

1 Thema

Projektgruppe: GenGa - An Engine for Generative Game Development

2 Zeitraum

Sommersemester 2017 und Wintersemester 2017/2018

3 Veranstalter

Dr. Johannes Neubauer <johannes.neubauer@tu-dortmund.de>, Tel. 7779

Dipl.-Inf. Dominic Wirkner <dominic.wirkner@tu-dortmund.de>, Tel. 5806

Prof. Dr. Bernhard Steffen <steffen@cs.tu-dortmund.de>, Tel. 5801

Informatik Lehrstuhl 5, Otto-Hahn-Straße 14, Raum 112/107

4 Aufgabe

4.1 Ziele

Die Spieleindustrie ist mit den immer leistungsfähigeren PCs und Spielekonsolen dazu übergegangen dem Trend der Filmindustrie zu folgen und immer umfangreichere, teurere Produktionen anzustreben. Einige der größeren Titel bewegen sich dabei in ähnlichen Regionen wie ausgewachsene Hollywoodproduktionen (z.B. Grand Theft Auto 5 mit über 130 Mio. USD¹).

Das erschwert den Einstieg für sogenannte Indie-Game-Developer, die versuchen ohne den Rückhalt eines großen Publishers, wie Activision oder Ubisoft, Spiele zu entwickeln und zu vermarkten. Dennoch ist und bleibt es ein Traum vieler Informatiker.

Ein Lichtblick ist der, eine Art Renaissance erlebende, Markt für Casual-Games. Das sind Spiele die, anstatt etablierte Genres zu bedienen, durch innovative Ideen Spieler gewinnen, welche sich selbst nicht als ambitionierte Gamer bezeichnen. Grund für die erneute Popularität ist die Verbreitung mobiler Endgeräte wie Smartphones und Tablets sowie Streaming-Boxen für den heimischen Fernseher. Doch auch die Produktion von Casual-Games birgt einige Hürden durch die Vielzahl der zu unterstützenden Plattformen und zugehörigen Entwicklungsumgebungen, Programmiersprachen und Bibliotheken.

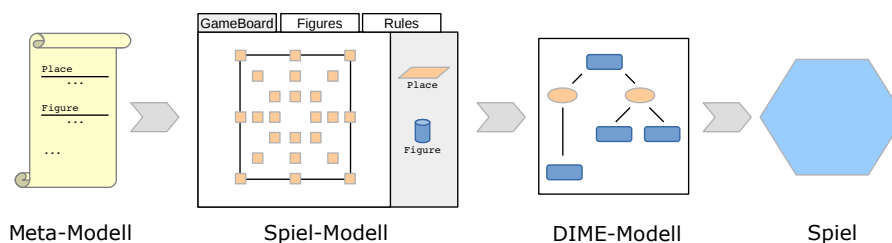


Abbildung 1: Darstellung der Toolchain und Übergänge durch Generierung

Es kommt erschwerend hinzu, dass im Zeitalter der *Always-On-Bewegung* und *Web 2.0* von einem erfolgreichen Spiel diverse Online-Funktionen, beginnend bei Mehrspielerfähigkeit und nicht endend bei der Integration sozialer Medien wie Facebook, Youtube und Twitch² erwartet werden.

¹https://en.wikipedia.org/wiki/List_of_most_expensive_video_games_to_develop

²Streamingplattform für Computerspiele, <https://www.twitch.tv>

Aufgabe der Projektgruppe wird es sein, eine effiziente Toolchain insbesondere für rundenbasierte Spiele zu entwickeln, die durch den Einsatz generativer Methoden von der Problematik der technischen Umsetzung abstrahiert. Ähnlich wie bei einigen Game-Engines (z.B. Unity³) sollen die Hürden bei der Entwicklung von Spielen gesenkt werden. Die Besonderheit ist der Einsatz moderner Methoden der modellbasierten Softwareentwicklung. Die Abkürzung des Titels der Projektgruppe *GenGa* soll dabei eine Analogie zu dem Spiel Jenga⁴ herstellen. Jenga ist rundenbasiert und zeichnet sich durch klare, einfache Regeln aus.

Der Fokus der Projektgruppe liegt u.a. auf der Erweiterung der Modellierungsumgebung für Webanwendungen DIME (Dynamic Web Application Integrated Modeling Environment, [1]) durch die Fähigkeit animierte 2D-Graphiken in Form eines Canvas zu modellieren und darzustellen. Die Vorgänger von DIME wurden bereits erfolgreich in PGs angewendet (z.B. für die Verwaltung von Daten für die Forschung im biomedizinischen Bereich⁵ sowie für Serious-Games⁶).

Für die Modellierung der Spielkomponenten, Spielregeln und Nicht-Spieler-Charaktere (NPC) werden jeweils graphische Modellierungssprachen als domänenspezifische Sprachen (DSL, [10]) entstehen. Die daraus resultierende Sprachfamilie soll die vollständige Beschreibung insbesondere rundenbasierter Spiele ermöglichen. Weiterhin wird die Projektgruppe eine Transformation umsetzen, die Spezifikationen aus der o.g. Sprachfamilie in die Modellierungsumgebung DIME übersetzt, um dann vollautomatisch zu einem Spiel generiert zu werden.

Hierbei kann die PG in großen Teilen auf Technologien zurückgreifen, welche am Lehrstuhl für Programiersysteme eingesetzt werden (siehe Abschnitt 4.2). Durch den generativen Ansatz werden Belange von einander getrennt (Separation of Concerns). Dies betrifft zum Beispiel die Abgrenzung der „was“- und „wie“-Beschreibungen, wodurch die technische Umsetzung austauschbar gemacht wird, ohne die Spezifikation eines Spieles anpassen zu müssen. Mit einem geeigneten Generator lassen sich dadurch neben einer browserbasierten Webanwendung auch Produkte für andere Systeme (z.B. Android, iOS) erstellen.

Die Projektgruppe evaluiert ihre Ergebnisse anhand der Umsetzung eines konkreten Spiels. Ein geeignetes Spiel können die Teilnehmer selbst wählen oder auch eine vollkommen neue Spielidee entwickeln. Aktuelle und vergangene Produkte der Spieleindustrie bieten dabei genügend Auswahl. Vorstellbar wären z.B. Civilization, die Siedler oder Fusionate (siehe Abbildung 2).



Abbildung 2: Screenshot aus Fusionate.

³<https://unity3d.com/de>

⁴<https://de.wikipedia.org/wiki/Jenga>

⁵<http://ls5-www.cs.tu-dortmund.de/cms/de/lehre/pgs/586-probio/index.html>

⁶<http://ls5-www.cs.tu-dortmund.de/cms/de/lehre/pgs/592-krisenplan/index.html>

4.2 Technologien

Für die Realisierung der Ziele bietet sich eine Kombination verschiedener Technologien und Konzepte an, die im Folgenden näher beschrieben werden.

Meta-Modellierung Im Rahmen der PG soll eine Familie domänenspezifischer Sprachen (DSL, [10]) entworfen werden. Dabei werden Konzepte der Meta-Modellierung angewendet, indem von konkreten Beispielen allgemeine Aspekte von rundenbasierten Spielen abstrahiert werden. Die Sprachfamilie der Domäne schränkt die mit ihr erstellbaren Produkte auf diese Art von Spielen ein. Sie erlaubt im Gegenzug eine sehr viel effizientere Umsetzung.

Dazu eignet sich die am Lehrstuhl entwickelte Open-Source Software CINCO Meta Tooling Suite⁷ [6]. CINCO ist ein auf dem Eclipse Modeling Framework (EMF, [8]) basierendes Werkzeug, mit dem aus Metamodell-Beschreibungen von Modellierungssprachen[10] automatisch spezialisierte Modellierungsumgebungen (RCP, [4]) für diese Sprachen generiert werden können.

Modell-Transformation In der Projektgruppe soll nicht auf der sprichwörtlichen „grünen Wiese“ entwickelt werden. Es besteht am Lehrstuhl bereits die Modellierungsumgebung DIME [1] sowie die zugehörige Laufzeitumgebung DyWA [7] (Dynamic Web Application) für Webanwendungen. Es gilt Spezifikationen rundenbasierter Spiele in Form von Modellen in der o.g. Sprachfamilie geeignet auf Spezifikationen von DIME abzubilden. Im Kontext von Model-Driven-Development (MDD[9][5]) nennt man diese Abbildungen auch Transformationen.

Prozessorientierte Entwicklung Modellgetriebene Softwareentwicklung bietet neben der Möglichkeit modellbasierte Verifikationstechnologien, wie zum Beispiel Model-Checking, anwenden zu können auch für Nicht-Programmierexperten den Vorteil, domänenspezifische Konzepte in Form von Prozessen einfach umzusetzen.

Abläufe von Spielen werden häufig in Phasen unterteilt, in denen die Spieler in einer vorher festgelegten Reihenfolge mögliche Aktionen durchführen. Die Spielregeln beschreiben, welche Aktionen abhängig von der aktuellen Spielsituation möglich sind. Hier bietet sich eine Modellierungssprache an, mit welcher der Spielverlauf als Prozess modelliert werden kann.

Generative Programmierung Eine Besonderheit von CINCO ist, dass nicht nur die Entwicklung der Modellierungsumgebung selbst vereinfacht wird, sondern auch die Realisierung dazugehöriger Codegeneratoren, welche ausführbaren Code zu den entsprechenden Modellen generieren. Hierzu setzt CINCO Methoden und Technologien der generativen Programmierung [2] und der modellbasierten Codegenerierung [3], wie zum Beispiel Xtend⁸, ein. Xtend ist eine objektorientierte Programmiersprache, die besonders zur Implementierung templatebasierter Codegeneratoren geeignet ist.

5 Teilnahmevoraussetzungen

Vorausgesetzt werden:

- Fundierte Kenntnisse in mindestens einer objektorientierten Programmiersprache; bevorzugt Java
- Elementare Kenntnisse über Modellierungstechniken, wie sie zum Beispiel in den Vorlesungen „Formale Methoden des Systementwurfs“, „Virtualisierung und Compilation“, „Softwarekonstruktion“ und „Dienstleistungsinformatik“ vermittelt werden

Wünschenswert sind zudem:

- Kenntnisse im Umgang mit Eclipse-Werkzeugen und -Programmierung
- Kenntnisse über Geschäftsprozesse
- Grundlegendes Interesse an der Entwicklung domänenspezifischer Sprachen

⁷Cinco SCCE Meta Tooling Suite, <http://cinco.scce.info>.

⁸Xtend - Modernized Java, <http://www.xtend-lang.org>

6 Minimalziel

Im Rahmen der PG sind folgende Ziele mindestens zu erreichen:

- Entwurf einer Sprachfamilie zur Definition rundenbasierter Spiele
- Anwendung der Modellierungssprachen an einem Beispiel
- Implementierung einer Modelltransformation zu DIME
- Erweiterung von DIME zur Möglichkeit der Verwendung eines Canvas

7 Literatur

- [1] S. Boßelmann, M. Frohme, D. Kopetzki, M. Lybecait, S. Naujokat, J. Neubauer, D. Wirkner, P. Zweihoff, and B. Steffen. DIME: A Programming-Less Modeling Environment for Web Applications. In *Proc. of the 7th ISOLA, Part II (2016)*, volume 9953 of *LNCS*, pages 809–832. Springer, 2016.
- [2] K. Czarnecki and U. W. Eisenecker. *Generative programming: methods, tools, and applications*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 2000.
- [3] S. Jörges. *Construction and Evolution of Code Generators - A Model-Driven and Service-Oriented Approach*, volume 7747 of *LNCS*. Springer Berlin Heidelberg, Germany, 2013.
- [4] J. McAffer, J.-M. Lemieux, and C. Aniszczyk. *Eclipse Rich Client Platform*. Addison-Wesley Professional, 2nd edition, 2010.
- [5] G. Mussbacher, D. Amyot, R. Breu, J.-M. Bruel, B. H. C. Cheng, P. Collet, B. Combemale, R. B. France, R. Heldal, J. Hill, J. Kienzle, M. Schöttle, F. Steimann, D. Stikkorum, and J. Whittle. The Relevance of Model-Driven Engineering Thirty Years from Now. In *Proc. of the 17th MODELS (2014)*, number 8767 in *LNCS*, pages 183–200. Springer International Publishing, 2014.
- [6] S. Naujokat, L.-M. Traonouez, M. Isberner, B. Steffen, and A. Legay. Domain-Specific Code Generator Modeling: A Case Study for Multi-faceted Concurrent Systems. In *Proc. of the 6th ISOLA, Part I (2014)*, volume 8802 of *LNCS*, pages 463–480. Springer, 2014.
- [7] J. Neubauer, M. Frohme, B. Steffen, and T. Margaria. Prototype-Driven Development of Web Applications with DyWA. In *Proc. of the 6th ISOLA, Part I (IsoLA 2014)*, number 8802 in *LNCS*, pages 56–72. Springer, 2014.
- [8] D. Steinberg, F. Budinsky, M. Paternostro, and E. Merks. *EMF: Eclipse Modeling Framework (2nd Edition)*. Addison-Wesley, Boston, MA, USA, 2008.
- [9] M. Völter, T. Stahl, J. Bettin, A. Haase, and S. Helsen. *Model-driven software development: technology, engineering, management*. John Wiley & Sons, 2013.
- [10] M. Völter, S. Benz, C. Dietrich, B. Engelmann, M. Helander, L. C. L. Kats, E. Visser, and G. Wachsmuth. *DSL Engineering - Designing, Implementing and Using Domain-Specific Languages*. dslbook.org, 2013.

8 Rechtlicher Hinweis

Die Ergebnisse der Projektarbeit und die dabei erstellte Software sollen der Fakultät für Informatik uneingeschränkt für Lehr- und Forschungszwecke zur freien Verfügung stehen. Darüber hinaus sind keine Einschränkungen der Verwertungsrechte an den Ergebnissen der Projektgruppe und keine Vertraulichkeitsvereinbarungen vorgesehen.