

Description

DATA TRANSMISSION SYSTEM HAVING IN-NETWORK STORAGE CAPABILITY

Related Application

This application claims priority to Chinese Patent Application No. 202011473017.0, filed on December 15, 2020, and entitled "DATA TRANSMISSION SYSTEM HAVING IN-NETWORK STORAGE CAPABILITY", which is incorporated herein by reference in its entirety.

Technical Field

The present invention relates to computer network technologies, and specifically to a data transmission system having an in-network storage capability.

Background

With the rapid development of internet of things and mobile internet, the existing IP network technology architecture gradually reveals inherent disadvantages in mobility, scalability, and data distribution efficiency, in which IP address is used to identify a host as well as its network location.. Information centric network (ICN) technology, which is characterized by the separation of identity and address, implements a content proximity service based on in-network caching, and provides an efficient and flexible data acquisition mode for a new 5G mobile network to cope with huge data processing in eMBB, URLLC, and mMTC scenarios. In ICN networks, intermediate transport nodes may also be used as response nodes for content requests. Therefore, transmission systems based on conventional end-to-end transport layer protocols cannot be applied to ICN networks.

Named data network (NDN) or content centric network (CCN), as a type of ICN network, performs hop-by-hop routing and addressing based on names, and is a revolutionary architecture in which addressing and routing are coupled, making it difficult to deploy and evolve in existing networks. In another manner of decoupling addressing and routing, for example, MobilityFirst, a network entity and a data name are mapped to one ID, and the ID is bound to a network address (NA) to which the ID is attached, with the ID used for addressing and the NA used for routing, thereby achieving compatibility with existing network

architectures such as IP networks and implementability in existing networks. Therefore, it is an urgent problem to provide a data transmission system having an in-network storage capability by making full use of storage and computational capabilities of intermediate transport nodes of a network under the premise of compatibility with an existing network routing mechanism.

Summary

An objective of the present invention is to provide, in view of the foregoing problems, a data transmission system having an in-network storage capability. The data transmission system includes a data request node, a data source service node, and intermediate transport nodes, each node is assigned a whole-network unique ID, is at least assigned one NA, and has a module for processing ID-based transport layer protocol messages, and a transmission object of the data transmission system is a data chunk identified by the whole-network unique ID;

the data request node is configured to generate and send a data request message and receive a data response message;

the data source service node is configured to respond to the data request, and when a data chunk identified by the ID is stored in the node, the node provides a sending service for the data chunk, and generates and sends a data response message; and

the intermediate transport nodes are configured to process ID-based transport layer protocol messages, and are provided with a self-decision making data storage function, and the nodes may decide subsequent processing behavior by information acquired from a message and a network status of the node, and may further respond to a data request, and when a data chunk identified by the ID is stored in the node, the nodes provide a sending service for the data chunk, and generate and send a data response message.

As an improvement to the foregoing system, a header field of the ID-based transport layer protocol messages includes, but not limited to, a data chunk ID, a data request node ID, a source address, a destination address, a message type, and a decision processing indication field; the data chunk ID and the data request node ID remain unchanged in a transmission process, the message type includes, but not limited to, a data request message and a data response message, and the decision processing indication field includes, but not limited to, a preference indication field, a cache indication field, a QoS indication field, and a combination thereof; and corresponding indication information included in the decision processing indication field is not

limited to being transferred through a message field, and may be transferred through a message payload.

As an improvement to the foregoing system, in the data request message, the source address is the NA of a data request node corresponding to the data request node ID, the destination address is the address of a node that stores a data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, and the destination address changes during transmission according to changes of the node that provides a sending service and an NA of the node.

As an improvement to the foregoing system, in the data response message, the source address is an address of a node that stores a data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, the destination address is the address of a data request node corresponding to the data request node ID, and the source address changes during transmission according to changes of the node that provides a sending service and an NA of the node.

As an improvement to the foregoing system, a value of the preference indication field represents indication information of an optimal network position for caching or storing a data chunk, and the optimal network position includes, but not limited to, a position close to a data request node, a position close to a data source, a position close to a network core, and a combination thereof;

a value of the cache indication field represents indication information of caching behavior of the data chunk in the message on the transmission path, and the caching behavior includes, but not limited to, a quantity of times that caching has been performed and a quantity of times that caching still needs to be performed for the data chunk on the transmission path; and

a value of the QoS indication field represents indication information of QoS requirements during transmission, caching, and storage of a data chunk, and the QoS requirements include, but not limited to, latency, a throughput, and reliability.

As an improvement to the foregoing system, each of the intermediate transport nodes has a decision module, an input of the decision module is a transmission processing information set and a network status set of the node, and the output of the decision module is configured to decide whether to perform a corresponding postprocessing action.

As an improvement to the foregoing system, the transmission processing information set is formed by information acquired by resolving messages by the intermediate transport nodes, and includes, but not limited to, message header field information and indication information transferred in a message payload; the network status set includes the status of the intermediate transport node itself and the status of its adjacent node; the status of the intermediate transport node is used for indicating a resource provision capability of a current intermediate transport node, and the resource provision capability includes, but not limited to, an available cache space, an available storage space, and an available bandwidth; the status of the adjacent nodes is used for indicating related status information of those intermediate transport nodes adjacent to the current intermediate transport node, and the related status information includes, but not limited to, the NAs and their available storage space of its adjacent intermediate transport nodes and the network distance between the adjacent intermediate transport nodes and the current node; and the adjacent another intermediate transport node is not limited to one intermediate transport node, and the status of the adjacent another intermediate transport node is obtained in an active detection manner or a passive detection manner.

As an improvement to the foregoing system, the decision module may be represented by calculation of a group of decision functions, one group of decision functions corresponds to one group of postprocessing actions, the value of a decision function is used to decide whether to perform a corresponding postprocessing action, and the functional form is different for each decision function; and one decision function corresponds to one or more postprocessing actions; and

assuming that there are N postprocessing actions, for the case that one decision function corresponds to one postprocessing action, an expression of a decision function $\mathcal{D}_n(X)$ of an n^{th} postprocessing action is as follows:

$$\mathcal{D}_n(X) = d_n, \text{ where } n = 1, 2, \dots, N,$$

where an input of the function is $X \in (PV \cup NV)$, PV is the transmission processing information set, NV is the network status set, n is a mark number of a postprocessing action, and the value d_n of the decision function $\mathcal{D}_n(X)$ is used for deciding whether to perform the postprocessing action with the mark number of n ; and

for the case that one decision function corresponds to a plurality of postprocessing actions,

the postprocessing actions are divided into M subgroups, $1 \leq M < N$, one decision function corresponds to one subgroup, and each subgroup includes k_l postprocessing actions, that is:

$$\sum_{l=1}^M k_l = N, \text{ where } 1 \leq k_l \leq N,$$

the expression of the decision function $\mathcal{D}_l(X)$ is as follows:

$$\mathcal{D}_l(X) = [d_{l1}, d_{l2}, \dots, d_{lk_l}], \text{ where } l = 1, 2, \dots, M,$$

where d_{l1} , d_{l2} , and d_{lk_l} are output values of an l^{th} decision function $\mathcal{D}_l(X)$, and each value respectively corresponds to one postprocessing action in an l^{th} subgroup.

As an improvement to the foregoing system, the postprocessing action specifies forwarding processing behavior of an intermediate transport node on a received transport layer protocol message, and the forwarding processing behavior includes, but not limited to, direct forwarding, caching, storage, adjacent forwarding, and modification forwarding.

As an improvement to the foregoing system, an action of the adjacent forwarding includes: modifying the destination address in the message with the address of an adjacent node and performing forwarding, where at the same time the original message may be forwarded, or the message is forwarded after related indication information in the message is modified; the adjacent node is another intermediate transport node selected according to a distance to a current node; and the distance includes, but not limited to, a network distance or a geographical distance.

As an improvement to the foregoing system, the address of the adjacent node is filled according to an output value of an adjacent node selection function $SN(Y)$, and an expression of the function is:

$$SN(Y) = \{NA_i, i = 1, \dots, I\},$$

where the function selects an appropriate adjacent node from other intermediate transport nodes adjacent to a current intermediate transport node according to a status of an adjacent node, the functional form is not limited, the input Y of the function is the status of an adjacent node, the output value NA_i is an NA of an i^{th} adjacent node that satisfies the selection condition of the selection function, and I is the total quantity of adjacent nodes that satisfy the selection condition of the selection function.

As an improvement to the foregoing system, a name resolution system is configured outside the data transmission system, and is configured to manage and maintain mapping

relationships between the ID and the NA of each node in the data transmission system and between a data chunk ID and the NA of a node that stores the data chunk, and one data chunk ID may correspond to a plurality of NAs; in this way, a resolution service from an ID to an NA is provided; and the node that stores the data chunk is the data source service node or the intermediate transport node that stores the data chunk identified by the ID and that can provide a sending service of the data chunk.

As an improvement to the foregoing system, the modification forwarding is modifying the destination address in the message and performing forwarding, the corresponding address modified in the modification forwarding is filled through an output value of a selection function $G(CNA)$, and an expression of the selection function is:

$$G(CNA) = \{NA_j, NA_j \in CNA, j = 1, \dots, J\},$$

where CNA is an NA set of nodes that store the data chunks identified by the ID, the set may be acquired through, but not limited to, querying from the name resolution system, and one or more functions $G(CNA)$ may be defined based on application and network requirements; and an output value NA_j is an NA of a j^{th} node that stores the data chunk identified by the ID, and J is the total quantity of nodes that store a data chunk identified by the ID.

The advantages of the present invention are as follows.

Through the system of the present invention, transmission of a data chunk is implemented between a data request node and an intermediate transport node or a data source service node based on an unchanged data chunk ID and data request node ID, and caching and storage of a data chunk in a transmission process are implemented through decision making and processing of a message field by an intermediate transport node.

Brief Description of Drawings

FIG. 1 is a schematic diagram of composition of a data transmission system having an in-network storage capability according to the present invention;

FIG. 2 is a schematic diagram of a message with a message field including decision processing indication information;

FIG. 3 is a schematic diagram of a message with a message payload including decision processing indication information;

FIG. 4 is a diagram of functional modules of an intermediate transport node;

FIG. 5 is a schematic diagram A of a data request and response procedure; and

FIG. 6 is a schematic diagram B of a data request and response procedure.

Detailed Description of Embodiments

The technical solutions of the present invention are described below in detail with reference to the accompanying drawings.

As shown in FIG. 1, a data transmission system having an in-network storage capability provided by the present invention is formed by nodes such as a data request node, a data source service node, and intermediate transport nodes that can process transport layer messages and that are provided with a self-decision making data storage function. Each node is assigned a whole-network unique ID, and is at least assigned one NA. A transmission object of the system is a data chunk identified by the whole-network unique ID (ID-c shown in the figure). The intermediate transport nodes and the data source service node in the system can all respond to a request initiated by the data request node. Every node in the system includes one module for processing ID-based transport layer protocol messages.

The data request node in the system is provided with a function of generating and sending a data request message and receiving a data response message. The data source service node in the system is provided with a function of responding to a data request message and generating and sending a data response message.

A name resolution system outside the system is responsible for managing and maintaining mapping relationships between the ID of each node and the NA of the node in the data transmission system and between a data chunk ID and the NA of a node that stores the data chunk, to provide a resolution service from an ID to an NA. The mapping relationships may be that one data chunk ID corresponds to a plurality of NAs. For example, in the figure, one data chunk ID-c corresponds to two NAs NA-s and NA-t3.

A header field of a transport layer protocol message provided in the present invention may include, but not limited to, a data chunk ID, a data request node ID, a source address, a destination address, a message type, and a decision processing indication field. FIG. 2 is a schematic diagram of the message in an IPv6 network.

The data chunk ID and the data request node ID remain unchanged in a transmission process. The message type includes, but not limited to, a data request message and a data

response message. The decision processing indication field includes, but not limited to, a preference indication, a cache indication, and a QoS indication, which may be combined. In the data request message, the source address is the NA of a data request node corresponding to the data request node ID, the destination address is the address of a node that stores a data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, and the destination address may change during transmission according to changes of the node that provides a sending service and an NA of the node.

In the data response message, the source address is an address of a node that stores a data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, the destination address is the address of a data request node corresponding to the data request node ID, and the source address may change during transmission according to changes of the node that provides a sending service and an NA of the node.

In the decision processing indication field, a value of the preference indication field represents indication information of a preferred network position for caching or storing a data chunk, and the Preferably network position includes, but not limited to, a position close to a data request node, a position close to a data source, a position close to a network core, a combination thereof, and the like. A value of the cache indication field represents indication information of caching behavior of the data chunk in the message on the transmission path, and the caching behavior includes, but not limited to, a quantity of times that caching has been performed, a quantity of times that caching still needs to be performed for the data chunk on the transmission path, and the like. A value of the QoS indication field represents indication information of QoS requirements during transmission, caching, and storage of a data chunk, and the QoS requirements include, but not limited to, requirements such as latency, a throughput, and reliability.

Decision processing indication information in a message field may be transferred through a message payload. FIG. 3 is a schematic diagram of a message for transferring the information through a message payload. A Tag, Length, and Value (TLV) encoding format is used for the message. Tags 1, 2, and 3 respectively correspond to a QoS indication, a preference indication, and a cache indication.

A diagram of functional modules of an intermediate transport node is shown in FIG. 4.

The modules include a transport layer protocol message processing module, a decision module, and a postprocessing module.

The transport layer protocol message processing module is responsible for receiving and identifying an ID-based transport layer protocol message, resolving the message, and acquiring information such as a data chunk ID, a data request node ID, a source address, a destination address, a message type, and a decision processing indication included in a header field or payload of the transport layer protocol message, and simultaneously constructing a transmission processing information set based on this and transferring the transmission processing information set to the decision module. The transport layer protocol message processing module is also responsible for transferring a message processed by the postprocessing module to a network layer to send out the message.

An input of the decision module is the transmission processing information set and a network status set of a current node that are transferred by the transport layer protocol message processing module. An output result of the decision module is transferred to the postprocessing module, and is used for deciding whether to perform a corresponding postprocessing action.

The decision module may be represented by calculation of a group of decision functions, and one group of decision functions corresponds to one group of postprocessing actions. The value of a decision function is used to decide whether to perform a corresponding postprocessing action. The functional form may be different for each decision function. One decision function may correspond to one postprocessing action, or may correspond to a plurality of postprocessing actions. Without loss of generality, assuming that there are N postprocessing actions ($N \geq 1$), for the case that one decision function corresponds to one postprocessing action, an expression of a decision function $\mathcal{D}_n(X)$ of an n^{th} postprocessing action is as follows:

$$\mathcal{D}_n(X) = d_n, \text{ where } n = 1, 2, \dots, N,$$

where an input of the function is $X \in (PV \cup NV)$, PV is the transmission processing information set, NV is the network status set, n is a mark number of a postprocessing action, and the value d_n of the decision function $\mathcal{D}_n(X)$ is used for deciding whether to perform the postprocessing action with the mark number of n .

For the case that one decision function corresponds to a plurality of postprocessing actions, the postprocessing actions are divided into M subgroups ($1 \leq M < N$), one decision function

corresponds to one subgroup, and each subgroup includes k_l postprocessing actions, that is:

$$\sum_{l=1}^M k_l = N, \text{ where } 1 \leq k_l \leq N,$$

the expression of the decision function $\mathcal{D}_l(X)$ is as follows:

$$\mathcal{D}_l(X) = [d_{l1}, d_{l2}, \dots, d_{lk_l}], \text{ where } l = 1, 2, \dots, M,$$

where d_{l1} , d_{l2} , and d_{lk_l} are output values of an l^{th} decision function $\mathcal{D}_l(X)$, and each value respectively corresponds to one postprocessing action in an l^{th} subgroup.

The postprocessing module determines, based on the output of the decision module, whether to perform postprocessing actions such as direct forwarding, caching, storage, adjacent forwarding, and modification forwarding.

In the direct forwarding, a message is directly forwarded without modification.

In the caching, a data chunk in a message is stored in an intermediate transport node locally, and at the same time the original message is forwarded, or the message is forwarded after related indication information in the message is modified.

In the storage, a data chunk in a message is stored in an intermediate transport node locally.

In the adjacent forwarding, a message is forwarded after the destination address in the message is modified with the address of an adjacent node, and at the same time, the original message may be forwarded, or the message is forwarded after related indication information in the message is modified. The distance between the current node and the adjacent node may be, but not limited to, a network distance or a geographical distance. The distance between the current node and the adjacent node may be, but not limited to, a network distance or a geographical distance. In the adjacent forwarding, the modified destination address may be filled according to an output value of an adjacent node selection function $SN(Y)$:

$$SN(Y) = \{NA_i, i = 1, \dots, I\}.$$

The function selects another appropriate transport node from transport nodes adjacent to a current intermediate transport node according to a status of an adjacent node, the functional form is not limited, the input Y of the function is the status of an adjacent node, the output value NA_i is an NA of an adjacent intermediate transport node that satisfies the selection condition of the selection function.

The modification forwarding is modifying the destination address in the message and performing forwarding, the corresponding address modified in the modification forwarding is

filled through an output value of a selection function $G(CNA)$, and an expression of the selection function is:

$$G(CNA) = \{NA_j, NA_j \in CNA, j = 1, \dots, J\},$$

CNA is an NA set of nodes that correspond to a data chunk ID and that store data chunks identified by the ID or an NA set of data request nodes corresponding to a data request node ID, the set may be acquired through, but not limited to, querying from the name resolution system, and one or more functions $G(CNA)$ may be defined based on application and network requirements.

In the decision function $\mathcal{D}_n(X)$, in a two-step decision making manner, determination is first performed once on some relatively independent decision items such as a data chunk caching/storage requirement, a caching/storage capability of a current node, and a selection and caching capability of an adjacent node. Joint decision making is then performed based on a result of determination. That is, the decision function may be represented as a joint decision function of output values of determination functions of a series of decision items:

$$\mathcal{D}_n(X) = \mathcal{JD}_n[\mathcal{F}_1(X), \mathcal{F}_2(X), \dots, \mathcal{F}_i(X), \dots],$$

where $\mathcal{JD}_n(\cdot)$ is a joint decision function for an n^{th} postprocessing action. An input of the function is output values of determination functions $\mathcal{F}_1(X), \mathcal{F}_2(X), \dots, \mathcal{F}_i(X), \dots$, triggering a related decision item of the n^{th} postprocessing action, and forms of $\mathcal{JD}_n(\cdot)$ and $\mathcal{F}_i(X)$ are not limited.

A determination function of the related decision item may be represented as:

$\mathcal{F}_C(X)$ represents a cache determination function, used to determine whether a data chunk in a message needs to be cached in the current intermediate transport node. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_C(X) = \begin{cases} Y, & \text{the data chunk needs to be cached in the current node} \\ N, & \text{the data chunk does not need to be cached in the current node} \end{cases}$$

$\mathcal{F}_{CA}(X)$ represents a cache capability determination function, used to determine whether the current intermediate transport node is capable of caching a data chunk. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_{CA}(X) = \begin{cases} Y, & \text{the current node is capable of caching a data chunk} \\ N, & \text{the current node is not capable of caching a data chunk} \end{cases}$$

$\mathcal{F}_{CN}(X)$ represents an adjacency cache determination function, used to determine whether

the current intermediate transport node can forward a message to an adjacent intermediate transport node for caching. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_{CN}(X) = \begin{cases} Y, & \text{the node can forward a message to an adjacent intermediate transport node for caching} \\ N, & \text{the node cannot forward a message to an adjacent intermediate transport node for caching} \end{cases}$$

$\mathcal{F}_S(X)$ represents a storage determination function, used to determine whether a data chunk in a message needs to be stored in the current intermediate transport node. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_S(X) = \begin{cases} Y, & \text{the data chunk needs to be stored in the current node} \\ N, & \text{the data chunk does not need to be stored in the current node} \end{cases}$$

$\mathcal{F}_{SA}(X)$ represents a storage capability determination function, used to determine whether the current intermediate transport node is capable of storing a data chunk. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_{SA}(X) = \begin{cases} Y, & \text{the current node is capable of storing a data chunk} \\ N, & \text{the current node is not capable of storing a data chunk} \end{cases}$$

$\mathcal{F}_{MD}(X)$ represents a destination address determination function, used to determine whether to modify a destination address. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_{MD}(X) = \begin{cases} Y, & \text{it is determined to modify a destination address} \\ N, & \text{it is determined not to modify a destination address} \end{cases}$$

$\mathcal{F}_{MS}(X)$ represents a source address determination function, used to determine whether to modify a source address. An output value of the function may be represented by, but not limited to, a Boolean value:

$$\mathcal{F}_{MS}(X) = \begin{cases} Y, & \text{it is determined to modify a source address} \\ N, & \text{it is determined not to modify a source address} \end{cases}$$

$\mathcal{F}_i(X)$ is another determination function involved according to an application or network requirement.

In this way, the joint decision function of postprocessing actions such as direct forwarding, caching, storage, adjacent forwarding, and modification forwarding may be:

(1) For a direct forwarding action

$$\mathcal{JD}_1(X) = [(\mathcal{F}_C(X) == N) \cap (\mathcal{F}_S(X) == N)] \cup [(\mathcal{F}_C(X) == Y) \cap (\mathcal{F}_{CA}(X) == N) \cap$$

$(\mathcal{F}_{CN}(X) == N)]$.

(2) For a cache action

$$\mathcal{JD}_2(X) = (\mathcal{F}_C(X) == Y) \cap (\mathcal{F}_{CA}(X) == Y).$$

(3) For a storage action

$$\mathcal{JD}_3(X) = (\mathcal{F}_C(X) == Y) \cap (\mathcal{F}_{CA}(X) == N) \cap (\mathcal{F}_{CN}(X) == Y).$$

(4) For an adjacent forwarding action

$$\mathcal{JD}_4(X) = [(\mathcal{F}_C(X) == Y) \cap (\mathcal{F}_{CA}(X) == N) \cap (\mathcal{F}_{CN}(X) == Y)] \cup [(\mathcal{F}_S(X) == Y) \cap (\mathcal{F}_{SA}(X) == N)].$$

(5) For a modification forwarding action

$$\mathcal{JD}_5(X) = (\mathcal{F}_{MD}(X) == Y) \cup (\mathcal{F}_{MS}(X) == Y).$$

When an output value of the joint decision function is "true", a corresponding postprocessing action is performed. Outputs of the function may be continuous values. Through comparison with a threshold, it is decided whether to perform a corresponding postprocessing action.

A specific implementation of a data request and response procedure of the transmission system provided in the present invention is shown in FIG. 5 and FIG. 6.

For FIG. 5, without loss of generality, it is assumed that no intermediate transport node has cached or stored any data chunk during initialization. A data source service node is assigned an NA NA-s, stores a data chunk with an ID ID-c, and has registered a mapping relationship between ID-c and an address NA-s of the data source service node in a name resolution system. A data request node 1 is assigned a whole-network unique ID ID-r1, and is assigned an NA NA-r1. Intermediate transport nodes 1, 2, and 3 are respectively assigned NAs NA-t1, NA-t2, and NA-t3. A two-step decision-making manner is used for decision functions in decision modules of the nodes. It is assumed that cache determination functions $\mathcal{F}_C(X)$ are all $\mathcal{F}_C(X) = Y$. That is, a current node always needs to perform caching. In a header field of a transport layer protocol message, values of a message type field are set to 0 and 1. 0 represents a data request message, and 1 represents a data response message.

(101) The data request node 1 sends a resolution request to the name resolution system. The request carries the data chunk ID-c. The name resolution system returns the address NA-s of the data source service node that stores the data chunk ID-c.

(102) A transport layer protocol message processing module of the data request node 1 generates and sends a data request message and initiates a request for a data chunk. A message type is set to 0, a data chunk ID is set to ID-c, a data request node ID is set to ID-r1, a destination address is set to NA-s, and a source address is set to NA-r1. The message is forwarded to the intermediate transport node 1 according to a routing mechanism of a network to which the message belongs.

(103) A transport layer protocol message processing module of the intermediate transport node 1 receives the data request message, identifies and resolves the message, reads the message type of 0 indicating that the message is a data request message, then reads the destination address, where the destination address of the message is not an address of a current node, and directly forwards the message, where the message is forwarded to the intermediate transport node 2 according to the routing mechanism of the network to which the message belongs.

(104) A transport layer protocol message processing module of the intermediate transport node 2 receives the data request message, identifies and resolves the message, reads the message type of 0 indicating that the message is a data request message, then reads the destination address, where the destination address of the message is not an address of a current node, and directly forwards the message, where the message is forwarded to the data source service node according to the routing mechanism of the network to which the message belongs.

(105) A transport layer protocol message processing module of the data source service node receives the data request message, identifies and resolves the message, reads the message type of 0 indicating that the message is a data request message, then reads the destination address, where the destination address NA-s of the message is an NA of the current node, therefore generates a data response message in response to the data request, where a message type is set to 1, a data chunk ID is set to ID-c, a data request node ID is set to ID-r1, a destination address is set to NA-r1, a source address is set to NA-s, decision processing indication field information at least includes a cache indication, then encapsulates a data chunk corresponding to ID-c, and sends the message, where the message is forwarded to the intermediate transport node 3 according to the routing mechanism of the network to which the message belongs.

(106) A transport layer protocol message processing module of the intermediate transport node 3 receives the data response message, identifies and resolves the message, reads the

message type of 1 indicating that the message is a data response message, reads decision processing indication information, and transfers the decision processing indication information to a decision module. The decision module calculates $\mathcal{F}_C(X)$ and $\mathcal{F}_{CA}(X)$. $\mathcal{F}_C(X) = Y$ needs to be cached. In addition, because the current node has not cached and stored any data chunk and has a caching capability. $\mathcal{F}_{CA}(X) = Y$. Based on this, a joint decision function $\mathcal{JD}_2(X) = Y$ for a caching action is calculated. That is, a decision result is to perform a caching process. Therefore, a postprocessing module performs the caching action, and caches the data chunk. After the caching of the data chunk succeeds, step (107) is performed, and at the same time an original message is forwarded, or the message may be forwarded after a cache indication information in the message is modified. The message is forwarded to the intermediate transport node 1 according to the routing mechanism of the network to which the message belongs.

(107) The intermediate transport node 3 registers the mapping relationship between a data chunk ID and an NA NA-t3 of the intermediate transport node 3 with the name resolution system.

(108) The transport layer protocol message processing module of the intermediate transport node 1 receives the data response message, identifies and resolves the message, reads the message type of 1 indicating that the message is a data response message, reads decision processing indication information, and transfers the decision processing indication information to a decision module. The decision module calculates $\mathcal{F}_C(X)$ and $\mathcal{F}_{CA}(X)$. $\mathcal{F}_C(X) = Y$ needs to be cached. In addition, because the current node has not cached and stored any data chunk and has a caching capability. $\mathcal{F}_{CA}(X) = Y$. Based on this, a joint decision function $\mathcal{JD}_2(X) = Y$ for a caching action is calculated. That is, a decision result is to perform a caching process. Therefore, the postprocessing module performs the caching action, and caches the data chunk. After the caching of the data chunk succeeds, step (109) is performed, and at the same time an original message is forwarded, or the message may be forwarded after a cache indication information in the message is modified. The message is forwarded to the data request node 1 according to the routing mechanism of the network to which the message belongs.

(109) The intermediate transport node 1 registers the mapping relationship between a data chunk ID and an NA NA-t1 of the intermediate transport node 1 with the name resolution system.

(110) The transport layer protocol message processing module of the data request node 1 receives a data response message, identifies and resolves the message, reads the message type of 1 indicating that the message is a data response message, reads the destination address, where the destination address NA-r1 of the message is the NA of the current node. Therefore, the data chunk is read, and the data chunk that has been read is provided to an upper-layer application.

In the process of performing the foregoing step (105), step (105a) may be first performed to update the destination address, and then the message is sent.

(105a) The data source service node sends a resolution request to the name resolution system. The request carries the ID ID-r1 of the data request node 1. The name resolution system returns the address NA-r1 corresponding to ID-r1, and updates the destination address.

In the process of performing the foregoing step (106), step (106a) may be first performed to update the destination address, and then the message is forwarded.

(106a) The intermediate transport node 3 sends a resolution request to the name resolution system. The request carries the ID ID-r1 of the data request node 1. The name resolution system returns the address NA-r1 corresponding to ID-r1, and updates the destination address.

In the process of performing the foregoing step (108), the intermediate transport node 1 may first update the destination address, and then forward the message. The step is similar to (106a). Details are not described again.

Starting from step (105) in the foregoing procedure, the data response message is sequentially transmitted to the intermediate transport nodes 3 and 1 and the data request node 1 according to the routing mechanism of the network to which the message belongs. According to the routing mechanism of the network, the message may alternatively be transmitted along a path of the intermediate transport nodes 2 and 1 and the data request node 1.

In step (105) of the foregoing procedure, when the transport layer protocol message processing module of the data source service node encapsulates a data response message, a size of a data chunk may be larger than a maximum length of a network layer message. A data chunk corresponding to ID-c may be divided into smaller data slices, which are encapsulated as a plurality of data packets for sending. The message may be sent in, but not limited to, a continuous manner, or according to requesting of a data request node.

For FIG. 6, based on FIG. 5, a data request node 2 is added, is assigned a whole-network

unique ID ID-r2, and is assigned an NA NA-r2. In addition, it is assumed that the data request node 1 has already requested and obtained a data chunk with an ID ID-c according to the procedure in FIG. 5, and the data chunk has been successfully cached and registered in the intermediate transport nodes 3 and 1. A specific implementation procedure in which the data request node 2 requests a data chunk with the ID ID-c is as follows:

(201) The data request node 2 sends a resolution request to the name resolution system. The request carries the data chunk ID-c. The name resolution system returns an NA set CNA storing the data chunk ID-c. The NA set CNA includes the address NA-s of the data source service node and the addresses NA-r3 and NA-r1 of the intermediate transport nodes 3 and 1. It is assumed that NA-r3 is selected. The name resolution system may alternatively select one address and return the address. A method of selection is not limited. It is assumed that NA-r3 is returned.

(202) A transport layer protocol message processing module of the data request node 2 generates and sends a data request message and initiates a request for a data chunk. A message type is set to 0, a data chunk ID is set to ID-c, a data request node ID is set to ID-r2, a source address is set to NA-r1, and a destination address is set to NA-r3. The message is forwarded to the intermediate transport node 1 according to a routing mechanism of a network to which the message belongs.

(203) The transport layer protocol message processing module of the intermediate transport node 1 receives the data request message, identifies and resolves the message, reads the message type of 0 indicating that the message is a data request message, then reads the destination address, where the destination address of the message is not an address of a current node, and directly forwards the message, where the message is forwarded to the intermediate transport node 3 according to the routing mechanism of the network to which the message belongs.

(204) The transport layer protocol message processing module of the intermediate transport node 3 receives the data request message, identifies and resolves the message, reads the message type of 0 indicating that the message is a data request message, then reads the destination address, where the destination address of the message is an address of the current node, therefore generates a data response message in response to the data request, where a

message type is set to 1, a data chunk ID is set to ID-c, a data request node ID is set to ID-r2, a destination address is set to NA-r2, a source address is set to NA-t3, and decision processing indication field information at least includes a cache indication, then encapsulates a data chunk corresponding to ID-c, and sends the message, where the message is forwarded to the intermediate transport node 1 according to the routing mechanism of the network to which the message belongs.

(205) The transport layer protocol message processing module of the intermediate transport node 1 receives the data response message, identifies and resolves the message, reads the message type of 1 indicating that the message is a data response message, reads decision processing indication information, and transfers the decision processing indication information to a decision module. The decision module calculates $\mathcal{F}_C(X)$ and $\mathcal{F}_{CA}(X)$. $\mathcal{F}_C(X) = Y$ and needs to be cached. It is assumed that the current node currently has a caching capability. $\mathcal{F}_{CA}(X) = Y$. Based on this, a joint decision function $\mathcal{JD}_2(X) = Y$ for a caching action is calculated. That is, a decision result is to perform a caching process. Therefore, the postprocessing module performs the caching action, and caches the data chunk. The node has cached the data chunk, and therefore does not need to perform caching again. At the same time, an original message is forwarded, or the message may be forwarded after a cache indication information in the message is modified. The message is forwarded to the data request node 2 according to the routing mechanism of the network to which the message belongs.

(206) The transport layer protocol message processing module of the data request node 2 receives a data response message, identifies and resolves the message, reads the message type of 1 indicating that the message is a data response message, and reads the destination address, where the destination address NA-r2 of the message is the NA of the current node. Therefore, the data chunk is read, and the data chunk that has been read is provided to an upper-layer application.

In the process of performing the foregoing steps (204) and (205), the destination address may be first updated, and then the message is forwarded. The step is similar to (106a). Details are not described again.

Finally, it should be noted that the foregoing embodiments are merely intended for describing the technical solutions of the present invention rather than limiting the present

invention. Although the present invention is described in detail with reference to the embodiments, persons of ordinary skill in the art should understand that they may still make modifications or equivalent replacements to the technical features of the present invention without departing from the spirit and scope of the technical solutions of the technical solutions of the present invention. These modifications or equivalent replacements shall all fall within the scope of the claims the present invention.

Claims

1. A data transmission system having an in-network storage capability, wherein the data transmission system comprises a data request node, a data source service node, and intermediate transport nodes, each node is assigned a whole-network unique identifier (ID), is at least assigned one network address (NA), and has a module for processing ID-based transport layer protocol messages, and a transmission object of the data transmission system is a data chunk identified by the whole-network unique ID;

the data request node is configured to generate and send a data request message and receive a data response message;

the data source service node is configured to respond to the data request, and when a data chunk identified by the ID is stored in the node, the node provides a sending service for the data chunk, and generates and sends a data response message; and

the intermediate transport nodes are configured to process ID-based transport layer protocol messages, and are provided with a self-decision making data storage function, and the nodes may decide subsequent processing behavior by information acquired from a message and a network status of the node, and may further respond to a data request, and when a data chunk identified by the ID is stored in the node, the nodes provide a sending service for data chunk, and generate and send a data response message.

2. The data transmission system having an in-network storage capability according to claim 1, wherein a header field of the ID-based transport layer protocol messages comprises, but not limited to, a data chunk ID, a data request node ID, a source address, a destination address, a message type, and a decision processing indication field; the data chunk ID and the data request node ID remain unchanged in a transmission process, the message type comprises, but not limited to, a data request message and a data response message, and the decision processing indication field comprises, but not limited to, a preference indication field, a cache indication field, a QoS indication field, and a combination thereof; and corresponding indication information comprised in the decision processing indication field is not limited to being transferred through a message field, and may be transferred through a message payload.

3. The data transmission system having an in-network storage capability according to

claim 2, wherein in the data request message, the source address is the NA of a data request node corresponding to the data request node ID, the destination address is the address of a node that stores a data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, and the destination address changes during transmission according to changes of the node that provides a sending service and an NA of the node.

4. The data transmission system having an in-network storage capability according to claim 2, wherein in the data response message, the source address is an address of a node that stores the data chunk identified by the data chunk ID and that can provide a sending service of the data chunk, the destination address is the address of a data request node identified by the data request node ID, and the source address changes during transmission according to changes of the node that provides a sending service and an NA of the node.

5. The data transmission system having an in-network storage capability according to claim 2, wherein a value of the preference indication field represents indication information of an optimal network position for caching or storing a data chunk, and the optimal network position comprises, but not limited to, a position close to a data request node, a position close to a data source, a position close to a network core, and a combination thereof;

a value of the cache indication field represents indication information of caching behavior of the data chunk in the message on the transmission path, and the caching behavior comprises, but not limited to, a quantity of times that caching has been performed and a quantity of times that caching still needs to be performed for the data chunk on the transmission path; and

a value of the QoS indication field represents indication information of QoS requirements during transmission, caching, and storage of a data chunk, and the QoS requirements comprise, but not limited to, latency, throughput, and reliability.

6. The data transmission system having an in-network storage capability according to claim 1, wherein each of the intermediate transport nodes has a decision module, an input of the decision module is a transmission processing information set and a network status set of the node, and the output of the decision module is configured to decide whether to perform a corresponding postprocessing action.

7. The data transmission system having an in-network storage capability according to claim 6, wherein the transmission processing information set is formed by information acquired

by resolving messages by the intermediate transport nodes, and comprises, but not limited to, the message header field information and indication information transferred in a message payload; the network status set comprises the status of the intermediate transport node itself and the status of its adjacent nodes; the status of the intermediate transport node is used for indicating a resource provision capability of a current intermediate transport node, and the resource provision capability comprises, but not limited to, an available cache space, an available storage space, and an available bandwidth; the status of the adjacent nodes is used for indicating related status information of those intermediate transport nodes adjacent to the current intermediate transport node, and the related status information comprises, but not limited to, the NAs and their available storage space of its adjacent intermediate transport nodes and the network distance between the adjacent intermediate transport nodes and the current node; and the adjacent another intermediate transport node is not limited to one intermediate transport node, and the status of the adjacent another intermediate transport node is obtained in an active detection manner or a passive detection manner.

8. The data transmission system having an in-network storage capability according to claim 6 or 7, wherein the decision module may be represented by calculation of a group of decision functions, one group of decision functions corresponds to one group of postprocessing actions, the value of a decision function is used to decide whether to perform a corresponding postprocessing action, and the functional form is different for each decision function; and one decision function corresponds to one or more postprocessing actions; and

assuming that there are N postprocessing actions, for the case that one decision function corresponds to one postprocessing action, an expression of a decision function $\mathcal{D}_n(X)$ of an n^{th} postprocessing action is as follows:

$$\mathcal{D}_n(X) = d_n, \text{ wherein } n = 1, 2, \dots, N,$$

wherein an input of the function is $X \in (PV \cup NV)$, PV is the transmission processing information set, NV is the network status set, n is a mark number of a postprocessing action, and the value d_n of the decision function $\mathcal{D}_n(X)$ is used for deciding whether to perform the postprocessing action with the mark number of n ; and

for the case that one decision function corresponds to a plurality of postprocessing actions, the postprocessing actions are divided into M subgroups, $1 \leq M < N$, one decision function

corresponds to one subgroup, and each subgroup comprises k_l postprocessing actions, that is:

$$\sum_{l=1}^M k_l = N, \text{ wherein } 1 \leq k_l \leq N,$$

the expression of the decision function $\mathcal{D}_l(X)$ is as follows:

$$\mathcal{D}_l(X) = [d_{l1}, d_{l2}, \dots, d_{lk_l}], \text{ wherein } l = 1, 2, \dots, M,$$

wherein d_{l1} , d_{l2} , and d_{lk_l} are output values of an l^{th} decision function $\mathcal{D}_l(X)$, and each value respectively corresponds to one postprocessing action in an l^{th} subgroup.

9. The data transmission system having an in-network storage capability according to claim 6, wherein the postprocessing action specifies forwarding processing behavior of an intermediate transport node on a received transport layer protocol message, and the forwarding processing behavior comprises, but not limited to, direct forwarding, caching, storage, adjacent forwarding, and modification forwarding.

10. The data transmission system having an in-network storage capability according to claim 9, wherein an action of the adjacent forwarding comprises: modifying the destination address in the message with the address of an adjacent node and performing forwarding, wherein at the same time the original message may be forwarded, or the message is forwarded after related indication information in the message is modified; the adjacent node is another intermediate transport node selected according to a distance to a current node; and the distance comprises, but not limited to, a network distance or a geographical distance.

11. The data transmission system having an in-network storage capability according to claim 10, wherein the address of the adjacent node is filled according to an output value of an adjacent node selection function $SN(Y)$, and an expression of the function is:

$$SN(Y) = \{NA_i, i = 1, \dots, I\},$$

wherein the function selects an appropriate adjacent node from other intermediate transport nodes adjacent to the current intermediate transport node according to a status of an adjacent node, the functional form is not limited, the input Y of the function is the status of an adjacent node, the output value NA_i is an NA of an i^{th} adjacent node that satisfies the selection condition of the selection function, and I is the total quantity of adjacent nodes that satisfy the selection condition of the selection function.

12. The data transmission system having an in-network storage capability according to claim 9, wherein a name resolution system is configured outside the data transmission system,

and is configured to manage and maintain mapping relationships between the ID and the NA of each node in the data transmission system and between a data chunk ID and the NA of a node that stores the data chunk, and one data chunk ID may correspond to a plurality of NAs; in this way, a resolution service from an ID to an NA is provided; and the node that stores the data chunk is the data source service node or the intermediate transport node that stores the data chunk identified by the ID and that can provide a sending service of the data chunk.

13. The data transmission system having an in-network storage capability according to claim 12, wherein the modification forwarding is modifying the destination address in the message and performing forwarding, the corresponding address modified in the modification forwarding is filled through an output value of a selection function $G(CNA)$, and an expression of the selection function is:

$$G(CNA) = \{NA_j, NA_j \in CNA, j = 1, \dots, J\},$$

wherein CNA is an NA set of nodes that store the data chunk identified by the ID, the set may be acquired through, but not limited to, querying from the name resolution system, and one or more functions $G(CNA)$ may be defined based on application and network requirements; and an output value NA_j is an NA of a j^{th} node that stores the data chunk identified by the ID, and J is the total quantity of nodes that store the data chunk identified by the ID.