
Abschlußbericht



Im Rahmen der Veranstaltung:
Dezentrale Systementwicklung am Beispiel GNU/LINUX
an der TU-Berlin – Sommersemester 2002

MICHAEL GLÄSSEL, MATTHIAS ERCHE, JÖRG KÜSTER

22. September 2002

Zusammenfassung blah ...

Inhaltsverzeichnis

1	Einleitung	4
1.1	Der Projektkontext	4
1.1.1	Dezentrale Systementwicklung am Beispiel GNU/Linux	4
1.1.2	Das Projekt Mumie	4
1.2	Ziele	5
2	Organisation des Gesamtprojekts	8
2.1	Gruppenaufteilung	8
2.2	Vorträge	8
2.3	Workshops	8
3	Muminav	9
3.1	Lizenz	9
3.2	Entwicklungsumgebung	9
3.2.1	Entwicklungswerkzeuge	9
3.2.2	Projekthoster	10
3.2.3	Probleme und ihre Lösungen	10
3.3	Verwendete Komponenten und Ressourcen	10
4	Beurteilung des Projekts	13
4.1	Diskussion des Status quo	13
4.2	Einsatzfähigkeit	13
4.3	Ausblick	13

1 Einleitung

1.1 Der Projektkontext

Für eine moderne mathematische Ausbildung von Ingenieuren ist mittlerweile nicht nur der Umfang des erworbenen Wissens wichtig, vielmehr muss neben der reinen Vermittlung von Fachwissen Wert auf das Verständnis von Zusammenhängen gelegt werden. Heutzutage ist es durch immer schneller anwachsendes Wissen mehr denn je erforderlich, dass sich Ingenieure einem ständigen Lernprozess unterziehen. Die Unterstützung dieses Prozesses durch zeitgemäße, multimediale Technologien ist dabei nur folgerichtig. Es werden neue Möglichkeiten eröffnet, Zusammenhänge darzustellen, die Kommunikation zwischen Dozent und Lernenden und das selbstständige Aneignen von Lehrstoff zu fördern.

So haben sich vier deutsche Universitäten im Projekt „Multimediale Mathematik-ausbildung für Ingenieure“¹ [MUMIE] zusammengeschlossen, um die Idee von einer „modernen Mathematikausbildung“ zu verwirklichen.

1.1.1 Dezentrale Systementwicklung am Beispiel GNU/Linux

Das Projekt „Mumie“ soll nach aktuellem Stand unter einer Open Source-Lizenz entwickelt und veröffentlicht werden. Die Zusammenarbeit zwischen den Universitäten impliziert eine dezentrale Entwicklung, an der auch das Projekt „MUMINAV“ beteiligt ist. Das bei der Entwicklung entstehende Produkt wird nicht zur „Insellösung“, da es in ein bestehendes Projekt integriert wird.

Eine Aussicht auf die Verwendung schon existierender Software besteht unter Umständen bei der Realisierung der Datenschnittstelle. So gehen erste Überlegungen in die Richtung, schon bestehende XML-Software zu verwenden, um die vom Server übertragenen Daten auszuwerten.

1.1.2 Das Projekt Mumie

Das Projekt wird vom Bundesministerium für Bildung und Forschung gefördert. Es soll eine WWW-basierte, modulare Umgebung entwickelt werden, welche sowohl den Lernen-

¹kurz: MUMIE

den als auch Dozenten eine grafisch ansprechende und leicht zu bedienende Oberfläche bietet. So wird einerseits durch die Visualisierung mathematischer Inhalte und dem interaktiven Umgang mit der Mathematik die Motivation gefördert. Zusätzlich werden einige lehr- und lernbegleitende Tools angeboten:

- Darstellung mathematischer Inhalte mit interaktiver Multimedia-Unterstützung
- Stoffnachbereitung, Wiederholungsunterstützung angeleitete und kommentierte Übungsaufgaben
- Selbstkontrolle durch individuelle Testumgebungen
- Einführung in mathematische Standardsoftwarepakete
- individuelles Trainingscenter weiterführender Inhalte
- Informationsplattform in der und vile und in die der es auch noch kein wer, was, wie, wo
- Kommunikations- und Austauschangebote

Die ursprüngliche Idee hierbei ist, mathematische Inhalte darzustellen. Die Umgebung soll allerdings so gestaltet werden, dass sie auch in anderen Disziplinen genutzt werden kann. Weitere Schwerpunkte liegen auf der Individualisierung der Oberfläche und einer intelligenten Benutzerführung, die verhindert, dass man in der inhaltlichen Fülle des Gesamtangebotes die Übersicht verliert.

1.2 Ziele

Der Dozent soll die Möglichkeit haben, aus einer Auswahl an Elementen ², die in einer Datenbank gespeichert sind, die Zusammenstellung für einen Kurs zu erstellen. Dies soll durch einfaches Ziehen und Ablegen von Elementsymbolen auf einer Zeichenfläche erfolgen. Zusammenhänge zwischen Elementen werden dabei durch Verbindungslinien dargestellt, die der Verfasser positionieren kann. Zusätzlich kann es zu den Elementen Subelemente ³ geben. Es entsteht durch die Zusammenstellung eine Repräsentation der mathematischen Zusammenhänge mittels eines gerichteten Graphen. Der Dozent kann zur Darstellung des Kursverlaufes eine Art „roten Faden“ festlegen, der allerdings nicht entlang der angelegten Verbindungen laufen muss ⁴.

Die Aufgabe für das Projekt MUMINAV besteht darin, dieses vorgegebene Netz mit Hilfe eines Java-Applets in einem Navigationsframe darzustellen und somit den Zugriff

²Motivation, Definition, Theorem, Lemma, Algorithmus, Anwendung

³Herleitung, Beweis, Motivation, Bemerkung, Historisches, Visualisierung, Beispiel, Tabelle

⁴Aktueller Stand, Änderungen möglich

auf die Inhalte zu ermöglichen. Die Darstellung der Elemente erfolgt durch unterschiedlich farbige, leicht dreidimensional angedeutete Kästen. Subelemente werden nach vorgeschriebenen Regeln an diesen Kästen angeordnet und sind standardmäßig nicht mit Unterscheidungsmerkmalen versehen.

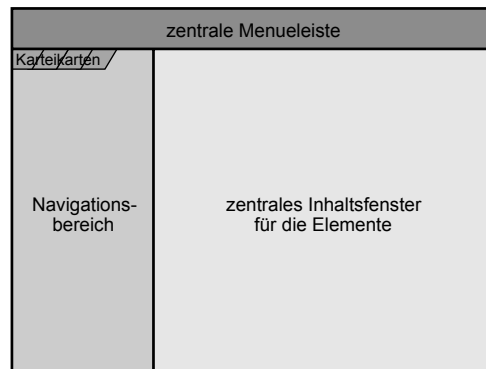


Abbildung 1.1: Layout der Hauptansicht

Die Abbildung des logischen Netzes geschieht, wie vorher beschrieben, durch Verbindungslinien zwischen den Elementensymbolen. Es wird ein „roter Faden“ entsprechend der linearen Anordnung der Kursinhalte gelegt.

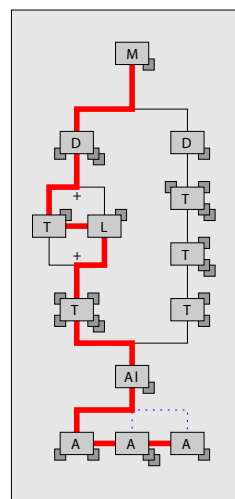


Abbildung 1.2: Kurspfade und roter Faden

Die Darstellung soll auf mehrere Maus-Aktionen reagieren. Wird die Maus über ein Element bewegt, soll dieses vergrößert dargestellt werden und die Subelemente werden unterscheidbar durch Beschriftung. Bei Klick mit der linken Maustaste soll der Inhalt

des jeweiligen (Sub-)Elementes im zentralen Inhaltsfenster dargestellt werden. Dieselbe Aktion auf der mittleren Maustaste führt zu einem Öffnen des Inhalts in einem externen Fenster und schließlich ist es zukünftig vorgesehen, mit der rechten Maustaste eine Liste von Optionen anzubieten.

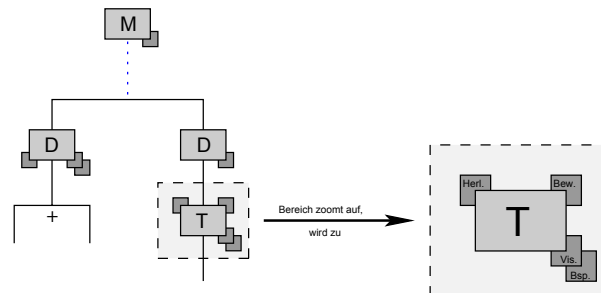


Abbildung 1.3: Detailansicht bei Zoom

2 Organisation des Gesamtprojekts

Regelmäßige Meetings/Treffen

2.1 Gruppenaufteilung

2.2 Vorträge

2.3 Workshops

3 Muminav

Beispiel Programm, welches zeigt wie man Muminav einbinden kann.
relevante andere Projekte - hoch spezifizierte Anforderungen

3.1 Lizenz

Vorgabe war, die gesamte Entwicklungsarbeit als Open Source Projekt durchzuführen. Dies schließt ein, die Projektergebnisse unter einer Open Source Lizenz [OSI2002] zu veröffentlichen.

Da wir unser Projekt in Zusammenarbeit mit der Mumie-Gruppe entwickeln, mussten wir uns im Vorfeld mit Ihnen auf eine Open Source Lizenz verständigen, welche in das Gesamtprojekt MUMIE integrierbar ist.

Die GPL [GPL1991] (GNU General Public License)
LGPL [LGPL1999].

3.2 Entwicklungsumgebung

3.2.1 Entwicklungswerkzeuge

Für das Projekt wurden eine Reihe von Entwicklungswerkzeugen und Technologien verwendet, welche für Open-Source-Projekte charakteristisch sind:

- **Mailinglisten** um mit den Entwicklern und allen, die sonst noch Interesse an dem Projekt haben zu kommunizieren. Wobei damit auch gleichzeitig eine Dokumentation des Projektverlaufs über die Mailinglisten-Archive entsteht.
- **CVS** Concurrent Version System [FOGEL2000]. Hat uns ermöglicht dezentral an den selben Quelldateien zu arbeiten. Über die CVS-Log Einträge lässt sich auch nachträglich die Entwicklungs-Historie nachvollziehen.
- **eMail** Als standard-Medium zum direkten persönlichen Informationsaustausch.
- **Instant-Messaging** hat bei Arbeiten, die ein hohes Maß an Absprachen bedürfen, nicht die Nachteile, die ein asynchrones Medium wie eMail und Mailinglisten haben.

- **Projekt-Homepage** Unter der URL: <http://muminav.berlios.de> haben wir eine Projekt-Homepage angelegt, die die Öffentlichkeit und die Teilnehmer des Projekts mit allgemeinen Informationen versorgt.
- **Newsgroups** Hier haben wir uns Anregungen und Informationen für die Planung des Projekts und bei Problemen, die in der täglichen Arbeit auftraten besorgt.

Für die Entwicklung der Quellen haben wir folgende Software eingesetzt.

- **JavaTM2 Platform, Standard Edition**
- **Apache Ant** des JAKARTA PROJEKT von Apache, als make-Tool.

Leider ist der Javacompiler, den wir eingesetzt haben nicht als Open-Source veröffentlicht worden. Aus Kompatibilitätsgründen mussten wir uns jedoch für diesen Compiler entscheiden, damit das Applet unter möglichst unterschiedlichen Umgebungen läuft ohne, dass der Benutzer weitere Software installieren muss.

Als Alternativen zum Java-Compiler von Sun wäre im Open-Source-Bereich z. B. JIKES von IBM [JIKES], der an der Universität von Boston entwickelte ESPRESSO [ESPR] oder der GCJ der FREE SOFTWARE FOUNDATION [GJC], denkbar. Da sich aber jeder bei uns die Java Quellcodes herunterladen kann, bleibt es jedem selbst überlassen, welchen Compiler er einsetzt [J2SE] .

3.2.2 Projekthoster

Bei der Wahl des Projekthosters kam für uns SOURCEFORGE *nicht* in Frage, da sich im Laufe der Zeit die Lizenz-Politik [SFORGE] des Hosters immer mehr gegen die Ideale der Open-Source-Gemeinde richten. Das Savannah Projekt der Free Software Foundation war zu Projektbeginn leider noch nicht genügend weit entwickelt.

Aus diesen Gründen haben wir uns für den Berliner Projekthoster BERLIOS [BerliOS] entschieden, der uns mit den meisten Diensten unterstützen konnte.

3.2.3 Probleme und ihre Lösungen

XML-Parser mitliefern - Größe - Entscheidung für Java 1.4 und Mozilla

3.3 Verwendete Komponenten und Ressourcen

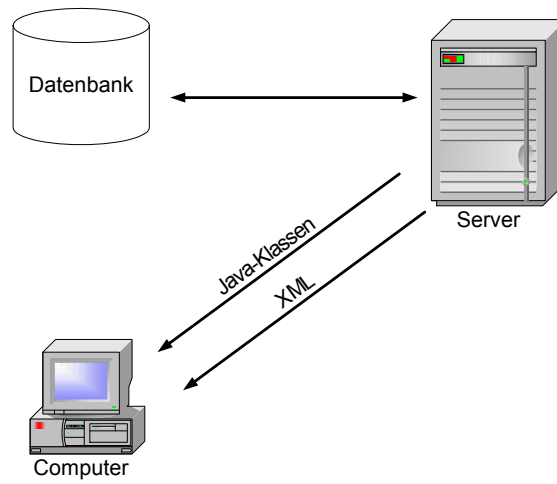


Abbildung 3.1: Kommunikation: Informationsfluss

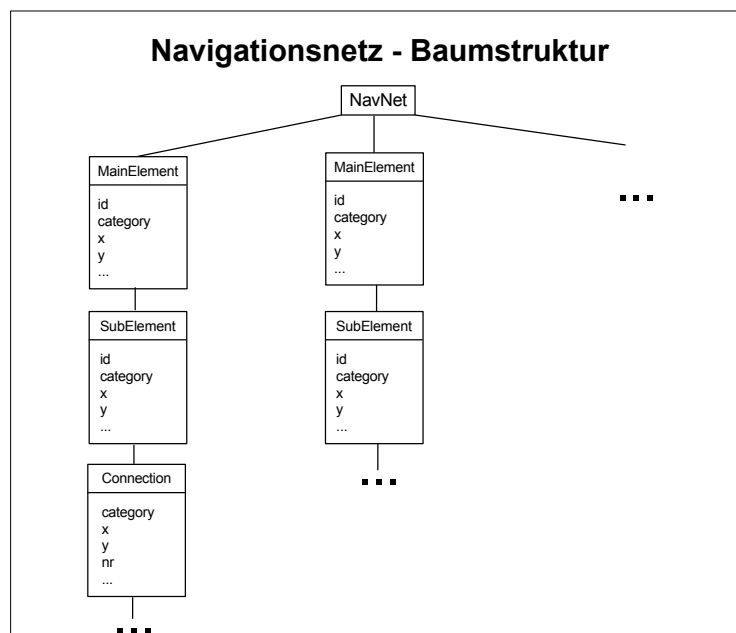


Abbildung 3.2: Baumstruktur

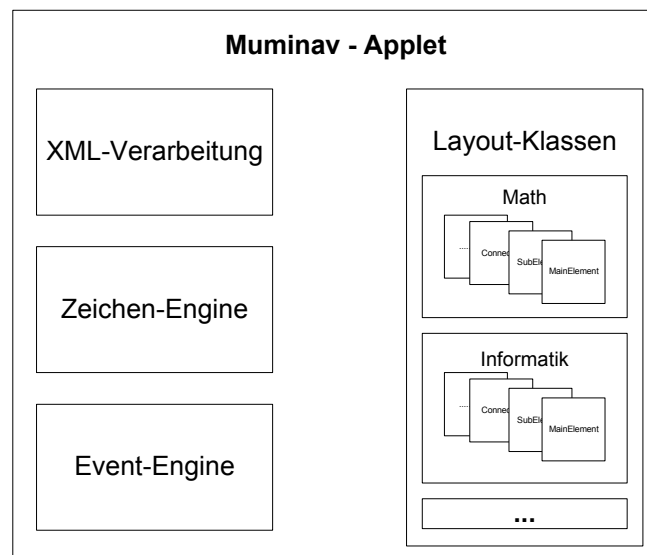


Abbildung 3.3: Aufbau: Hauptteile von Muminav

4 Beurteilung des Projekts

4.1 Diskussion des Status quo

4.2 Einsatzfähigkeit

MUMINAV wurde als hoch spezialisierte Softwarekomponente für den Einsatz im Gesamtprojekt MUMIE ins Leben gerufen. Nichts desto trotz haben wir großen Wert darauf gelegt, die Software so allgemein wie möglich zu gestalten damit sie für möglichst viele andere Open-Source-Projekte interessant wird. Letztendlich wurde, durch den Einsatz von Skins und XML als Datenformat, aus der Aufgabe, ein Java-Applet zu programmieren, welches Navigationsnetze für Mathematische Kurse im Internet dynamisch erzeugt, ein Softwaremodul, welches für die Darstellung jeglicher graphischer Darstellungen geeignet ist. Zum Beispiel ließen sich leicht Skins erstellen, mit denen man Flowcharts, E/R Diagramme oder sogar UML darstellen kann.

Zusätzlich ist die Software so ausgelegt, dass sie nicht auf Applets beschränkt ist. Durch das Packaging in ein Java-Swing-Panel lässt es sich genau so gut in eine Java-application einbauen.

4.3 Ausblick

Für die Zukunft des Projekts haben wir uns vorgestellt, den Abstraktionsgrad weiter zu steigern, damit es immer leichter wird, seine eigenen Skins zu entwerfen.

Angedacht ist auch ein einfacher Skin-Editor, mit dem man sich bequem per Maus seine eigenen Skins entwerfen kann.

Interessant wäre es auch, unsere Quelle, die wir bisher auf der Java-Plattform von SUN entwickelt und getestet haben, auf die Kompatibilität zu Open-Source-Kompilern zu testen und gegebenenfalls anzupassen. Damit könnten wir dann auch den Bytecode unter einer entsprechenden Lizenz veröffentlichen.

Wir werden auch versuchen eine kleine Community zum leben zu erwecken, die gegenseitig Skins austauscht. Dabei könnte man die MUMINAV-Homepage zu einem öffentlich zugänglichen Skin-Archiv erweitern, in das jeder seine Skin einbringen und bei Bedarf von der Kreativität anderer profitieren kann.

Was eigentlich gar nicht erwähnt werden müsste: MUMINAV, wie auch jedes andere Open-Source-Projekt, ist mit der Fertigstellung der ersten Version nicht abgeschlossen. Vielmehr wird es, solange es aus der Open-Source-Community Interesse und Anregungen gibt, sich weiterentwickeln. Sei es durch entdeckte Bugs oder Vorschläge für neue Features, die durch uns oder Mitglieder aus der Open-Source-Community in das Projekt einfließen.

Literaturverzeichnis

- [MUMIE] DOMINIK EBERLEIN, ANDREAS HOFFMANN: Die Mumie - Projekthomepage. <http://www.mumie.org/>
- [OSI2002] OPEN SOURCE INITIATIVE: The Approved Licenses. <http://www.opensource.org/licenses/> 2002
- [GPL1991] FREE SOFTWARE FOUNDATION, INC: The GNU General Public License (GPL). <http://www.opensource.org/licenses/gpl-license.php> 1991, version 2,
- [LGPL1999] FREE SOFTWARE FOUNDATION, INC: GNU Lesser Public License. <http://www.opensource.org/licenses/lgpl-license.php> 1999, version 2.1,
- [FOGEL2000] KARL FOGEL: Open Source Development with CVS. <http://cvsbook.red-bean.com/> 2000, version 1.2
- [JIKES] IBM: Jikes Official Homepage. <http://oss.software.ibm.com/developerworks/opensource/jikes/> 2002
- [ESPR] PROF A J KFOURY: Java Espresso. <http://types.bu.edu/Espresso/JavaEspresso.html> 2002
- [GJC] FREE SOFTWARE FOUNDATION, INC: The GNU Compiler for the Java Programming Language. <http://gcc.gnu.org/java/> 2002
- [J2SE] PROF A J KFOURY: Java Espresso. <http://types.bu.edu/Espresso/JavaEspresso.html> 2002
- [SFORGE] SOURCESORGENET: Terms and Conditions of Use. http://sourceforge.net/docman/display_doc.php?docid=6048&group_id=1 2002
- [BerliOS] BERLIOS: BerliOS Site Documentation. <http://developer.berlios.de/docs/site/> 2002