

CLAIMS

1. A computer-implemented method of providing application message delivery, characterized in that it comprises the following steps performed with at least one data processor:
 - 5 at a plurality of log servers (203 a - b) coupled to at least an application server (201 a - c): receiving asynchronously, from the at least one application server (201 a - c), application messages containing application information, each application message being received by at least some log servers (203 a - b) among the plurality of log servers (203 a - b); receiving asynchronously, from
10 the at least one application server (201 a - c), control messages at a predetermined interval, each control message being received by at least some log servers (203 a - b) among the plurality of log servers (203 a - b);
at each of the plurality of log servers (203 a - b): storing the received application messages in a current application data file (209); storing the
15 received control messages in a control file (211) and upon receiving an open-close control message, closing the current application data file, storing said closed application data file and creating a new application data file as the current application data file;
comparing the control files of the plurality of log servers (203 a - b) for a
20 given interval; and
based on this comparison, determining from among a plurality of application data files (209) from each of the log servers (203 a - b), an application data file as a best-candidate for a given interval and forwarding the best-candidate file for post-processing.
25
2. The method according to claim 1, wherein each log server (203 a - b) is coupled to a plurality of application servers (201 a - c) associated each to at least an application.
- 30 3. The method according to any one of claims 1 to 2, wherein the control message comprises a number (309) of application messages transmitted by the application server (201 a-c) and a control message number (307) identifying each control message in a sequence of control messages.

4. The method according to any one of claims 1 to 3, wherein the control message comprises at least one of: an identifier (303) of an application and a timestamp (305) of the application server.
5. The method according to any one of the preceding claims, wherein the interval for forwarding a control message is a given time period.
6. The method according to any one of the preceding claims, wherein the open/close control message is a Nth control message in a sequence of control messages, N being predetermined.
7. The method according to the preceding claim, wherein the Nth control message is the fifth control message in the sequence of control messages and the given time period is a two minute time period.
8. The method according to any one of the preceding claims, wherein the best-candidate file is chosen from a set of application data files from the plurality of log servers (203 a - b) and that have the same start and stop points.
9. The method according to the preceding claim, wherein the best-candidate file is chosen from among the chosen set of files, the application data file with the lowest application message loss rate.
10. The method according to the preceding claim, wherein in case some application data files have the same number of application messages, then the best-candidate file is chosen from among the application data files with the lowest application message loss rate, the application data file with the lowest control message loss rate.

11. The method according to any one of the two preceding claims, wherein the best-candidate file having lost application messages and not having lost more than x percent of application messages for the interval is augmented by the lost application messages existing in other files of the set of files, x being predetermined.
12. The method according to any one of the preceding claims, wherein the application is an airline billing transaction application.
13. A non-transitory computer-readable medium that contains software program instructions, where execution of the software program instructions by at least one data processor results in performance of operations that comprise execution of the method as in any one of the preceding claims.
14. A system for high reliability and high performance application message delivery, characterized in that it comprises:
- a plurality of log servers (203 a -b) coupled to the output of at least one application server (201 a - c);
 - each log server (203 a -b) being configured to receive asynchronously, from the at least one application server (201 a - c): application messages containing application information and control messages;
 - each log server (203 a - b) being also configured to: store the received application messages in a current application data file (209) and to store the received control messages in a control file (211); and upon receiving an open-close control message, to close the current application data file, to store said closed application data file and to create a new application data file as the current application data file;
 - a server coupled to the plurality of log servers (203 a -b), said server being configured to: compare the control files of the plurality of log servers (203 a - b) for a given interval; based on this comparison, determine from among a plurality of application data files (209) from each of the log servers (203 a - b), an application data file as a best-candidate for a given interval; and forward the best-candidate file for post-processing.

15. The system according to claim 14, wherein the system comprises a plurality of application servers (201 a - c) and a plurality of applications executing on a processor of any of the plurality of application servers (201 a - c), each of
5 the application servers (201 a - c) having an output coupled to an input of each of the log servers (203 a -b).

10

CLAIMS

1. A computer-implemented method of providing application message delivery, characterized in that it comprises the following steps performed with at least one
5 data processor:

at a plurality of log servers coupled to at least an application server : receiving asynchronously, from the at least one application server, application messages containing application information, each application message being received by at least some log servers among the plurality of log servers;
10 receiving asynchronously, from the at least one application server, control messages at a predetermined interval, wherein the control messages comprise at least one of: an identifier of an application and a timestamp of the application server, each control message being received by at least some log servers among the plurality of log servers;

15 at each of the plurality of log servers: storing the received application messages in a current application data file; storing the received control messages in a control file and upon receiving an open-close control message, closing the current application data file, storing said closed application data file and creating a new application data file as the current application data file;
20 comparing the control files of the plurality of log servers for a given interval; and

based on this comparison, determining from among a plurality of application data files from each of the log servers, an application data file as a best-candidate for a given interval and forwarding the best-candidate file for post-
25 processing to a network entity.

2. The method according to claim 1, wherein each log server is coupled to a plurality of application servers associated each to at least an application.

30 3. The method according to any one of claims 1 to 2, wherein the control message comprises a number of application messages transmitted by the

application server and a control message number identifying each control message in a sequence of control messages.

4. The method according to any one of claims 1 to 3, wherein the network entity is a billing framework.

5. The method according to any one of the preceding claims, wherein the interval for forwarding a control message is a given time period.

6. The method according to any one of the preceding claims, wherein the open/close control message is a Nth control message in a sequence of control messages, N being predetermined.

7. The method according to the preceding claim, wherein the Nth control message is the fifth control message in the sequence of control messages and the given time period is a two minute time period.

8. The method according to any one of the preceding claims, wherein the best-candidate file is chosen from a set of application data files from the plurality of log servers and that have the same start and stop points.

9. The method according to the preceding claim, wherein the best-candidate file is chosen from among the chosen set of files, the application data file with the lowest application message loss rate.

10. The method according to the preceding claim, wherein in case some application data files have the same number of application messages, then the best-candidate file is chosen from among the application data files with the lowest application message loss rate, the application data file with the lowest control message loss rate.

11. The method according to any one of the two preceding claims, wherein the best-candidate file having lost application messages and not having lost more than x percent of application messages for the interval is augmented by the lost application messages existing in other files of the set of files, x being predetermined.

5

12. The method according to any one of the preceding claims, wherein the application is an airline billing transaction application.

13. A non-transitory computer-readable medium that contains software program instructions, where execution of the software program instructions by at least one data processor results in performance of operations that comprise execution of the method as in any one of the preceding claims.

14. A system for high reliability and high performance application message delivery, characterized in that it comprises:

a plurality of log servers coupled to the output of at least one application server;

each log server being configured to receive asynchronously, from the at least one application server: control messages received at a predetermined interval and application messages containing application information wherein the control messages comprise at least one of: an identifier of an application and a timestamp of the application server;

each log server being also configured to: store the received application messages in a current application data file and to store the received control messages in a control file; and upon receiving an open-close control message, to close the current application data file, to store said closed application data file and to create a new application data file as the current application data file;

a server coupled to the plurality of log servers, said server being configured to: compare the control files of the plurality of log servers for a given interval; based on this comparison, determine from among a plurality of application data files from each of the log servers, an application data file as a best-

candidate for a given interval; and forward the best-candidate file for post-processing to a network entity.

15. The system according to claim 14, wherein the system comprises a plurality of
5 application servers and a plurality of applications executing on a processor of any of
the plurality of application servers, each of the application servers having an output
coupled to an input of each of the log servers.
16. The system according to any one of claims 14 or 15 wherein said server is
10 configured to forward the best-candidate file for post-processing to a billing
framework.