

most of MMS applications, attention is scarcely paid to provide or improve interaction services between end users and servers (e.g. MMS providing servers) and to aim at a MMS available on a server, and a more meaningful local interaction at users mobile terminal is impossible. For example, an associated operation on individual information items in a received multimedia message is not realizable. The contents of multimedia message that can be displayed on a user terminal are limited and, the user needs either by repeating many times of operations or operating on the message displayed to get a further information, or by compiling request message and sending the message thus composed to the server to get more information. Needless to say, it is not only that the message contents that a user can get are limited, but that the procedure a user must take is boring and troublesome, all of these reduce the experiences a user get in multimedia messages.

[0009] As it is shown in FIG. 2, where a MMS structure is simply illustrated, a multimedia message consists of MMS headers and a message body. The MMS headers contain such information on how to transfer the multimedia message from an original to a recipient, as an original address, destination address and etc. The message body consists of several parts including multimedia objects, such as image (jpeg format), text (plain format), audio (way format), each in separate part, as well as the optional presentation part. The presentation part contains instructions to explain how the multimedia content should be rendered. In the prior art there are various alternatives of the computer presentation language, which deals with how to display presentations.

[0010] One of the presentation languages mostly used by those skilled in the art is Synchronized Multimedia Integration Language (SMIL). SMIL is a common deployment for MMS presentation. It is an important way to integrate multimedia into a Web content. SMIL is an extendible markup language (XML) based language. The XML language allows to describe timing indicated by multimedia, to associate super-links with multimedia objects and to define the layout presented on the screen. SMIL is regarded as a way for enriching of the current text message based transmission technologies. SMIL consists of a set of modules that define the semantics and syntax for certain areas of functionality. These modules are layout module, timing and synchronization module and animation module. SMIL merely describes how to show a message on a screen, but it does not provide any interactive features for end users. So the SMIL needs to be extended to support interactive presentation, because interactive messages can provide good user experience to popularize the MMS technology to wider application areas.

[0011] Another presentation language mostly used by those skilled in the art is Hypertext Markup Language (HTML). The HTML is the most popular language supporting interactive presentation in Web applications. But the HTML form (a form presenting as a HTML document) cannot be used in MMS, because the HTML form cannot separate the content from the presentation, in other words, in a HTML form the presentation is often mixed with the data. As matter of fact, there are so many MMS clients with quite different capabilities, and the separation between the content and the presentation would make it possible to support different MMS clients. This is a basic requirement for a MMS message presentation language. Furthermore, the pro-

ocol adopted in the network service program, for example the communication model of Hypertext Markup Language (HTML), is request/response. This model leads to more round trips between the service and the client's mobile terminal. MMS is a message system, which runs over wide-area wireless network. It is not a real-time message. If the request/response model could be implemented, the round-trip will bring huge latency and reduce the performance of the whole system. What is needed is to display a richer dynamic form in an optimum way on a MMS client's terminal.

DISCLOSURE OF THE INVENTION

[0012] In view of the above problems in the prior art the aim of the present invention therefore is to provide an interactive method for multimedia message services.

[0013] An object of one or more embodiments of the present invention is to realize an interactive multimedia communication between subscribers and a MMS back end system, for example a multimedia message providing server, preferably in response to an piece of multimedia message from a server a response message is generated automatically to be sent back to the server without a multiple round-trips between the user terminal and the server, and according to the present invention a flexible, richer and more convenient local interaction at user terminal is possible.

[0014] Another object of one or more embodiments of the present invention is to provide an optimized user interface of the MMS message for users. To this end one or more interactive controllers are embedded into the multimedia message, the user performs the local interaction of the user terminal as well as the interaction between the user terminal and the server (for example a multimedia message sending server).

[0015] Yet another object of an embodiment of the present invention is to create a dynamic Web form, which is displayed on the user terminal, in XForms language, preferably a plurality of dynamic forms included in an single message presenting page to provide richer interactive presentations and more optimized multimedia messages for the user, to enhance his or her experience.

[0016] According to the present invention an interactive method for multimedia message services is provided, comprising steps of receiving a multimedia message from a server, generating a page displaying or playing the multimedia message at a user terminal, wherein one or more controllers are embedded into the displayed or played multimedia message and associations between said controllers are defined; and triggering at least one action event and performing a corresponding action in response to the operations of one or more controllers.

[0017] In one embodiment, said triggered action event is to send back a response message to the server which renders said multimedia message; and in response to the operation of said controllers, automatically generating a response message for request to submit an order.

[0018] In another embodiment, said triggered action event is to modify the content of said multimedia message; and in response to the operation of said controllers, displaying or playing said modified content of said multimedia message.