are allocated to the session between the player and the primary server, (including a back channel for client messages). The primary server therefore can satisfy

---

1 Here the word link is used loosely; it refers to the information necessary for DMIF to communicate interactive input.
the all the requests caused because of the interaction. However, there is no guaranty that a user interaction that requires a connection/session to a secondary server shall be satisfied. This results in a situation where it is desirable to have all the important objects in a presentation on the primary server. Further, an object that is inaccessible from a secondary server should not cause disruptions to the current presentations. Once a connection is established to secondary server, the message exchange again shall be cgi-like.

![Figure 2. Message exchange with secondary servers](image-url)