

CLAIMS:

1. A computer-implemented system for propagating timer management and event triggering, the system comprising:
 - a memory storing instructions; and
 - at least one processor configured to execute the instructions to:
 - receive, from a first system via a network, a request to instantiate a timer, the request comprising at least one unique identifier;
 - accessing a service level agreement in at least one database based on the at least one unique identifier;
 - determining, based on the service level agreement:
 - an end time for the timer, and
 - an event for triggering at the end time;
 - save the received request to a data store comprising a plurality of requests;
 - trigger the event upon reaching the end time for the timer associated with the request; and
 - responsive to triggering the event, sending, via a network, the event associated with the request.
2. The system of claim 1, wherein the data store comprising the plurality of requests comprises a key-value data structure.
3. The system of claim 2, wherein the received request is stored in the data store by storing the end time for the respective timer as a key and the event as an associated value.
4. The system of claim 3, wherein the instructions further cause the processor to:
 - maintain a repeating master timer with a pre-defined interval;
 - determining that the repeating master timer interval has expired; and
 - based on determining that the repeating master time interval has expired:

locating one or more keys in the data store;
comparing the time stored in the located keys to the current time;
triggering the associated event for at least one located key with an
associated time in the past.

5. The system of claim 2, wherein the instructions further cause the processor to:
periodically store snapshots of the key-value data store on a computer-readable medium;
determine that data loss has occurred in the key-value data store due to power outage, program error, or other incident;
reading the stored snapshots; and
restoring data to the key-value data store using the stored snapshots.
6. The system of claim 1, wherein the instructions further cause the processor to:
maintain a repeating master timer with a pre-defined interval;
determining that the repeating master timer interval has expired; and
based on determining that the repeating master time interval has expired:
consulting the data store to determine one or more requests whose
respective end times have been reached;
triggering the associated event for the one or more determined
requests.
7. The system of claim 1, wherein the event for triggering at the end time comprises
sending a notification to a destination system indicating that the timer has
elapsed, the notification including instructions for performance by
the destination system.
8. The system of claim 1, wherein the instructions further cause the processor to:
receive, from the first system, a request to cancel the timer; and
clear the received request from the data store.

9. The system of claim 1, wherein sending the event comprises sending the event to the first system.
10. A computer implemented method for propagating timer management and event triggering, the method comprising:
 - receiving, from a first system via a network, a request to instantiate a timer, the request comprising at least one unique identifier;
 - accessing a service level agreement in at least one database based on the at least one unique identifier;
 - determining, based on the service level agreement:
 - an end time for the timer, and
 - an event for triggering at the end time;
 - saving the received request to a data store comprising a plurality of requests;
 - triggering the event upon reaching the end time for the timer associated with the request; and
 - responsive to triggering the event, sending, via a network, the event associated with the request.
11. The method of claim 10, wherein the data store comprising the plurality of requests comprises a key-value data structure.
12. The method of claim 11, wherein the received request is stored in the data store by storing the end time for the respective timer as a key and the event as an associated value.
13. The method of claim 12, further comprising:
 - maintaining a repeating master timer with a pre-defined interval;
 - determining that the repeating master timer interval has expired; and
 - based on determining that the repeating master time interval has expired:
 - locating one or more keys in the data store;

comparing the time stored in the located keys to the current time;
triggering the associated event for at least one located key with an
associated time in the past.

14. The method of claim 11, further comprising:

periodically store snapshots of the key-value data store on a computer-
readable medium;
determining that data loss has occurred in the key-value data store due to
power outage, program error, or other incident;
reading the stored snapshots; and
restoring data to the key-value data store using the stored snapshots.

15. The method of claim 10, further comprising:

maintain a repeating master timer with a pre-defined interval;
determining that the repeating master timer interval has expired; and
based on determining that the repeating master time interval has expired:
consulting the data store to determine one or more requests whose
respective end times have been reached;
triggering the associated event for the one or more determined
requests.

16. The method of claim 10, further comprising:

sending a notification to a destination system indicating that the timer has
elapsed, the notification including instructions for performance by
the destination system.

17. The method of claim 10, further comprising:

receive, from the first system, a request to cancel the timer; and
clear the received request from the data store.

18. The method of claim 10, wherein sending the event comprises sending the event to the first system.
19. A computer-implemented system for propagating timer management and event triggering, the system comprising:
- a memory storing instructions; and
 - at least one processor configured to execute the instructions to:
 - receiving, from a first system via a network, a request to instantiate a timer, the request comprising at least one unique identifier;
 - accessing a service level agreement in at least one database based on the at least one unique identifier;
 - determining, based on the service level agreement:
 - an end time for the timer, and
 - an event for triggering at the end time;
 - saving the received request to a data store comprising a plurality of requests;
 - triggering the event upon reaching the end time for the timer associated with the request;
 - responsive to triggering the event, sending, via a network, the event associated with the request;
 - maintain a repeating master timer with a pre-defined interval;
 - determining that the repeating master timer interval has expired;
 - and
 - based on determining that the repeating master time interval has expired:
 - locating one or more keys in the data store;
 - comparing the time stored in the located keys to the current time;
 - triggering the associated event for at least one located key with an associated time in the past.