

Projektantrag

[Jorge - Lego MindStorms]

Autoren	Nik Lutz [I3STW, lutzn@hti.bfh.ch] Stefan Feissli [I3STW, feiss@hti.bfh.ch] Christof Seiler [I3STW, seilc@hti.bfh.ch]
Version	1.05
Datum	10.03.2005
Status	Freigegeben
Webseite	jorge.hta-bi.bfh.ch



Inhaltsverzeichnis

1 Zweck des Dokuments	4
2 Management-Übersicht	4
3 Ausgangslage	5
3.1 Problemstellung	5
3.2 Anlass und Begründung des Projekts	6
3.3 Projektrahmenbedingungen	6
3.4 Situationsanalyse	6
3.5 Erbrachte Vorleistungen	7
4 Ziele und Lösungen	7
4.1 Zielvorstellungen	7
4.2 Mögliche Lösungen	8
5 Mittelbedarf	9
5.1 Bewertung der Sicherheits- und Datenschutzaspekte	9
5.2 Sachmittel	9
5.3 Personal	9
5.4 Ausbildung	10
6 Planung und Organisation	10
6.1 Projektorganisation	10
6.2 Termine	11
6.3 Prioritäten	11
7 Wirtschaftlichkeit	12
7.1 Kosten	12
7.1.1 Personalkosten	12
7.1.2 Sachkosten	12
7.1.3 Gesamtkosten	12
7.2 Nutzen	13
8 Konsequenzen	13
8.1 Auswirkung	13
8.2 Bei Nichtrealisierung	13
8.3 Bei verspäteter Realisierung	13
8.4 Auswirkung auf Schnittstellen zu anderen Systemen	14
8.5 Qualitätsverbesserung	14
8.6 Risikobeurteilung	14
8.7 Ausweichmöglichkeiten	15



9 Antrag

9.1 Bisherige Entscheide

9.2 Formulierung des Projektantrags

15

15

15

1 Zweck des Dokuments

In diesem Dokument wird die Ausgangslage für das Projekt "Jorge" definiert, um über das weitere Vorgehen zu entscheiden. Die für die Projektbeurteilung notwendigen Angaben werden in den folgenden Abschnitten zusammenfassend dargestellt.

2 Management-Übersicht

Das Ziel ist es, die im Handel für ein paar hundert Franken erhältlichen Lego MindStorms Roboter, auf dem Computer 3-dimensional abzubilden und dessen Verhalten zu simulieren.

MindStorms besitzen verschiedene Fähigkeiten. In der Standardausstattung sind folgende Komponenten enthalten:

- Ein RCX Mikrocomputer. Dieser ist sozusagen das Gehirn des Roboters. Er ermöglicht es Befehle durch eine Infrarotschnittstelle zu empfangen und auszuführen.
- Durch zwei Motoren kann der Roboter im Raum bewegt werden.
- Ein Lichtsensor dient dem Roboter zur Erkennung von Helligkeit.
- Zwei Berührungssensoren.
- Die Infrarotschnittstelle dient zur Übertragung von Befehlen oder ganzen Programmen.



Die Steuerung des Roboters erfolgt über die Infrarotschnittstelle. Es ist möglich dem Roboter via Computer einzelne Befehle zu geben oder ganze Programme zu übermitteln. So können sich MindStorms mithilfe intelligenter Programme selbständig in der Umwelt bewegen und auf Inputs der entsprechenden Sensoren verhältnismässig reagieren. Die Kommunikation mit anderen RCX Mikrocomputern ist ebenfalls möglich.



Die Realisierung dieser Simulation soll in zwei Schritten erfolgen.

1. Die Projektarbeit dient vor allem der Evaluation und der Entwicklung verschiedener Prototypen.
2. Während der Diplomarbeit sollen die Ergebnisse aus der Projektarbeit erweitert werden. wobei vor allem die Interaktion mit der Applikation verbessert werden soll.

Dieses Projekt wird im Rahmen der Projektarbeit an der HTI Biel durch die Studenten Niklaus Lutz, Stefan Feissli und Christof Seiler durchgeführt. Betreut werden die Studenten für dieses Projekt im technischen Bereich von Herrn Claude Fuhrer und Jean-Paul Dubois sowie im Bereich des Projektmanagement von Herrn Frank Helbling. Die Studenten werden für dieses Projekt die Führung, Entwicklung und Qualitätssicherung durchführen.

3 Ausgangslage

3.1 Problemstellung

Es bestehen bereits diverse 2-dimensionale Simulationen und Emulationen. Es ist jedoch schwierig ein System zu finden, das erstens 3-dimensional ist, zweitens einfach modifizierbar ist, drittens auf dem alle gängigen Roboterbetriebssysteme ausführbar sind. Mit unserem Tool versuchen wir eine skalierbare Umgebung zu schaffen, die es ermöglicht eigene Welten zu kreieren, um so dem Roboter virtuelles Leben einzuhauchen. Es wäre schwierig den Lego Robotern ein derart grosses und flexibles Universum in der Wirklichkeit bereitzustellen. Ausserdem ist es für gewisse Anwender (z. Bsp. Schulklassen) unrealistisch unzählige Roboter anzuschaffen, da der Preis, obwohl im Einzelnen nicht übermässig gross, jedoch in grossen Mengen sich schnell kumuliert. Andererseits können wir auch in der Realität nicht schnell mal eine kleine Stadt oder ein Labyrinth zusammenstellen. Alle diese Einschränkungen können wir dank der unendlich scheinenden Unwirklichkeit, also in der virtuellen Welt, nur mit ein paar Mausklicks realisieren.



3.2 Anlass und Begründung des Projekts

Einerseits ist der Grund für dieses Projekt sicher, dass es ein fester Teil unserer Ausbildung ist ein solches durchzuführen und somit obligatorisch, und andererseits ist das Projekt aus persönlichen Interessen entstanden. Es gibt bereits ein paar ähnliche Projekte, jedoch sind sie als Ganzes nicht das was wir uns von einem solchen versprechen würden. Da es schon eine sehr grosse „Lego MindStroms Community“ gibt, sollte unser Projekt leicht anpassbar und erweiterbar sein, um eine eventuelle Verbesserung oder Erweiterung durch andere Entwickler zu ermöglichen.

3.3 Projektrahmenbedingungen

- Die Anwendung sollte wenn möglich Multiplattform fähig sein, windows 2000 und höher sowie auf den gängigsten Linux Distributionen.
- Das Projekt wird mit Microsoft Visual .NET und einem Linux-Pendant (evtl. Anjuta oder auch mit Eclipse) realisiert.
- Das Projekt muss nach HERMES 2003 abgewickelt werden.

3.4 Situationsanalyse

Der Lego MindStroms Roboter ist frei im Handel für ein paar hundert Franken erhältlich. Die Standardausführung besteht aus 3 Sensoren, um die Umwelt wahrzunehmen, 2 Motoren, um den Roboter bewegen zu können, ein RCX-Computer, das „Gehirn“ des Roboters, ein Infrarot Sender/Empfänger, um Programme auf den Roboter zu laden und vielen Legobausteinen. Wie schon erwähnt, ist der RCX ein kleiner Computer der Befehle ausführen kann. Er ist mit einer proprietären Lego Firmware ausgerüstet, diese Firmware ist vergleichbar mit einem Betriebssystem. Es besteht ausserdem die Möglichkeit dieses Betriebssystem durch alternative Betriebssysteme auszutauschen. Solche Alternativen gibt es bereits, und meistens sind sie frei unter einer open source Lizenz erhältlich. Es existieren ausserdem verschiedene Emulatoren, die es ermöglichen Programme für solche Betriebssystem zu entwickeln und diese lokal auf dem PC zu testen. Diese verschiedenen Betriebssysteme ermöglichen es Entwicklern in ihrer bevorzugten Programmiersprache Programme zu schreiben, und sie auf dem Lego Roboter zu installieren. Es wäre also sinnvoll einen weg zu wählen der es ermöglicht möglichst unabhängig von einer Programmiersprache bzw. Betriebssystem zu bleiben, damit möglichst vielen Leuten unsere Simulation nutzen bringt.

3.5 Erbrachte Vorleistungen

Das Projektteam hat bereits mögliche 3D-Frameworks ausgelotet und damit Beispiel-Applikationen erstellt. Ausserdem haben die meisten Projektmitgliedern Erfahrungen mit der Programmierung von Lego Robotern gesammelt und sich einen Überblick über existierende RCX-Emulatoren verschafft.

4 Ziele und Lösungen

4.1 Zielvorstellungen

Im Rahmen der Projektarbeit wollen wir das Wissen erarbeiten, um eine 3D-Simulation für Lego Roboter erstellen zu können. Da es sich um eine Simulation handelt, müssen wir festlegen, welche Funktionen wir simulieren wollen und welche Technologien wir dafür einsetzen. Die Simulation kann grob in 6 Teile aufgeteilt werden:

- Virtuelle 3D Landschaften
- Virtuelle Roboter, die sich in der Landschaft bewegen können und mit ihr interagieren
- Programme, die für Lego Roboter geschrieben werden und das Verhalten der Roboter definieren
- Eine Schnittstelle zwischen den Lego Roboter Programmen und der 3D Landschaft. Über sie können die Roboter gesteuert und Sensor-Daten übermittelt werden.
- Einen Editor um eine 3D Landschaft erstellen zu können
- Einen Editor um neue Roboter definieren zu können.

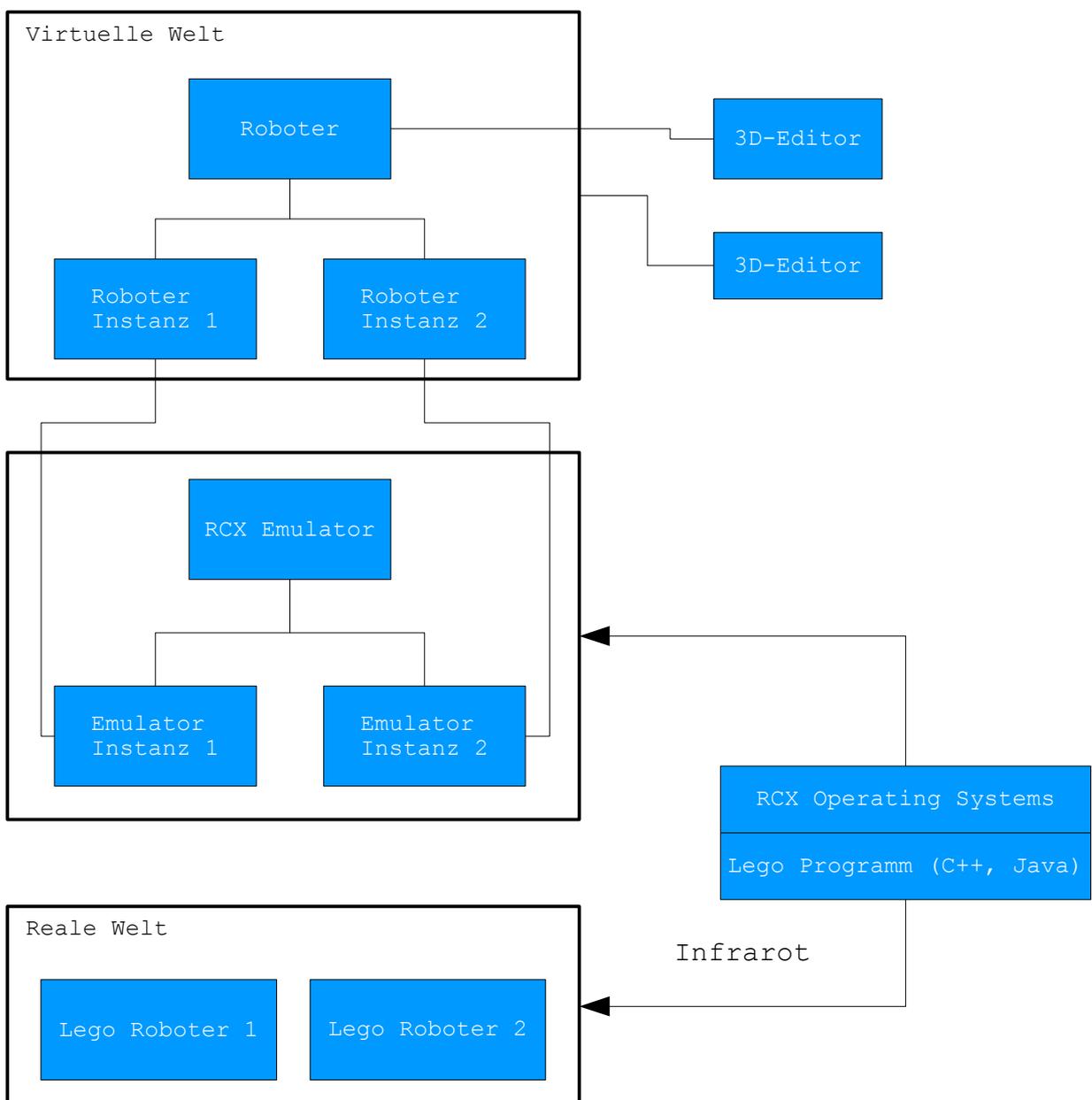
In unserer Projektarbeit wollen wir eine einfache 3D Welt erstellen und einen virtuellen Roboter mit einfachen Sensoren entwerfen. In einem weiteren Schritt wollen wir bestehende RCX-Emulatoren testen, um abzuklären, ob einer dieser Emulatoren sich als Schnittstelle zwischen der virtueller 3D Welt und bestehender RCX-Programmen einsetzen lässt.

Der Vorteil eines Emulators besteht darin, dass verschiedenste RCX-Betriebssysteme eingesetzt werden könnten, was die Akzeptanz unter den Anwendern sicher erhöhen würde. Falls der Aufwand zu gross oder die Integration zu komplex wäre, würde wir uns auf eine Schnittstelle zu einem populären Lego Betriebssystem (LeJOS) beschränken.

Zum Testen der Simulation werden wir einfache RCX-Programme erstellen und das Verhalten des virtuellen Roboters mit dem Verhalten eines echten Lego Roboters vergleichen.

Falls wir beschliessen sollten die Projektarbeit in unserer Diplomarbeit weiterzuführen, müsste eine erneute Evaluation stattfinden, um festzulegen ob der eingeschlagene weg ein brauchbares Resultat liefert. In die Diplomarbeit könnte auch die Implementation einfacher Editoren für die 3D welt und für virtuelle Roboter, aber auch komplexere Programme für die Roboter einbezogen werden.

4.2 Mögliche Lösungen





Wir haben uns ein kleines Übersichtsmodell aufgestellt, um so mögliche Lösungen visuell besser erfassen zu können. Drei Teilgebiete stechen in der obigen Grafik hervor. Die reale Welt, die virtuelle Welt sowie der Emulator. Evident ist, dass es uns die Emulation ermöglicht irgend ein "RCX Operating System" zu verwenden. Auf die Simulation hat dies keine Auswirkung.

5 Mittelbedarf

Schätzung über den Bedarf an Sachmitteln, Personal und Ausbildung.

5.1 Bewertung der Sicherheits- und Datenschutzaspekte

Die Projektarbeit ist ein Forschungsprojekt und ergibt keine Lösung für den Kunden, das heißt dass Sicherheitsaspekte im Rahmen des Forschungsprojekts keine Rolle spielen.

5.2 Sachmittel

- 3 Arbeitsplätze mit PC ausgerüstet
- Microsoft Visual Studio .NET
- Modeling Software, Maya oder/und Milkshape
- Diverse Linux Open Source Software
- Lego MindStorms Basispakete

5.3 Personal

- 3 Studenten
- 1 Projektbetreuer

5.4 Ausbildung

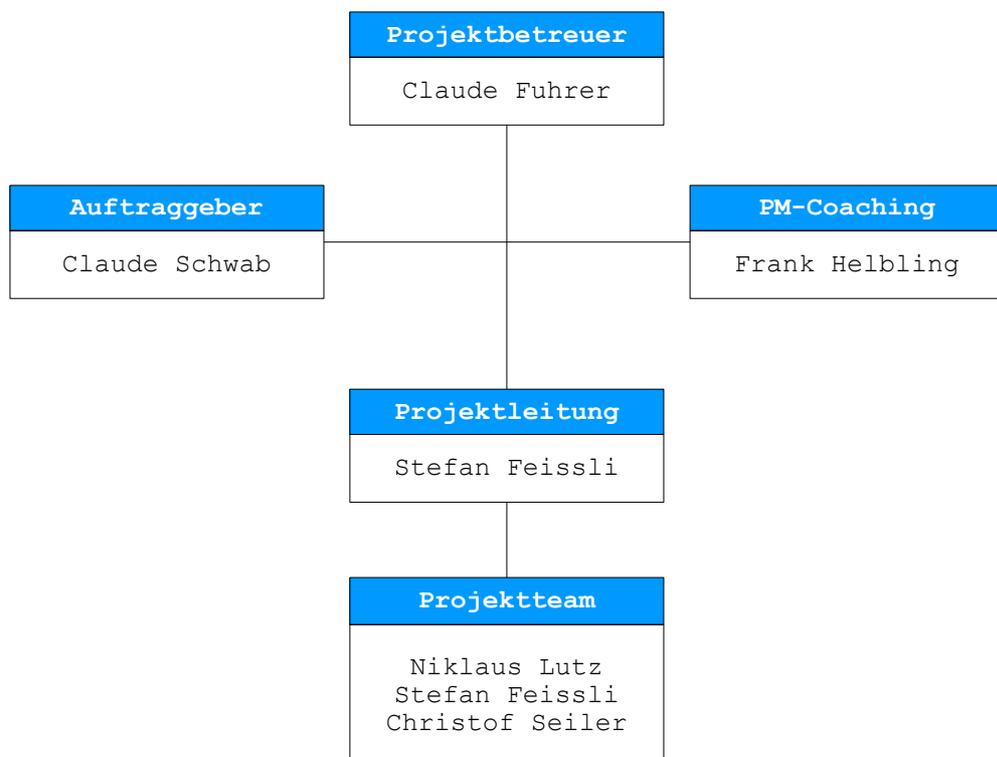
Da es sich um ein Projektarbeit handelt, ist das ganze Projekt zugleich auch Ausbildung. Da es sich um eine 3D-Anwendung handelt, sollten Basiskenntnisse in diesem Bereich vorhanden sein.

6 Planung und Organisation

6.1 Projektorganisation

Rolle	Personen
Projektauftraggeber	Claude Fuhrer
Projektleiter	Stefan Feissli
Stellvertr. Projektleiter	Christof Seiler
Projektbetreuer	Claude Fuhrer
Betreuer Projektführung	Frank Helbling
Realisierungsteam	Lutz Niklaus, Stefan Feissli, Christof Seiler

Tabelle 1: Projektorganisation





6.2 Termine

Folgende Termine sind von der HTI-Biel vorgegeben:

Termine	Datum
Projektstart	21. Februar 2005
Projektantrag	11. März 2005
Projekt-Handbuch & Projektplan	15. April 2005
Projektantrag & Abschlussbericht	17. Juni 2005

Tabelle 2: Termine

6.3 Prioritäten

1. 3D-Model einer Umgebung, mit Besichtigungsfunktion. Erfahrungen sammeln mit Kamera/Blickwinkel, Kollisionen/Interaktion mit Umgebung, „physikalische“ Eigenschaften ausloten.
2. 3D-Model des Lego Roboters erstellen.
3. Roboter in die Umgebung setzen, ihn bewegen könne, mit Kollisionen Umgehen (einfacher Sensor-Output)
4. Dem Roboter Programme übergeben, die mit der 3D-welt interagieren.



7 Wirtschaftlichkeit

7.1 Kosten

7.1.1 Personalkosten

Personal	Total Stunden	Stundenansatz sFr.	CHF
Niklaus Lutz	300	150	45'000
Stefan Feissli	300	150	45'000
Christof Seiler	300	150	45'000
Claude Fuhrer	40	200	8'000
Frank Helbling	20	200	4'000
Total	960		147'000

Tabelle 3: Personalkosten

7.1.2 Sachkosten

Sache	Anzahl	Preis/Stück	CHF
Benutzung Infrastruktur	1	4'000	4'000
PC	3	4'500	13'500
Softwarelizenzen	3	800	2'400
Total			19'900

Tabelle 4: Sachkosten

7.1.3 Gesamtkosten

Teilkosten	CHF
Personalkosten	147'000
Sachkosten	19'900
Nettokosten	166'900
Risikozuschlag 10%	16'690
Total Projektkosten	183'590
Gewinn 0%	0
Bruttokosten	183'590

Tabelle 4: Gesamtkosten



7.2 Nutzen

Es handelt sich bei dieser Projektarbeit um eine Art Forschungsprojekt ohne direkten finanziellen Nutzen. Für Entwickler/Designer von Lego MindStorms könnte diese Simulation jedoch von Interesse sein. Durch die Realisierung dieses Projektes wird es möglich sein, das Modell und sein Verhalten zuerst am Bildschirm zu testen, bevor es gebaut wird. Kosten und Zeit werden somit eingespart.

8 Konsequenzen

8.1 Auswirkung

Die Projektarbeit soll aufzeigen mit welchen Mitteln wir die Ziele für die Diplomarbeit erreichen können und welche für die Diplomarbeit überhaupt realisiert werden können. Des Weiteren sollten in der Projektarbeit die grundlegenden Teile der geplanten Lösung realisiert werden. Die Projektarbeit bildet in dem Sinne die Basis für die Zielsetzung der Diplomarbeit.

8.2 Bei Nichtrealisierung

Bei Nichtrealisierung des Projekts fehlen die Grundlagen für die Realisierung, der im Rahmen der Diplomarbeit geplanten Lösung. Der Zeitrahmen für die Diplomarbeit ist nicht ausreichend um die Grundlagen in den verschiedenen Teilbereichen zu erarbeiten oder alle Teile der Lösung von Grund auf neu zu entwickeln. Nichtrealisierung des Projekts würde nur eine sehr beschränkte Realisierung der als Diplomarbeit geplanten Lösung nach sich ziehen.

8.3 Bei verspäteter Realisierung

Der Abgabetermin ist durch die HTI-Biel festgelegt und kann nicht verschoben werden. Eine verspätete Realisierung ist ausgeschlossen.



8.4 Auswirkung auf Schnittstellen zu anderen Systemen

Im Rahmen des Projekts werden keine Schnittstellen zu anderen Systemen definiert oder verändert.

8.5 Qualitätsverbesserung

Ziel der Diplomarbeit ist es, die Interaktion von Lego Mindstorm Robotern mit ihrer Umwelt zu simulieren. Durch diese Simulation wird es möglich sein das Verhalten von Lego Robotern in einer vordefinierten oder selbst gestalteten Umgebung zu testen. Zudem hat die Simulation den Vorteil, dass Fehler in den von Anwendern geschriebenen Programmen für Roboter, einfacher zu finden sind, da die Emulation des RCX-Computers auch die Möglichkeit bietet zu jedem Zeitpunkt den genauen Programm-Zustand zu analysieren.

Ausserdem ermöglicht die Simulation Anwendern, die kein oder nicht ausreichend viele RCX-Computer besitzen, ihre Programme zu testen. Die Simulation spricht aber auch Menschen an, die mit dem Gedanken spielen, sich einen Lego Mindstorm Bausatz zu kaufen, aber nicht sicher sind, ob ihnen die Programmierung und das Spielen mit Roboter gefallen wird. Da unsere Simulations-Applikation frei erhältlich sein wird, fallen keine Anschaffungskosten an, so dass diese Hürde auf dem Weg zum Roboter Programmierer wegfällt.

8.6 Risikobeurteilung

Die geringe Planungserfahrung des Projektteams stellt ein mögliches Risiko dar. Es kann sein, dass der aufzubringende Aufwand zu niedrig eingeschätzt wurde. Auch fehlt die praktische Erfahrung im Bereich der Computer Grafik. So ist der zeitliche Aufwand für die Modellierung der virtuellen Welten nur sehr schwer einzuschätzen.

Da wir als Schnittstelle zwischen dem virtuellen Raum und den RCX-Programmen einen Emulator einsetzen wollen, besteht die Gefahr, dass sich bestehende Open Source RCX-Emulatoren nur schwer in unsere Applikation integrieren lassen oder der Funktionsumfang nicht unseren Ansprüchen genügt.

Eine weitere Hürde unseres Projektes wird die angestrebte Plattform-Unabhängigkeit sein. C++ und das Ogre 3D-Framework sind für Windows, MacOS X und Linux erhältlich, solange wir uns an Standard C++ halten wird dies kein allzu grosses Problem sein, es kann aber immer wieder vorkommen, dass gewisse Teile vom Betriebssystem abhängig sein müssen, z. Bsp. die Fenstermanager.



8.7 Ausweichmöglichkeiten

Sollten wir uns im Aufwand massiv überschätzt haben, müssten wir verschiedene Teilaufgaben vereinfachen. Falls der Aufwand für die Erstellung der 3D Landschaft oder des virtuellen Roboters unsere Erwartung übertrifft, können wir uns darauf einigen als Landschaft eine einfache flache Ebene und einen Würfel für den Roboter mit einem bekannten 3D-Designer (wie Maya oder 3D Studio Max) zu erstellen. Auf der Homepage von Ogre 3D gibt es Tutorials, die den Import solcher 3D-Modelle detailliert beschreiben und erklären.

Wenn alle Open Source RCX-Emulatoren nicht unseren Erwartungen entsprechen, besteht die Möglichkeit, den Emulator-Teil wegzulassen, und eine einfache Schnittstelle zu einem populären Lego Betriebssystem (z.B. LeJOS) zu implementieren. Auch dieser Task kann sehr viel Aufwand in Anspruch nehmen, so dass wir die Schnittstelle gegebenenfalls auf ein Minimum reduzieren müssten.

Da es ab einem gewissen Zeitpunkt nicht mehr möglich sein wird (da gewisse Teile bereits implementiert sein werden), uns von C++ und Ogre 3D abzuwenden, um die Plattform-Unabhängigkeit zu wahren, besteht die Möglichkeit uns auf eine Zielplattform zu konzentrieren. So stellen wir sicher, schlussendlich ein funktionierendes Produkt abliefern zu können.

9 Antrag

9.1 Bisherige Entscheide

Die Ergebnisse dieses Projektes sollen im Rahmen der Diplomarbeit weiterverwendet werden. Während der Projektarbeit werden die Phasen Initialisierung, Voranalyse und Konzept durchlaufen und abgeschlossen.

9.2 Formulierung des Projektantrags

Wir stellen den Antrag das Projekt „Jorge“ durchzuführen und für die Voranalyse freizugeben.