CLAIMS

1. A computer-implemented method, comprising:

receiving an animation generation instruction identifying key frames to use for generating an animation;

obtaining a plurality of tweens corresponding to the animation, wherein each of the tweens comprises a tween object between a start key frame and an end key frame;

generating one or more timelines when a quantity of tweens is greater than or equal to 2, wherein each timeline corresponds to one or more tweens, and a quantity of timelines is less than the quantity of tweens;

generating the animation using the plurality of tweens based on the one or more timelines, wherein generating the animation comprises:

when a given timeline of the one or more timelines corresponds to a plurality of tweens, determining whether there is an M-level conflicting tween group in the plurality of tweens, wherein M-level conflicting tweens in the M-level conflicting tween group have the same tween object and conflicting tween time, M is a natural number, and an initial value is a default count initial value;

when there is the M-level conflicting tween group, updating an M-level conflicting tween that corresponds to the latest end time in the M-level conflicting tween group with an (M+1)-level conflicting tween, and determining whether there is an (M+1)-level conflicting tween group in the (M+1)-level tween, until there is no (M+K)-level conflicting tween group in an (M+K)-level conflicting tween, wherein K is a natural number greater than or equal to 0; and

generating an animation for a corresponding non-conflicting tween and the (M+K)-level tween based on the timeline.

2. The computer-implemented method according to claim 1, wherein determining whether there is an *M*-level conflicting tween group in the plurality of tweens comprises:

obtaining a tween object of each tween; and

when two or more tweens have the same tween object, determining whether a tween time of any one of the two or more tweens conflicts with a tween time of another tween.

3. The computer-implemented method of claim 2, further comprising:

if the tween time of the any tween conflicts with the tween time of the another tween, determining that there is at least one *M*-level conflicting tween group in the plurality of tweens, wherein the *M*-level conflicting tween group comprises the any tween and the another tween.

4. The computer-implemented method according to claim 2, wherein the determining whether a tween time of any one of the two or more tweens conflicts with a tween time of another tween comprises:

determining whether there is an intersection between a tween time between a start key frame and an end key frame of the any tween and a tween time between a start key frame and an end key frame of the another tween, and if there is an intersection, determining that the tween time of the any tween conflicts with the tween time of the another tween.

5. The computer-implemented method according to claim 1, wherein updating the *M*-level conflicting tween that corresponds to the latest end time in the *M*-level conflicting tween group with an (*M*+1)-level conflicting tween comprises:

updating an *M*-level conflicting tween having the largest frame number of a tween end key frame in the *M*-level conflicting tween group with the (*M*+1)-level conflicting tween.

6. The computer-implemented method according to claim 1, wherein the M-level conflicting tweens in the M-level conflicting tween group is determined to have conflicting tween time based on an intersection between tween times between start key frames and end key frames of the conflicting tweens.

7. The computer-implemented method according to any one of claims 1 to 6, wherein generating the animation comprises:

using a single tween object from a plurality of tween objects having conflicting tween time.

8. An apparatus for generation of an animation, the apparatus comprising a plurality of modules configured to perform the method of any one of claims 1 to 7.