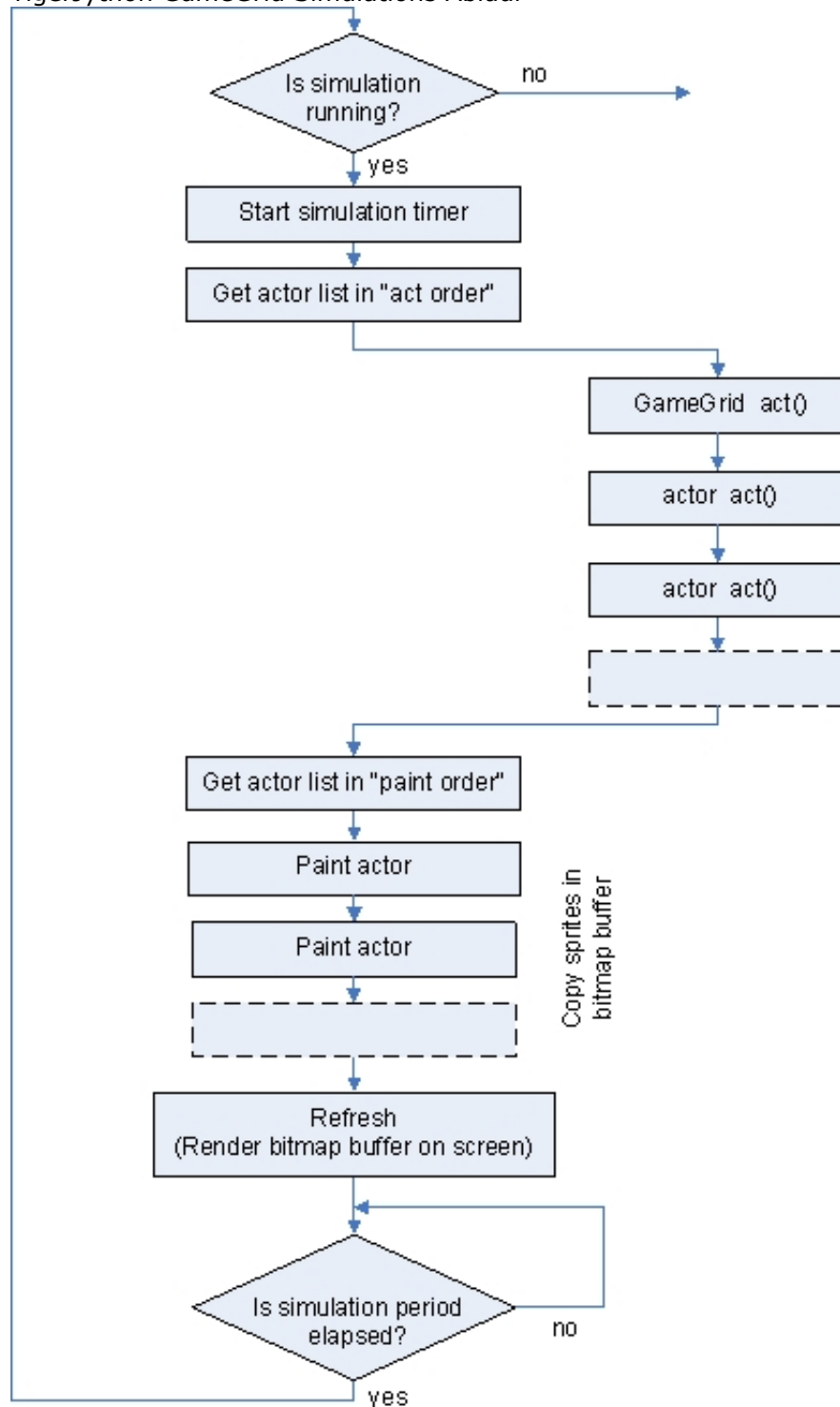


TigerJython GameGrid Simulations Ablauf

TigerJython GameGrid Simulations Ablauf



Actors sind teilweise aktive Objekte, die von demselben Thread animiert werden, der die Simulationsschleife ausführt. In jedem Simulationszyklus wird die act()-Methode des Actor aufgerufen, wobei der Benutzer die gewünschte Aktion ausführt. Dieses Konzept hat den Vorteil, dass act() in einer strikten Reihenfolge (die geändert werden kann) und von demselben Thread aufgerufen wird. Dadurch werden Gleichzeitigkeitsprobleme vermieden, wie sie bei vollständig aktiven Objekten auftreten, bei denen jedes Objekt seinen eigenen Thread hat. Ein Nachteil ist, dass alle Actors kooperativ sein müssen: act() muss schnell zurückkehren, um anderen Actors die Chance zu geben, zu handeln. Wenn Actors nicht synchron agieren sollen, kann die Ausführung für eine bestimmte Anzahl von Simulationszyklen übersprungen werden, indem setSlowDown() individuell benutzt (gesetzt) wird.

Call act() of all actors

Copy sprites in bitmap buffer

Auszug aus der GameGrid-Dokumentation: <http://www.aplu.ch/classdoc/jgamegrid/ch/aplu/jgamegrid/Actor.html>

Actors are partially active objects, animated by the same thread that runs the simulation loop.

In every simulation cycle, the actor's act() method is called, where the user performs the desired action.

This concept has the advantage that act() is called in a strict order (that can be modified) and by the same thread.

This avoids concurrency problems that is an issue with fully active object where each object has its own thread.

As a drawback all actors must be cooperative: act() must return quickly in order to give other actors the chance to act.

If actors should not act synchronously, the execution may be skipped for a given number of simulation cycles by using individually setSlowDown()