CLAIMS

1. A baseline monitoring method, comprising:

determining a plurality of tasks to be monitored in a baseline;

determining whether there is a first task to be monitored that has cyclic dependency according to dependencies among the plurality of tasks to be monitored, wherein the cyclic dependency indicates mutual dependency among the first tasks to be monitored;

determining a task instance to be checked for each of the tasks to be monitored, wherein in presence of the first task to be monitored that has cyclic dependency, for the first task to be monitored that is not a guaranteed task, determining the task instance to be checked for the first task to be monitored according to the cyclic dependency and dependency between the first task to be monitored and its downstream task to be monitored as the latest task instance which depends on the first task to be monitored and cyclic process between the first task to be monitored is no longer performed so as to avoid the cycle dependency from entering a process of continuous circulating, and for a second task to be monitored that is not the guaranteed task, determining the task instance to be checked for the second task to be monitored according to dependency between the second task to be monitored and its downstream task to be monitored as the latest task instance which depends on the second task to be monitored, wherein the second task to be monitored is a task to be monitored except the first task to be monitored, and the guaranteed task is a task added to the baseline by a user; and

for each of the tasks to be monitored, monitoring the baseline according to the task instance to be checked for the task to be monitored.

2. The baseline monitoring method of claim 1, wherein the determining the task instance to be checked for the first task to be monitored according to the cyclic dependency and dependency between the first task to be monitored and its downstream task to be monitored as the latest task instance which depends on the first task to be monitored includes:

determining a second task instance which is depended on for the first task to be monitored according to a first task instance for its upstream first task to be monitored on which the first task to be monitored depends, and offset information of a dependency offset between the upstream first task to be monitored when it is downstream of the first task to be monitored and the first

task to be monitored, wherein the dependency offset represents that a task to be monitored depends on historical execution situation of another task to be monitored; and

determining a latest task instance which is depended on for the first task to be monitored according to the second task instance and a task instance for the first task to be monitored on which its downstream task to be monitored depends, and determining the latest task instance as the task instance to be checked for the first task to be monitored.

3. The baseline monitoring method of claim 1, wherein the determining the task instance to be checked for the second task to be monitored according to dependency between the second task to be monitored and its downstream task to be monitored as the latest task instance which depends on the second task to be monitored includes:

determining a latest task instance for the second task to be monitored which is depended on according to offset information of a dependency offset between the second task to be monitored and its downstream task to be monitored, and taking the latest task instance as the task instance to be checked for the second task to be monitored.

4. The baseline monitoring method of claim 1, wherein the determining the task instance to be checked for each of the tasks to be monitored includes:

for the task to be monitored as a guaranteed task, determining the task instance to be checked for the task to be monitored according to a latest task instance for the task to be monitored.

5. The baseline monitoring method of claim 1, further comprising:

determining check time information corresponding to each of the tasks to be monitored according to a preset baseline committed time and a warning margin, as well as the dependencies among the plurality of tasks to be monitored;

the monitoring of the baseline according to the task instance to be checked for the task to be monitored includes:

checking a running state of the task instance to be checked for the task to be monitored at a time indicated by the check time information corresponding to the task to be monitored; and

determining a baseline state of the baseline according to a check result, so as to monitor the

baseline.

6. The baseline monitoring method of claim 5, wherein the check time information includes a latest warning start time, a latest committed start time, a task warning completion time, and a task committed completion time;

the determining the baseline state of the baseline according to the check result includes:

if the check result indicates that the task instance has started to run before the latest warning start time, and has run successfully before the task warning completion time, determining the baseline state to be a safe state;

if the check result indicates that the task instance has not started running at the latest warning start time, or the task instance has not successfully run at the task warning completion time, determining the baseline state to be a warning state; and

if the check result indicates that the task instance has not started running at the committed latest start time, or the task instance has not run successfully at the task committed completion time, determining the baseline state to be a broken state.

7. The baseline monitoring method of claim 1, further comprising:

for each of the tasks to be monitored, determining a real-time margin of the task to be monitored according to a current instance state of the task instance to be checked for the task to be monitored, wherein task instances in different instance states correspond to different ways of determining real-time margins of tasks to be monitored;

determining a real-time margin of the baseline according to a minimum value of the real-time margin for each task to be monitored; and

determining the baseline state of the baseline according to the real-time margin of the baseline.

8. The baseline monitoring method of claim 7, wherein the instance state includes a waiting execution state, an in-execution state and an execution failure state;

the determining the real-time margin of the task to be monitored according to the current instance state of the task instance to be checked for the task to be monitored includes:

in a case that the current instance state of the task instance is the waiting execution state,

determining the real-time margin of the task to be monitored according to a current time and the task committed completion time, an expected scheduling time, and an average running duration of the task to be monitored;

in a case that the current instance state of the task instance is the in-execution state, determining the real-time margin of the task to be monitored according to the current time and the task committed completion time, the actual start execution time, and the average running duration of the task to be monitored; and

in a case that the current instance state of the task instance is the execution failure state, determining the real-time margin of the task to be monitored according to the current time and the task committed completion time and the average running duration of the task to be monitored.

9. The baseline monitoring method of claim 1, further comprising:

in a case that an actual running duration of the task to be monitored is greater than a preset ratio of an average running duration of the task to be monitored, determining the task to be monitored to be a task running slowly, wherein different average running durations correspond to different preset ratios.

10. The baseline monitoring method of claim 8 or 9, wherein the average running duration is determined according to historical running durations except a maximum duration and a minimum duration of a plurality of historical running durations of the task to be monitored.

11. A baseline monitoring apparatus, comprising:

a determination module for tasks to be monitored configured to determine a plurality of tasks to be monitored in a baseline;

a cyclic dependency determination module configured to determine whether there is a first task to be monitored that has cyclic dependency according to dependencies among the plurality of tasks to be monitored, wherein the cyclic dependency indicates mutual dependency among the first tasks to be monitored;

a task instance determination module configured to determine a task instance to be checked for each of the tasks to be monitored, wherein in presence of the first task to be monitored that

has cyclic dependency, for the first task to be monitored that is not a guaranteed task, determine the task instance to be checked for the first task to be monitored according to the cyclic dependency and dependency between the first task to be monitored and its downstream task to be monitored as the latest task instance which depends on the first task to be monitored and cyclic process between the first task to be monitored is no longer performed so as to avoid the cycle dependency from entering a process of continuous circulating, and for a second task to be monitored that is not the guaranteed task, determining the task instance to be checked for the second task to be monitored according to dependency between the second task to be monitored and its downstream task to be monitored as the latest task instance which depends on the second task to be monitored, wherein the second task to be monitored is a task to be monitored except the first task to be monitored, and the guaranteed task is a task added by a user to the baseline; and

a monitoring module configured to, for each of the tasks to be monitored, monitor the baseline according to the task instance to be checked for the task to be monitored.

12. A computer-readable medium with a computer program stored thereon, which, when executed by a processing apparatus, implements steps of the baseline monitoring method of any of claims 1 to 10.

13. An electronic device comprising:

a storage apparatus with a computer program stored thereon; and

a processing apparatus configured to execute the computer program in the storage apparatus, to implement steps of the baseline monitoring method of any of claims 1 to 10.

14. A computer program comprising:

instructions, which, when executed by a processor, cause the processor to perform steps of the baseline monitoring method of any of claims 1 to 10.

15. A computer program product comprising instructions, which, when executed by a processor, cause the processor to perform steps of the baseline monitoring method of any of claims 1 to 10.