CLAIMS

1. A method comprising:

receiving a first request to preconfigure a server for serving a stitched video;

responsive to the first request, identifying a sequence of video segments to include in the stitched video;

extracting, from video files corresponding to the video segments, video header blocks for the video segments;

generating a merged video header block by merging the video header blocks together based on the sequence and on the lengths of video data blocks within the video files;

storing the merged video header block as a header file;

generating a Uniform Resource Locator ("URL") that includes or references information that identifies: each of the video files, the header file, the lengths, and offsets within the video files at which the video data blocks are found;

responding to the first request with the URL;

receiving, from a client device, one or more second requests directed to the URL;

responsive to the one or more second requests, and based on the information included in or referenced by the URL: locating the video files and the header file based on the URL received in the one or more second requests;

streaming the stitched video to the client device by serving the header file followed by content from each of the video data blocks, in turn, from their respective video files.

2. The method of claim 1, wherein the URL directly identifies each of the video files, the header file, the lengths, and the offsets, wherein the URL further includes checksum data, the method further comprising: verifying that the URL has not been tampered with based on the checksum data.

3. The method of claim 1, further comprising:

generating the merged video header block and the URL only after determining that the merged video header block and the URL are not already stored in one or more storage locations; storing the merged video header block and the URL in the one or more storage locations.

4. The method of claim 1, further comprising:

receiving the first request from a web portal;

wherein the web portal is configured to generate a web page that embeds the URL and to send the web page to the client device; receiving the one or more second requests responsive to the client device accessing the web page.

5. The method of claim 1, further comprising receiving the one or more second requests via a proxy server.

6. The method of claim 1, further comprising skipping streaming of certain contents in certain video data blocks of the video data blocks responsive to byte range requests from the client device, the byte range requests specifying byte addresses relative to the sequence, the streaming comprising translating the sequence-relative byte addresses to byte addresses within the video files based on the lengths and the offsets.

7. A system comprising:

a video repository that stores video segments;

a sequence selector configured to identify sequences of the video segments;

a sequence metadata generator configured to read video header blocks of video segments that the sequence selector selected to be in a sequence, and to merge the video header blocks together to form a combined video header for the sequence;

a link generator configured to generate links for the sequences, each link including or referencing sequence information that identifies at least a set of video segments selected to be in a corresponding sequence for the link;

a link server configured to send links generated by the link generator in response to initial requests for links to sequences;

a video server configured to receive requests to send contents of the sequences to clients, and to respond to the requests by sending, for each requested sequence, the combined video header generated for the sequence followed by video data blocks of the video segments that the sequence selector selected to be in the sequence, in an order defined by the sequence, the requests received by the video server specifying particular links generated by the link generator, the video sever configured

to use the sequence information included in or referenced by a given link to locate a set of video files to use when sending the contents of a given sequence requested by the given link.

8. The system of claim 7:

wherein the sequence information in each link generated by the link generator further identifies at least a location in which the sequence metadata generator has stored the combined video header for the sequence; wherein the video server is further configured to use the sequence information included in or referenced by a given link to locate a given combined video header of a given sequence requested by the given link.

9. The system of claim 7, further comprising:

a cache configured to store links generated by the link generator, wherein the link server is configured to search the cache for a pre-existing link for a particular sequence prior to requesting that the link generator generate a particular link for the particular sequence.

10. The system of claim 7, further comprising:

a proxy server that rewrites or redirects requests for shortened links to links generated by the link generator, the links generated by the link generator including the corresponding sequence information.

11. The system of claim 7, further comprising:

one or more web portals configured to send the initial requests to the link server and embed the links received from the link generator within web pages served to the client; client devices configured to send requests for the links over a network to the video server in response to reading the links embedded within the web pages.

12. The system of claim 7, wherein the video server is configured to respond to byte-range requests to skip contents in the sequences.

13. The system of claim 7, wherein the sequence selector is configured to dynamically select one or more of the video segments included in certain sequences.

14. A method comprising:

identifying a sequence of video files to stitch together, each video file in the sequence having a video header block and a video data block;

based on the sequence and on the length of each video data block, merging each video header block together to form a merged video header block for the sequence;

receiving, from a client device, across a network, one or more requests to stream the sequence of video files;

responsive to the request, streaming the merged video header block to the client device, followed by content from each video data block, in an order defined by the sequence, the content from each video data block being streamed from its containing video file;

receiving, prior to the one or more requests to stream the sequence, an initial request; in response to the initial request: determining the video files to include in the sequence, the determining including selecting at least one of the video files based on information in the initial request;

generating a link to the sequence, the link including one or more identifiers associated with the sequence;

returning the link in response to the initial request;

wherein identifying the sequence is responsive to receiving the link in the one or more requests to stream the sequence, the video files being identified based on the one or more identifiers in the link.

15. The method of claim 14, further comprising streaming the sequence of video files as a single video stream to the client device without storing the single video stream as a distinct file at a server system that performs the method.

16. The method of claim 14, wherein determining the video files to include in the sequence comprises dynamically selecting at least one of the video files to include in the sequence based on information in the initial request.

17. The method of claim 14, wherein the one or more identifiers include file names of the video files, and the link includes integrity verification data.

18. The method of claim 14, further comprising:

in response to the initial request, generating stitching metadata, the stitching metadata including segment lengths that indicate byte sizes of each video data block, and segment offsets that indicate offsets within the video files at which each video data block begins; wherein the link includes or references the stitching metadata.

19. The method of claim 14, wherein merging each video header block comprises merging different types of sections found within each video header block, the different types of sections including chunk offset information and sample information, the merging further comprising adjusting addresses within the sections based on lengths of each video data block and the sequence of the video files.

20. The method of claim 14, further comprising skipping streaming of certain contents in certain video data blocks of the video data blocks responsive to byte range requests from the client device, the byte range requests specifying byte addresses relative to the sequence, the streaming comprising translating the sequence-relative byte addresses to byte addresses within the video files.

21. The method of claim 14, further comprising generating the merged video header block prior to the one or more requests to stream the sequence, the link specifying a location at which the merged video header block is stored.

22. One or more non-transitory computer-readable media storing instructions that, when executed by one or more computing devices, cause:

receiving a first request to preconfigure a server for serving a stitched video;

responsive to the first request, identifying a sequence of video segments to include in the stitched video;

extracting, from video files corresponding to the video segments, video header blocks for the video segments;

generating a merged video header block by merging the video header blocks together based on the sequence and on the lengths of video data blocks within the video files;

storing the merged video header block as a header file;

generating a Uniform Resource Locator ("URL") that includes or references information that identifies: each of the video files, the header file, the lengths, and offsets within the video files at which the video data blocks are found;

responding to the first request with the URL;

receiving, from a client device, one or more second requests directed to the URL;

responsive to the one or more second requests, and based on the information included in or referenced by the URL: locating the video files and the header file based on the URL received in the one or more second requests; streaming the stitched video to the client device by serving the header file followed by content from each of the video data blocks, in turn, from their respective video files.

23. The one or more non-transitory computer-readable media of claim 22, wherein the URL directly identifies each of the video files, the header file, the lengths, and the offsets, wherein the URL further includes checksum data, the method further comprising: verifying that the URL has not been tampered with based on the checksum data.

24. The one or more non-transitory computer-readable media of claim 22, wherein the instructions, when executed by the one or more computing devices, further cause: generating the merged video header block and the URL only after determining that the merged video header block and the URL are not already stored in one or more storage locations; storing the merged video header block and the URL in the one or more storage locations.

25. The one or more non-transitory computer-readable media of claim 22, wherein the instructions, when executed by the one or more computing devices, further cause: receiving the first request from a web portal;

wherein the web portal is configured to generate a web page that embeds the URL and to send the web page to the client device;

receiving the one or more second requests responsive to the client device accessing the web page.

26. The one or more non-transitory computer-readable media of claim 22, wherein the instructions, when executed by the one or more computing devices, further cause receiving the one or more second requests via a proxy server.

27. The one or more non-transitory computer-readable media of claim 22, wherein the instructions, when executed by the one or more computing devices, further cause skipping streaming of certain contents in certain video data blocks of the video data blocks responsive to byte range requests from the client device, the byte range requests specifying byte addresses relative to the sequence, the streaming comprising translating the sequence-relative byte addresses to byte addresses

within the video files based on the lengths and the offsets.

28. One or more non-transitory computer-readable media storing instructions that, when executed by one or more computing devices, cause: identifying a sequence of video files to stitch together, each video file in the sequence having a video header block and a video data block;

based on the sequence and on the length of each video data block, merging each video header block together to form a merged video header block for the sequence;

receiving, from a client device, across a network, one or more requests to stream the sequence of video files;

responsive to the request, streaming the merged video header block to the client device, followed by content from each video data block, in an order defined by the sequence, the content from each video data block being streamed from its containing video file;

receiving, prior to the one or more requests to stream the sequence, an initial request; in response to the initial request: determining the video files to include in the sequence, the determining including selecting at least one of the video files based on information in the initial request;

generating a link to the sequence, the link including one or more identifiers associated with the sequence; returning the link in response to the initial request;

wherein identifying the sequence is responsive to receiving the link in the one or more requests to stream the sequence, the video files being identified based on the one or more identifiers in the link.

29. The one or more non-transitory computer-readable media of claim 28, wherein the instructions, when executed by the one or more computing devices, further cause streaming the sequence of video files as a single video stream to the client device without storing the single video stream as a distinct file at a server system that executes the instructions.

30. The one or more non-transitory computer-readable media of claim 28, wherein determining the video files to include in the sequence comprises dynamically selecting at least one of the video files to include in the sequence based on information in the initial request.

31. The one or more non-transitory computer-readable media of claim 28, wherein the one or more identifiers include file names of the video files, and the link includes integrity verification data.

32. The one or more non-transitory computer-readable media of claim 28, wherein the instructions, when executed by the one or more computing devices, further cause:

in response to the initial request, generating stitching metadata, the stitching metadata including segment lengths that indicate byte sizes of each video data block, and segment offsets that indicate offsets within the video files at which each video data block begins; wherein the link includes or references the stitching metadata.

33. The one or more non-transitory computer-readable media of claim 28, wherein merging each video header block comprises merging different types of sections found within each video header block, the different types of sections including chunk offset information and sample information, the merging further comprising adjusting addresses within the sections based on lengths of each video data block and the sequence of the video files.

34. The one or more non-transitory computer-readable media of claim 28, wherein the instructions, when executed by the one or more computing devices, further cause:

skipping streaming of certain contents in certain video data blocks of the video data blocks responsive to byte range requests from the client device, the byte range requests specifying byte addresses relative to the sequence, the streaming comprising translating the sequence-relative byte addresses to byte addresses within the video files.

35. The one or more non-transitory computer-readable media of claim 28, wherein the instructions, when executed by the one or more computing devices, further cause generating the merged video header block prior to the one or more requests to stream the sequence, the link specifying a location at which the merged video header block is stored.