Präsentation

Ziel dieser Phase war der Entwurf einer objektorientierten Architektur für unser Simulationsprogramm.

Zunächst ein kurzer Nachtrag zur Spezifikation: TODO

Kommen wir zur Programmarchitektur. Unser Entwurf basiert im Wesentlichen auf dem Model-View-Controller Prinzip, besteht also aus drei großen Oberpaketen: View, Controller und Model. Die View informiert den Controller über Interaktionen des Nutzers mit der UI. Dieser verwendet denn gegebenenfalls das Model, um die Benutzeranfragen zu bearbeiten, also etwa um eine Simulation zu starten oder eine Konfigurationsdatei zu laden. Da die Durchführung im Allgemeinen durchaus lang dauern kann, werden Simulationen asynchron gestartet. Der Controller beobachtet dann das Model, um über Neuigkeiten im Simulationsprozess informiert zu werden und gegebenenfalls die View entsprechend zu aktualisieren.  
Wie man sieht sind durch diesen Entwurf View und Model komplett voneinander entkoppelt und wir haben das Model so entworfen, dass es keine Annahmen darüber trifft, wie es verwendet wird; dadurch wird das Model wiederverwendbar.

Als UI-Framework verwenden wir JavaFX, das heißt die GUI ist vollständig in xml beschrieben und die View-Klassen werden zur Laufzeit daraus erzeugt. Mit jedem der Fenster ist ein Controller assoziiert, der zusammen mit dem Fenster erzeugt wird und dessen Nutzereingaben verarbeitet.

* Controller

Die Controller sind in einer baumartigen Hierarchie angeordnet. Oben steht der HeadController. Dieser gehört zum Startfenster und wird damit gleich zu Programmstart erzeugt. Darunter befinden sich Controller für die verschiedenen Popup-Fenster wie das Konfigurationsfenster oder die Fenster zur Erstellung von Strategien, Populationen, etc. sowie für die Ergebnisausgabe im Hauptfenster.