from the origin server **110**, may be retrieved from the streaming media cache's in-core memory. It will be noted, that the inventive technique of servicing RTSP requests utilizing data stored by the cache **200** in the MMS format described herein is applicable generally to VOD configuration.

[**0029**] If the client-requested data stream is not resident in the streaming media cache, the cache **200** may, in turn, send a request to the origin server to retrieve it. Here, the origin server **110** is a computer that provides the streaming media cache with copies of data streams that may be served to the plurality of clients **120**. In this arrangement, the streaming media cache may serve as an "accelerator" for the origin server. That is, a larger number of clients can concurrently access an origin server's data stream through the streaming media cache than through the origin server itself. As such, the streaming media cache can prevent high volumes of network traffic from "overloading" the origin server's network resources, e.g., its available bandwidth and memory.

[**0030**] A client request for streaming media typically specifies a streaming media format, e.g., MMS or RTSP. The determination of whether the streaming media cache **200** has a previously stored copy of the requested data stream is done, in one embodiment, by a lookup for a media file header. Each media file header stored in the streaming media cache **200** has an associated lookup key. Generally, a lookup key is used to determine whether the requested data can be obtained directly from the streaming media cache **200** or needs to be retrieved from the origin server **110**. In one embodiment of the present invention, the streaming media cache **200** determines the version of the client media player prior to processing the request for streaming data, and, if the version of an associated media player is identified as an impermissible version, a failure is reported to the requesting client **120**. For example, an RTSP access request from a client is denied unless the request is associated with a Windows® Media player, version 9.0 or above.

[**0031**] In one embodiment of the present invention, the keying and the lookup for the media file headers is driven by the Uniform Resource Locator (URL) specified by the client. For example, a URL associated with the client request may be of the following format: mms:/servername/filename. The "mms" portion of the URL identifies the request as an MMS request. The streaming media cache **200** utilizes an MMS session file header lookup key (e.g., NetCache:mms//servername:1755/filename/header) to determine if it stores the requested MMS session file header. If a URL associated with the client request specifies the RTP format (e.g., rtsp://servername/filename) by the "rtsp" portion of the URL, the streaming media cache **200** utilizes an RTSP session file header lookup key (e.g., Cache: rtsp//servername:554/filename/header) to determine if it stores the requested RTSP session file header. If the RTSP session file header lookup returns a miss, then the streaming media cache **200** obtains the Session Description Protocol (SDP) from the origin server **110** and writes the necessary information to disk **270**. Once the SDP information has been obtained and written to disk **270**, the streaming media cache **200** utilizes MMS lookup keys to determine if it stores the requested RTSP stream object and data. If the requested RTSP stream object and data cannot be located at the streaming media cache **200**, the requested data stream is obtained from the origin server **110** utilizing RTP.

[**0032**] In one embodiment of the present invention, the keying and the lookup for the data is MMS based. Specifically, although data that was previously cached utilizing RTSP may already be stored by the cache **200** in the RTP format, all newly cached data is stored by the streaming media cache **200** only in the MMS format. When a user requests a data stream in the RTP format and the requested data is already stored by the streaming media cache **200** in the MMS format, the requested data may be located by the cache **200** utilizing an MMS lookup key. The MMS lookup key is generated by converting the RTSP lookup key associated with the request into an MMS lookup key. The data, once located, is converted into the RTP format (e.g., by manipulating the packet headers) and served to the client in the RTP format.

[**0033**] In general, when a unified caching format is not used, the streaming media cache **200** stores content on disk with an MMS file header if the content has ever been streamed by a media player using an mms:// URL, and with an RTSP file header if the content has ever been streamed by a media player using an rtsp:// URL. An object having an associated MMS file header may be located with an MMS lookup key (e.g., NetCache:mms://server_name:1755/filename/header). An object having an associated RTSP file header may be located with an RTSP lookup key (e.g., NetCache:rtsp://server_name:554/filename/header).

[**0034**] **FIG. 2** is a computer system that may be used to embody the illustrative streaming media cache **200**. The streaming media cache may be any type of special-purpose computer (e.g., server) or general-purpose computer, including a standalone computer. The streaming media cache **200** comprises a network adapter **220**, a processor **230**, a memory controller **240**, a memory **250** and a storage adapter **260** interconnected by a system bus **210**.

[**0035**] The memory **250** comprises storage locations that are addressable by the processor and adapters for storing software program code. The memory comprises a form of random access memory (RAM) that is generally cleared by a power cycle or other reboot operation (e.g., it is a "volatile" memory). Those skilled in the art will appreciate that other memory means, such as FLASH memory media, may also be used for storing the program instructions and data structures shown in the memory **250**. The streaming media cache's processor **230** and adapters **220** and **260** can address data and instructions stored in the memory through the memory controller **240**. More specifically, the controller enables addressed portions of the memory's contents to be communicated over the bus **210**.

[**0036**] Portions of a storage operating system **300** are typically resident in the memory **250**. The term "storage operating system" as used herein generally refers to computer-executable code operable on a computer system that implements file system semantics and manages data access. The storage operating system **300** may be implemented as a microkernel, as well as an application program operating over a general-purpose operating system (e.g., a general-purpose operating system with configurable functionality).

[**0037**] In addition, a portion of the memory **250** may be organized as a "buffer cache"**255** for storing memory buffer ("mbuf") data structures that are passed between layers of the storage operating system **300** during normal runtime operation. The streaming media cache's processor and