



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Projektbericht Sommersemester
2009
Björn Jensen

Vorbereitenden Arbeiten zur semiautomatischen
Generierung von Simulationsumgebungen

Inhaltsverzeichnis

1 Einführung	3
2 Navigatorische Schwerpunkte	3
3 Definitionen	4
3.1 Positionierung	4
3.2 Orientierung	4
3.3 Navigation	5
3.4 Leitung	5
3.5 Routing	5
4 Anforderungen	6
5 Evaluation	6
5.1 Import von 3D/2D-CAD-Daten	6
5.2 Simulationsumgebung	7
5.3 Architektur der Entwicklungsumgebung	7
6 SeaGS	8
6.1 Vorgehen	8
6.2 Simulation	9
6.3 Mockobjekte und Artefakte	10
7 Zusammenfassung und Ausblick	11
Literatur	12

1 Einführung

Navigationssysteme nehmen heutzutage einen immer größeren Stellenwert ein. Die Verfügbarkeit eben jener Systeme wächst rasant und deren Anforderungen werden mittlerweile durch mobile Kleinstgeräte erfüllt. So erstreckt sich die Bandbreite der Umgebungen, in denen Navigationssysteme vorhanden sind, von großen Fahrzeugen über explizite mobile Lösungen (bspw. von TomTom) hin zu Applikationen, die auf Mobiltelefonen installiert werden können. Der Fokus dieser Systeme liegt jedoch in allen Fällen auf dem Anwendungsbereich außer Haus - outdoor. Für den Bereich der sogenannten Indoornavigation gibt es am Markt keine etablierten Lösungen, die die in Kapitel 2 genannten Punkten in Gänze abdeckt. Vielmehr handelt es sich - sollte ein System vorhanden sein - um die singuläre Betrachtung eines Schwerpunktes als Anwendung für einen der in Kapitel 2 genannten Punkte. Das Finden von Lösungen, die alle Punkte aus Kapitel 2 abdecken, gestaltet sich als schwierig. Der Forschungsbereich Ambient Navigation an der Hochschule für Angewandte Wissenschaften Hamburg beschäftigt sich mit den genannten Punkten im Kontext Indoornavigation. Um Aufwände jeglicher Art zu minimieren, ist es notwendig, die erdachten Konzepte, Theorien und Lösungen zu virtualisieren. Dazu soll in diesem Projekt eine Entwicklungsumgebung erstellt werden, welche aus herkömmlichen Daten im Kontext Gebäudearchitektur die Grundlagen für eine virtuelle Umgebung schafft und die Möglichkeiten bereitstellt, möglichst einfache Konzepte, Theorien und technische Lösungen in der virtuellen Umgebung als auch die Umgebung selbst zu erstellen, zu erforschen, zu verifizieren, zu modifizieren und zu optimieren.

2 Navigatorische Schwerpunkte

Die in Kapitel 1 genannten Produkte und Lösungen beziehen sich allesamt auf den Anwendungsfall außerhalb geschlossener Gebäude. Der Einsatz innerhalb von Gebäuden gestaltet sich deutlich schwieriger, da bestimmte Rahmenbedingungen hier nicht gelten. Elementar ist beispielsweise die Position des Anwenders. Diese kann im Außenbereich bspw. mittels GPS oder Galileo ermittelt werden. Im Innenbereich sind diese Signale jedoch nicht oder nur sehr schwach verfügbar, so dass diese für die Ermittlung der Position des Anwenders unzureichend sind. Es müssen Alternativen u.a. zur Positionsbestimmung und andere Grundlagen hierfür geschaffen werden, damit die grundlegenden Funktionalitäten realisiert werden können:

- Positionierung
- Orientierung
- Navigation
- Leitung

- Routing

3 Definitionen

Die Begriffe aus Kapitel 2 werden nachfolgend in dem Umfang definiert, wie es zum Verständnis dieser Arbeit notwendig ist.

3.1 Positionierung

Unter der Position wird der momentane Aufenthaltsort der Anwenders in einem topografischen Raum verstanden (de Lange (2005)). Zur Bestimmung der Position sind verschiedene Techniken geeignet. Man kann hier zwischen aktiven und passiven Verfahren unterscheiden.

Aktive Verfahren sind solche Verfahren, bei denen die Position im Bezugssystem durch das Mitführen entsp. Komponenten immer gegeben ist. Unter passiver Positionierung versteht man die Bestimmung der Position durch Interaktion mit Komponenten, deren Position im System bekannt ist. Die Nutzung einer solchen Komponente erlaubt Rückschlüsse auf die Position des Anwenders. Das kann bspw. durch Interaktion mit RFID-Kartenlesern geschehen. Somit ist die Position des Anwenders implizit bekannt. Jedoch kann hier nicht zu jeder Zeit eine Aussage darüber getroffen werden, wo sich der Anwender befindet.

3.2 Orientierung

In der Medizin und der Allgemeinen Psychologie gilt die mentale Orientierung als die kognitive Fähigkeit, sich bezüglich Zeit, Ort, Situation und der eigenen Person (autopsychische Orientierung) zurechtzufinden (Pschyrembel (2007)). Als Voraussetzungen für eine normgerechte Orientierung gelten ein ungestörtes Bewusstsein, eine leistungsfähige Wahrnehmung, Aufmerksamkeit, Zeitsinn und Gedächtnis. Wenn einer dieser Punkte negativ belastet ist, stellt sich die Frage der Assistenz. Was kann zur Kompensation der Negativbelastung eingesetzt werden? Wenn also neben der Navigation auch die Orientierung verbessert werden soll, ist eine Kenntnis des Kontextes und der Einflüsse auf die o.g. Punkte zwingend notwendig. Hierbei sind bestimmte charakteristische Eigenschaften der Umgebung zu bestimmten Zeitpunkten oder permanent hervorzuheben. Dies können auch assistierende Artefakte sein.

Im allgemeinen erfolgt eine Orientierung und eine Navigation mittels Landmarken. Dies sind markante, für den Anwender eindeutig identifizierbare Punkte im topografischen Raum (Pohl (2009)).

3.3 Navigation

Unter Navigation versteht man im allgemeinen das Zurechtfinden in einem topografischen Raum zum Erreichen eines gewünschten Ortes (Pohl (2009)). Dabei wird wie folgt vorgegangen:

1. Bestimmen der geografischen Position durch Ortsbestimmung nach verschiedensten Methoden,
2. Berechnen des optimalen Weges zum Ziel und
3. Führen des Fahrzeugs zu diesem Ziel, also vor allem das Halten des optimalen Kurses.

Wie die Position bestimmt werden kann, wurde in Kapitel 3.1 erläutert, auf die Berechnung von optimalen Wegen wird in Kapitel 3.5 näher eingegangen und der letzte Punkt der o.g. Aufzählung wird in Kapitel 3.4 geklärt.

3.4 Leitung

Unter der Leitung versteht man in diesem Kontext das Anleiten des Navigateurs zur Erreichung des gewünschten Ortes innerhalb des topografischen Raumes. Dieser soll sich möglichst an die optimalen Route halten.

Damit das System in der Lage ist, den Benutzer entsprechend der Vorgabe optimal zu führen, muss dieses zum Zeitpunkt einer möglichen Interaktion Kenntnis über die aktuelle Position und die Aktionsmöglichkeiten des Nutzers besitzen. Wenn es keine Interaktion des Systems mit dem Benutzer gibt, müssen an dieser Stelle Annahmen darüber getroffen werden, wo sich dieser zu einem Zeitpunkt X befinden (ähnlich Dead Reckoning) und zu welchem Ziel er sich bewegen könnte.

3.5 Routing

Bei der Berechnung eines optimalen Weges zum Ziel sind gewisse Randinformationen nötig. Die mathematischen Grundlagen zur Berechnung sind hinreichend bekannt (Golden u. a. (2008)). Grundlegend ist hier, dass das System zur Kalkulation über eine Abbildung der realen Wege auf einen Graphen (bestehend aus Ecken und Kanten) verfügen muss. Im allgemeinen werden Wege durch die Kanten und Wegpunkte/Ziele durch die Ecken dargestellt. Bei Bedarf kann noch eine Kantengewichtung und -richtung mit aufgenommen werden. Dabei könnte die Gewichtung im Fall der Navigation schwierige Wege kennzeichnen (Diestel (2006)). Die Richtung einer Kante könnte bspw. eine Absperrung symbolisieren, die nur in eine Richtung durchschritten werden kann (Drehtür).

4 Anforderungen

In Kapitel 1 wurde erwähnt, dass für Erstellung von Simulationsumgebungen herkömmliche Daten im Kontext Gebäudearchitektur. Dies sind in der Regel 3D-CAD-Daten. In Fall der Hochschule für Angewandte Wissenschaften Hamburg lagen diese Daten nicht vor. Statt dessen lassen sich aus dem computergestützten Facilitymanagementsystem 2D-CAD-Daten extrahieren. Zu diesen kommen dann die entsprechenden Höhenschnitte in Form von PDF-Dateien.

Aus diesen soll dann eine virtuelle Umgebung geschaffen werden, durch die der Nutzer mit den herkömmlichen Sicht- und Steuergeräten (Monitor, Tastatur, Mouse) navigieren und ggf. interagieren kann. Der Ersteller der Umgebung soll ebenfalls die Möglichkeit haben, die Umgebung zu modifizieren, sei es durch Abbildung denkbaren Menschengaufkommens, Verkehr, Geräusche, Witterungsbedingungen etc. Zu dem kommt die einfache Erstellung von zu simulierenden Artefakten wie bspw. PublicDisplays, Informationsterminals usw. Die Umsetzung sollte möglichst unabhängig von Hochsprachen sein.

5 Evaluation

Die in Kapitel 4 genannten Punkte führten zu einer Recherche in folgenden Bereichen mit folgenden Ergebnissen:

5.1 Import von 3D/2D-CAD-Daten

Es gibt viele verfügbare Lösungen zum Im- und Export von CAD-Daten. Auch die Bearbeitung derselbigen stellt in der Regel kein Problem dar. Jedoch ist die Verschmelzung von 2D-CAD-Daten und Höhenschnitten in PDF so nicht vorgesehen. Also wurde dazu eine eigene Lösung erstellt.

Der Import erfolgt in drei Schritten:

1. Einlesen der 2D-CAD-Daten
2. Einlesen der Höhenschnitte
3. Erstellen der Verknüpfungen

In aktuellen Stand muss man manuell entsprechende Verknüpfungen der Höhendaten mit den 2D-Daten erstellen. In Zukunft soll dies weitestgehend automatisiert geschehen.

5.2 Simulationsumgebung

Damit die virtuelle Umgebung möglichst natürlich wird, werden die Technologien verwendet, die sich in einem verwandten Kontext bewährt haben: der Spiele-Industrie. Es werden 3D-GameEngines evaluiert, auf deren Basis die Simulationsumgebung erstellt werden soll. Dazu wurde folgende Vorauswahl getroffen:

- Ogre
- Blender
- jMonkey
- OpenArena
- DelID

Bei den vier erstgenannten Systemen handelt es um OpenSource-Engines, hinter denen eine mehr oder weniger große Community steckt. Ogre und Blender sind in diesem Kontext die größten Gemeinschaften und sehr aktiv. In Bezug auf die hochsprachliche Umsetzung gibt es sowohl für Blender als auch für Ogre Wrapper für die populärsten Sprachen. jMonkey ist eine 3D-GameEngine komplett auf Basis von Java und in Hinblick auf die Performanz nicht ganz so schnell. Für kleine Anwendungen ohne große Rechenleistung sollte die Verwendung von jMonkey kein Problem darstellen. Rechenintensivere Szenarien sind in der Handhabung und der Benutzbarkeit auf Grund der Performanz eher nicht auf dieser Plattform umzusetzen. Ein Vorteil dieser Engines ist die Verfügbarkeit aller simulierbarer Aspekte (Umwelt, Geräusche, Physik etc.).

OpenArena ist sehr performant, jedoch erfolgt die Simulation auf Basis von Karten für das zu Grunde liegende Spiel. Generell erreicht man mit der Verwendung von OpenArena schnell ansehnliche Ergebnisse. Die Integrationen weiterer Personen ist durch die Verwendung von Bots sehr einfach. Die Integration neuer Artefakte ist jedoch mit höherem Aufwand verbunden.

DelID ist eine sehr einfache Simulationsumgebung und sowohl von der Flexibilität in Richtung Hochsprachen als auch in Richtung Performanz eher schlecht. Jedoch ist die Einarbeitung in diese Umgebung sehr einfach und somit sollte sich diese für erste Gehversuche in diesem Bereich durchaus angesehen werden.

5.3 Architektur der Entwicklungsumgebung

Die Erstellung der in Kapitel 1 genannten Entwicklungsumgebung wird losgelöst von der hochsprachlichen Umsetzung betrachtet. Die vorangegangene kurze Erläuterung der Zielplattformen - der 3D-GameEngines - zeigt, dass die IDE möglichst viele Hochsprachen beherrschen

sollte. Außerdem sollte die Architektur der IDE eine hohe Flexibilität und Erweiterbarkeit vorsehen.

Hier fiel die Wahl auf Verwendung der Eclipse-RCP-Plattform. Das Eclipseprojekt ist genau auf die Anforderungen ausgelegt und beherrscht eine Vielzahl von Hochsprachen, so dass hier ein relative Unabhängigkeit gewährleistet werden kann. Die Erstellung einer eigenen DSL für die Loslösung der Paradigmen ist angedacht.

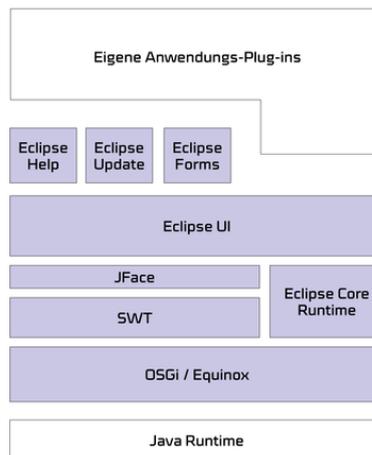


Abbildung 1: RCP im Detail

6 SeaGS

Der Projektname für die zu erstellenden Entwicklungsumgebung lautet SeaGS (semiautomatische Generierung von Simulationsumgebungen). Die folgenden Kapitel geben einen Überblick, wie der in Kapitel ?? genannte Schwerpunkt untersucht wird und wie SeaGS allgemein funktioniert. Notwendiges Hintergrundwissen zur Begründung des Vorgehens wird ebenfalls referenziert.

6.1 Vorgehen

Folgendes Vorgehen kann man zur Untersuchung von Methoden zur Verbesserung von Navigation, Leitung und Orientierung innerhalb von Gebäuden einige Herangehensweisen zu Grunde legen. Um die Methoden bewerten zu können, sind folgende Schritte notwendig (vergl. [Subramaniam und Hunt \(2006\)](#)):

- (Re)Definition von Testkriterien
- Abbilden der Realität auf eine Testumgebung
- Bereitstellen von Interaktionsmöglichkeiten, Artefakten und Mockobjekten
- Testdurchführung
- Feedback einholen
- Test ggf. erneut durchführen

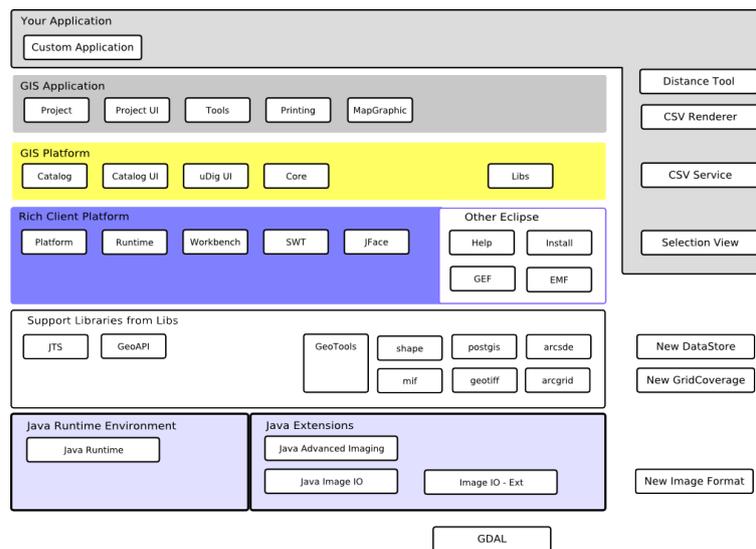


Abbildung 2: Eclipse RCP angereichert mit GIS Komponenten

6.2 Simulation

Alle vorgestellten Projekte simulieren in einem gewissen Umfang. Damit lassen sich Mehraufwände¹ durch Realisierung in Grenzen halten (Nygard (2007)). Des weiteren führt die Simulation zu kürzeren Iterationen als in Realexperimenten.

Virtuelle Umgebungen sind für diesen Zweck ideal. Dabei ist der Begriff der virtuellen Umgebung wie folgt definiert:

¹ Diese Mehraufwände können sich Zeit, Personenkraft, Geld etc. ausdrücken.

An artificial environment which is experienced through sensory stimuli (as sights and sounds) provided by a computer and in which one's actions partially determine what happens in the environment ([Webster \(2004\)](#)).

Diese lassen sich an Hand des Immersionsgrad unterteilen (vergl. [Menzel \(2004\)](#)). Unter dem Immersionsgrad versteht man den Grad des Eintauchens des Benutzers in die virtuelle Umgebung. Dem liegt die Evolution *Interaktion* \Rightarrow *Imagination* \Rightarrow *Immersion* zu Grunde. Der Benutzer hat das Gefühl, mittendrin statt nur dabei zu sein. Die Verwendung vorhandener Infrastruktur steht bei der Umsetzung mit im Vordergrund. Aus diesem Grund beschränkte sich der Immersionsgrad auf die Desktop VR. Ein höherer Immersionsgrad ist denkbar, aber nicht Ziel des Projektes.

6.3 Mockobjekte und Artefakte

Artefakte, mit denen man im Kontext des Szenarios zu tun hat, müssen simuliert werden. Je nach Anwendungsfall sind ggf. Daten aus der Umgebung zu berücksichtigen. Diese müssen ebenfalls in der virtuellen Umgebung abgebildet sein, um eine möglichst hohe Realitätsnähe zu gewährleisten. Dazu eignen sich sogenannte Mockobjekte (vergl. [Richardson und Gwaltney \(2005\)](#)). Zur Implementierung und Integration steht dafür das SeaGS-SDK zur Verfügung. Mockobjekte und Artefakte werden in einer Metasprache (einer Domain-Specific-Language ([Parr \(2009\)](#))) implementiert und mit dem SDK positioniert. Kapitel ?? liefert die Antwort, wie und wo eine Positionierung stattfinden muss. Des weiteren ist es möglich, äußere Einflüsse zu virtualisieren ([Hess \(2007\)](#) und [Roosendaal und Wartmann \(2009\)](#)).



Abbildung 3: Vorgehen zur Erstellung von Simulationsumgebungen mit SeaGS

7 Zusammenfassung und Ausblick

Bei der Arbeit in diesem Projekt mussten einigen Hürden genommen werden. Zum einen erforderte die Arbeit mit den entsprechenden Spieleumgebungen eine hohe Einarbeitungszeit. Zum anderen war die sinnvolle Aufbereitung der vorhandenen CAD-Daten sehr schwierig, was an der Zerteilung in 2D-CAD und PDF-Höhenschnitte lag. Zudem erschwerte die nicht einheitliche Unterteilung der CAD-Daten in sehr viele Layer die Importierung und Aufbereitung zusätzlich. Jedoch ist die Bearbeitung des Projekts so weit fortgeschritten, dass die Integration der Raumdaten und der Spieleumgebungen vorliegt und nun die weiteren Punkte in Richtung Endanwendbarkeit vollzogen werden können.

Literatur

- [Klann 2006] : *Emergency Response*. 2006
- [Schill 2009] : *Die Karte im Kopf: Strukturelle Eigenschaften der mentalen Raumrepräsentation*. 2009
- [Barkowsky u. a. 2005] BARKOWSKY, Thomas ; BATEMAN, John ; FREKSA, Christian ; BURGARD, Wolfram ; KNAUFF, Markus: Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition: Reasoning, Action, Interaction / Universität Bremen and Universität Freiburg. 2005. – Forschungsbericht
- [B.Hillier u. a. 1976] B.HILLIER ; A.LEAMAN ; P.STANSALL ; M.BEDFORD: Space Syntax. In: *Environment and Planning* Bd. 3 University College London (Veranst.), August 1976, S. 147 – 185
- [Brito 2008] BRITO, Allan: *Blender 3D Architecture, Buildings and Scenery*. Packt Publishing, 2008. – ISBN 9781847193674
- [Clark 2004] CLARK, Mike: *Pragmatic Project Automation*. The Pragmatic Bookshelf, 2004. – ISBN 0974514039
- [Czarnecki und Eisenecker 2000] CZARNECKI, Krzysztof ; EISENECKER, Ulrich W.: *Generative Programming. Methods, Tools and Applications*. Addison-Wesley, 2000. – ISBN 9780201309775
- [Diestel 2006] DIESTEL, Reinhard ; (Hrsg.): *Graphentheorie*. Springer Verlag, Berlin, 2006. – ISBN 9783540213918
- [Golden u. a. 2008] GOLDEN, Bruce L. ; RAGHAVAN, S. ; WASIL, Edward A. ; (Hrsg.): *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer Verlag, Berlin, 2008. – ISBN 9780387777771
- [Hantel 2009] HANTEL, Tim: *Erstellung von Simulationsumgebungen unmöglicher Räume*. 2009. – -
- [Hess 2007] HESS, Roland: *The Essential Blender: Guide to 3D Creation with the Open Source Suite Blender*. No Starch Press, 2007. – ISBN 9781593271664
- [Hillier 2005] HILLIER, Bill: The art of place and the science of space. In: *World Architecture 11/2005* University College London (Veranst.), 2005, S. 96 – 102
- [Hitchner] HITCHNER, Lewis E.: *The NASA Ames Virtual Planetary Exploration (VPE) Testbed*. NASA Ames Research Center

- [Hunt und Thomas 1999] HUNT, Andrew ; THOMAS, David: *The Pragmatic Programmer: From Journeyman to Master*. Addison-Wesley, 1999. – ISBN 9780201616224
- [Klann 2008] KLANN, Markus: *Playing With Fire: Participatory Design of Wearable Computing for Fire Fighters*, Universität Siegen, Dissertation, 2008
- [Klar und Klar 2006] KLAR, Michael ; KLAR, Susanne: *Einfach generieren. Generative Programmierung verständlich und praxisnah*. Hanser Fachbuchverlag, 2006. – ISBN 9783446404489
- [de Lange 2005] LANGE, Norbert de ; (Hrsg.): *Geoinformatik in Theorie und Praxis. 2.* Springer Verlag, Berlin, 2005. – ISBN 9783540282914
- [Menzel 2004] MENZEL, Moritz: *Virtual and Augmented Reality*, Universität München, Diplomarbeit, 2004
- [Nygard 2007] NYGARD, Michael ; (Hrsg.): *Release It!: Design and Deploy Production-Ready Software*. The Pragmatic Bookshelf, 2007. – ISBN 9780978739218
- [O'Sullivan und Turner 2001] O'SULLIVAN, David ; TURNER, Alasdair: Visibility graphs and landscape visibility analysis. In: *International Journal of Geographical Information Science* University College London (Veranst.), 2001, S. 221 – 237
- [Pallister 2005] PALLISTER, Kim: *Game Programming Gems 5*. Charles River Media, 2005. – ISBN 9781584503521
- [Parr 2009] PARR, Terence: *Language Design Patterns: Techniques for Implementing Domain-Specific Languages*. The Pragmatic Bookshelf, 2009. – ISBN 9781934356456
- [Pohl 2009] POHL, Friedrich-Wilhelm: *Die Geschichte der Navigation*. Koehlers Verlagsgesellschaft mbH, Juli 2009. – ISBN 9783782210003
- [Pschyrembel 2007] PSCHYREMBEL, Willibald: *Medizinisches Lexikon*. Walter de Gruyter, 2007. – ISBN 9783110185348
- [Richardson und Gwaltney 2005] RICHARDSON, Jared ; GWALTNEY, William A. ; (Hrsg.): *Ship It!: A Practical Guide to Successful Software Projects*. The Pragmatic Bookshelf, 2005. – ISBN 9780974514048
- [Roosendaal und Wartmann 2009] ROOSENDAAL, Ton ; WARTMANN, Carsten ; (Hrsg.): *The Blender GameKit: Interactive 3D for Artists*. No Starch Press, 2009. – ISBN 9781593272050
- [Shore und Warden 2007] SHORE, James ; WARDEN, Shane ; (Hrsg.): *The Art of Agile Development*. O'Reilly, 2007. – ISBN 9780596527679
- [Subramaniam und Hunt 2006] SUBRAMANIAM, Venkat ; HUNT, Andrew ; (Hrsg.): *Practices of an Agile Developer*. The Pragmatic Bookshelf, 2006. – ISBN 9780974514086

-
- [T.Barkowsky u. a. 2005] T.BARKOWSKY ; J.BATEMAN ; C.FREKSA ; W.BURGARD ; M.KNAUFF: Reasoning, Action, Interaction. In: *Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition*, 2005
- [Webster 2004] WEBSTER, Miriam: Virtual Reality. In: *ACM Journal*. New York, NY, USA : ACM Press, 2004
- [Zetsche u. a. 2007] ZETZSCHE, Christoph ; GALBRAITH, Christopher ; WOLTER, Johannes ; SCHILL, Kerstin: Navigation Based on a Sensorimotor Representation: a Virtual Reality Study. In: *Human Vision and Electronic Imaging XII*, 2007