

What is claimed is:

1. A method for processing multiple kinds of event messages in a computing system, the method including:

storing received event messages in a data storage system, the data storage system including storage for a first plurality of records associated with event messages, where each event message includes a timestamp, and records associated with event messages include a field indicating a target delivery time for an event result for the event message, the target delivery time for at least some event messages being determined according to a kind of the event message; and

processing the stored event messages to deliver event results based on information in the stored event messages and the target delivery times in the records associated with the event messages, the processing including:

prioritizing event messages to deliver event results according to information stored in the event messages indicating priority,

computing a target delivery time for each event message in a first set of one or more event messages that store information indicating a same priority for all of the event messages in the first set based on a fixed delay relative to its respective timestamp, and

delivering event results for event messages that are deliverable based on a comparison of their target delivery times to a clock time.

2. The method of claim 1 wherein the processing further includes reordering a second set of event messages containing fewer than all of the stored event messages to process event messages in the second set of event messages according to the timestamps in the event messages instead of processing event messages in the second set of event messages according to an order in which those event messages were received.

3. The method of claim 2 wherein the target delivery time associated with a particular event message is computed after the particular event message is received using a function that is defined at least in part by information included in the particular event message.

4. The method of claim 3 wherein, for event messages associated with a first event class, the function returns a target delivery time that is based on the clock time.

5. The method of claim 4 wherein event messages associated with the first event class are not in the second set that is reordered, and are processed according to the order in which those event messages were received.

6. The method of claim 4 wherein, for event messages associated with a second event class of lower priority than the first event class, the function returns a target delivery time that is a sum of the fixed delay and a timestamp in the event message.

7. The method of claim 6 wherein event messages associated with the second event class are in the second set that is reordered, and are reordered according to timestamps stored in the event messages associated with the second class.

8. The method of claim 6 wherein, for event messages associated with a third event class of lower priority than the second event class, the function returns a target delivery time that is based on a timestamp stored in the event message.

9. The method of claim 8 wherein event messages associated with the third event class are in the second set that is reordered, and are reordered according to timestamps stored in the event messages associated with the third class.

10. The method of claim 6 wherein, for event messages associated with a third event class of lower priority than the second event class, the function returns a target delivery time that is based on a repeating schedule.

11. The method of claim 2 wherein the second set includes event messages that all include a common key value.

12. The method of claim 1 wherein prioritizing event messages to deliver event results according to information stored in the event messages includes storing event messages associated with event classes of different priorities in different respective ordered queues, and selecting a next event message to evaluate for potential delivery of an event result from the front of an ordered queue of the highest priority that is not empty.

13. Software stored in a non-transitory form on a computer-readable medium, for processing event messages in a computing system, the software including instructions for causing the computing system to:

store received event messages in a data storage system, the data storage system including storage for a first plurality of records associated with event messages, where each event message includes a timestamp, and records associated with event messages include a field indicating a target delivery time for an event result for the event message, the target delivery time for at least some event messages being determined according to a kind of the event message; and

process the stored event messages to deliver event results based on information in the stored event messages and the target delivery times associated with the event messages, the processing including: prioritizing event messages to deliver event results according to information stored in the event messages indicating priority, computing a target delivery time for each event message in a first set of one or more event messages that store information indicating a same priority for all of the event messages in the first set based on a fixed delay relative to its respective timestamp, and delivering event results for event messages that are deliverable based on a comparison of their target delivery times to a clock time.

14. A computing system for processing event messages, the computing system including:

a data storage system for storing received event messages, the data storage system including storage for a first plurality of records associated with event messages, where each event message includes a timestamp, and records associated with event messages include a field indicating a target delivery time for an event result for the event message, the target delivery time for at least some event messages being determined according to a kind of the event message; and

at least one processor configured to process the stored event messages to deliver event results based on information in the stored event messages and the target delivery times associated with the event messages, the processing including:

prioritizing event messages to deliver event results according to information stored in the event messages indicating priority,

computing a target delivery time for each event message in a first set of one or more event messages that store information indicating a same priority for all of the event messages in the first set based on a fixed delay relative to its respective timestamp, and

delivering event results for event messages that are deliverable based on a comparison of their target delivery times to a clock time.

15. The software of claim 13 wherein the processing further includes reordering a second set of event messages containing fewer than all of the stored event messages to process event messages in the second set of event messages according to the timestamps in the event messages instead of processing event messages in the second set of event messages according to an order in which those event messages were received.

16. The software of claim 15 wherein the target delivery time associated with a particular event message is computed after the particular event message is received using a function that is defined at least in part by information included in the particular event message.

17. The software of claim 16 wherein, for event messages associated with a first event class, the function returns a target delivery time that is based on the clock time.

18. The software of claim 17 wherein event messages associated with the first event class are not in the second set that is reordered, and are processed according to the order in which those event messages were received.

19. The software of claim 17 wherein, for event messages associated with a second event class of lower priority than the first event class, the function returns a target delivery time that is a sum of the fixed delay and a timestamp in the event message.

20. The software of claim 15 wherein the second set includes event messages that all include a common key value.

21. The software of claim 13 wherein prioritizing event messages to deliver event results according to information stored in the event messages includes storing event messages associated with event classes of different priorities in different respective ordered queues, and selecting a next event message to evaluate for potential delivery of an event result from the front of an ordered queue of the highest priority that is not empty.

22. The computing system of claim 14 wherein the processing further includes reordering a second set of event messages containing fewer than all of the stored event messages to process event messages in the second set of event messages according to the timestamps in the event messages instead of processing event messages in the second set of event messages according to an order in which those event messages were received.

23. The computing system of claim 22 wherein the target delivery time associated with a particular event message is computed after the particular event message is received using a function that is defined at least in part by information included in the particular event message.

24. The computing system of claim 23 wherein, for event messages associated with a first event class, the function returns a target delivery time that is based on the clock time.

25. The computing system of claim 24 wherein event messages associated with the first event class are not in the second set that is reordered, and are processed according to the order in which those event messages were received.

26. The computing system of claim 24 wherein, for event messages associated with a second event class of lower priority than the first event class, the function returns a target delivery time that is a sum of the fixed delay and a timestamp in the event message.

27. The computing system of claim 22 wherein the second set includes event messages that all include a common key value.

28. The computing system of claim 14 wherein prioritizing event messages to deliver event results according to information stored in the event messages includes storing event messages associated with event classes of different priorities in different respective ordered queues, and selecting a next event message to evaluate for potential delivery of an event result from the front of an ordered queue of the highest priority that is not empty.