

GT4: Game and Simulation Technologies

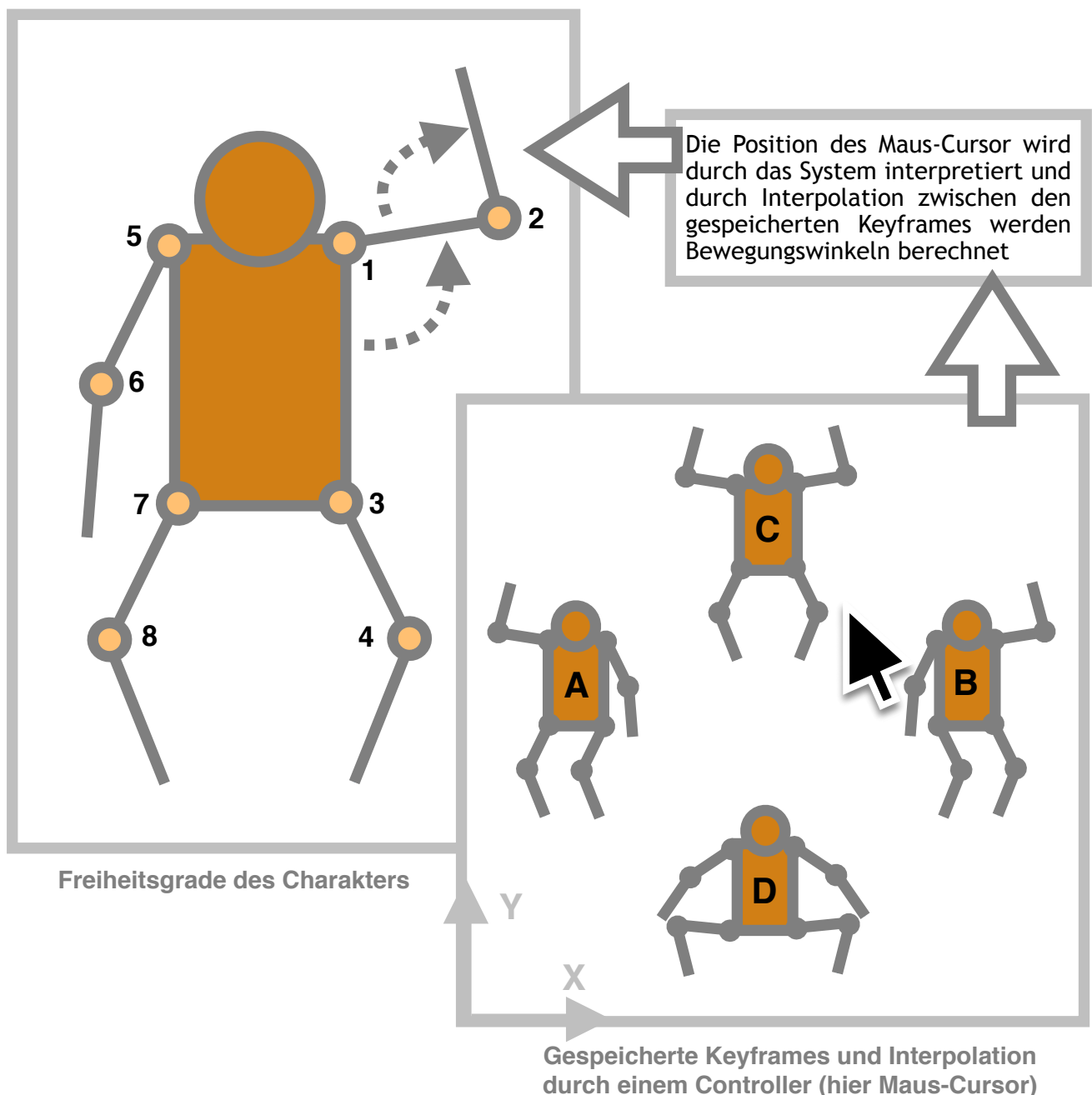
Abstract

Das Animieren von Objekten bzw. Charakteren - In Spielen oder Animationen - ist meistens mit fertig gerenderten Keyframes verbunden, d.h. alle mögliche Bewegungen müssen komplett gespeichert werden und während des "Echtzeit"-Animieren wiedergerufen werden. Was an Rechenressourcen spart aber die Beweglichkeit des Charakters eingrenzt.

Die Idee

Da nicht unendlich viele keyframes gespeichert werden können, wird durch die Simulation von Bewegungen die Interpolation zwischen zwei gespeicherte Keyframes verwendet und in Echtzeit gerendert.

Die Idee bei diesem Projekt besteht daraus die Interpolation zwischen mehreren gespeicherten Keyframes zu realisieren (Beispielsweise mit 4 Keyframes). Und damit die Bewegungsfreiheit eines Charakters nicht zu begrenzen.

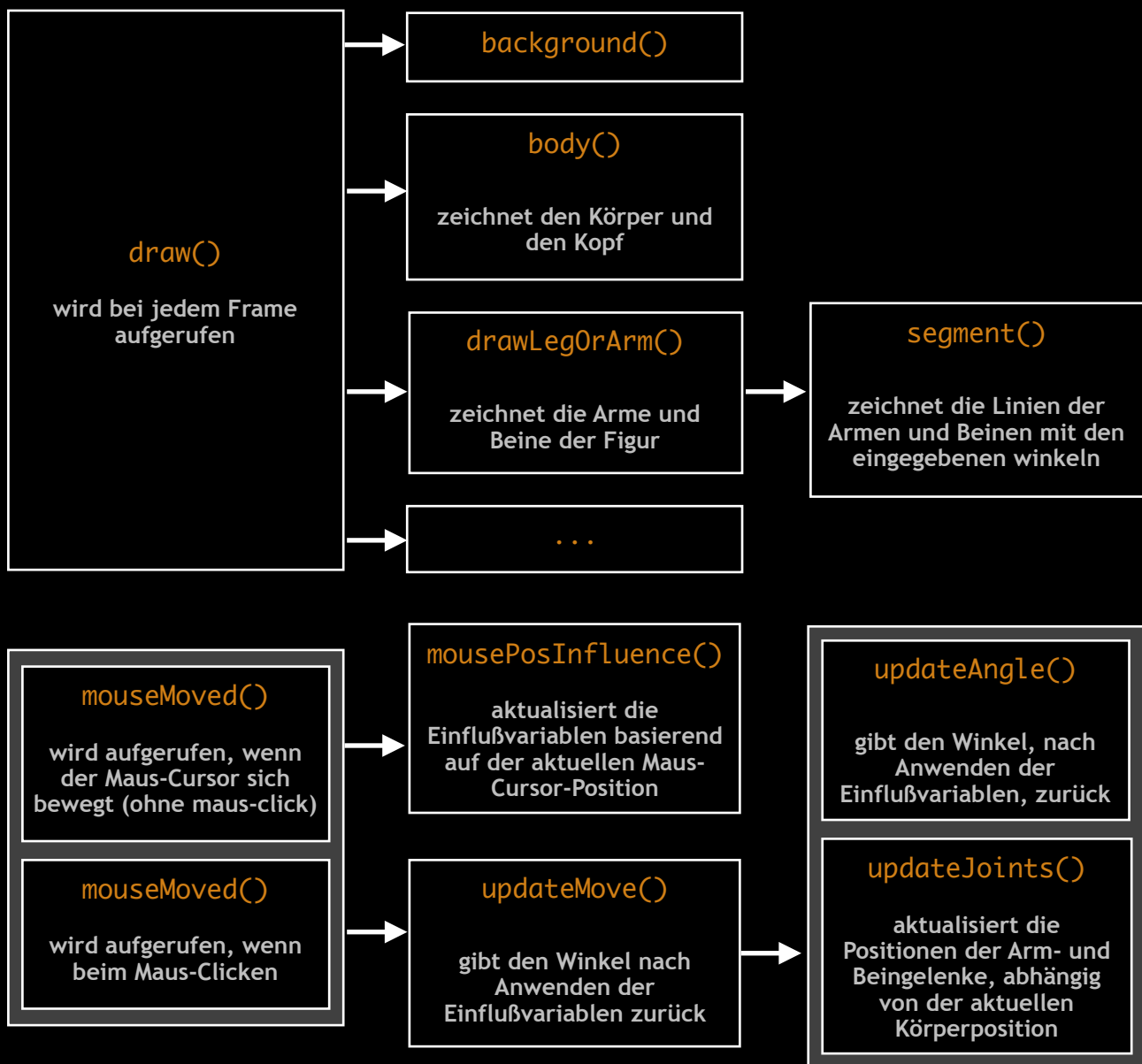


Algorithmus

Der Code zur Applikation der Charakteranimation ist in der Open Source Programmiersprache Processing (version 2.2.1) geschrieben und besteht aus drei Hauptteilen:

- Globale Variablen (Anfangswinkeln, Winkeln zu den 4 vordefinierten Körperhaltungen: A/B/C/D, Hintergrundbild, etc.), die von den Funktionen manipuliert werden, und Variablen, die dazu dienen die Bearbeitung der Darstellung zu erleichtern (z.B. Länge der Arme und Beine, Körpergrößen, Anfangsposition des Körpers, etc.).
- Processing Grundstrukturen: `setup()`, `draw()`, `mouseMoved()`, `mouseDragged()`
- Methoden der Charakteranimation: `mousePosInfluence()`, `updateAngle()`, `drawLegOrArm()`, `segment()`, `updateMove()`

Damit keine Rechenressourcen unnötig benutzt werden, werden in `draw()` nur die Zeichenmethoden aufgerufen. Während die Methoden, die den Charakter animieren, nur bei einer Mausbewegung, aufgerufen werden.



Weitere Details zum Algorithmus befinden sich in dem Code als Kommentare.