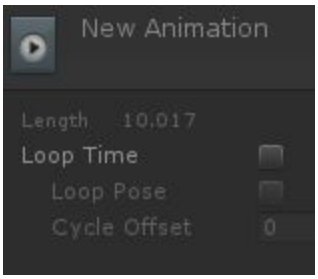
Animator Extended Uptime

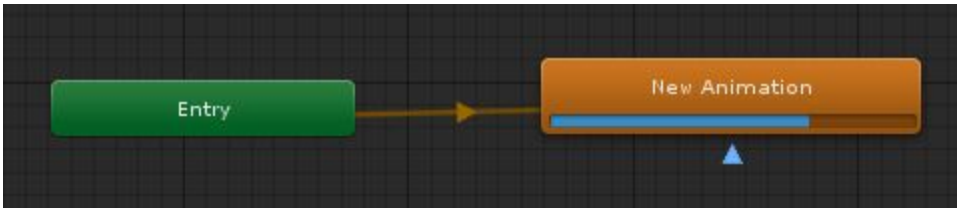
Unity animations will be stuttering or completely stuck after 4 days of uptime. Observed in both Unity Editor and Windows standalone player.

Reproduction steps:

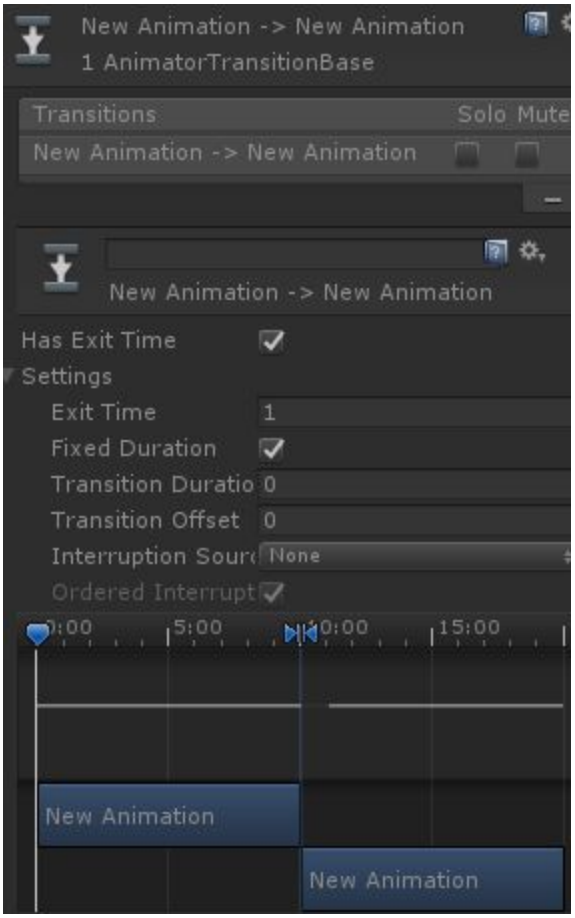
- Create new project with any version of Unity
- Create GameObject with Animator component
- Create AnimationController and assign to Animator
- Create looping Animation which smoothly moves the position of GameObject and assign to AnimationController state
- Play in Editor
- Change Time.timeScale to 100 (maximum), can be done in TimeManager inspector or code
- Wait approximately 1 hour unscaled time or more to simulate 4 days or more of extended uptime
- Restore Time.timeScale to 1
- Observe Animation which updates GameObject position is now stuttering or completely frozen. Changing GameObject selected in Editor Hierarchy window will force update the Animation frames, so position of GameObject will likely jump through frames.
- Disable and re-enable the GameObject with Animator. Observe the Animation is now smoothly updating the position like normal. This is due to the Animator reallocating its statemachine internals and starting the default state animation from beginning.

Suggested workaround:

- Disable “loop time” on Animation
  - 

- Create new transition within AnimationController which starts from the animation state and ends at the same animation state. Observe the triangle below the "New Animation" state which represents a self transition.
  - 

- Change the newly created self transition settings in inspector window.

- Enable "Has Exit Time" to auto transition when animation completes
- Remove transition time to avoid blending
    - Set "Exit Time" to 1
    - Set "Transition Duration" to 0
    - 

- Now you have effectively created a looping animation which resets the animation uptime values in native Unity code during the animation state self transition.