

Abschlussbericht zum "Fellowship für Innovationen in der digitalen Hochschullehre"

FLIPPEDOS – Individualisierung des Unterrichts mit Hilfe von virtuellen Arbeitsplätzen

Preisträger

Dr. rer. nat. Stefan Lankes, Akademischer Oberrat
RWTH Aachen University, E.ON Energy Research Center
Institute for Automation of Complex Power Systems
Mathieustr. 10, 52074 Aachen
E-Mail: slankes@eonerc.rwth-aachen.de
URL: <http://www.acs.eonerc.rwth-aachen.de>

Zielveranstaltung

Betriebssysteme (Operating Systems)
Pflichtveranstaltung der Studienrichtung *Technische Informatik*
Bestandteil des Bachelor-Studiengangs *Elektrotechnik, Informationstechnik und Technische Informatik*



1 Zielsetzung

Die Vorlesungen *Grundgebiete der Informatik 4*¹ und *Betriebssysteme*² sind Pflichtveranstaltungen der Studienrichtung *Technische Informatik* des Bachelor-Studiengangs *Elektrotechnik, Informationstechnik und Technische Informatik*, welcher von der Fakultät für Elektrotechnik und Informationstechnik angeboten wird. Im Rahmen dieser Vorlesungen wird ein tiefgehendes Verständnis für Systemsoftware vermittelt. Die erworbenen Kenntnisse unterstützen die Studierenden bei der Entwicklung von Hochleistungsrechnersystemen sowie echtzeitfähigen und/oder sicherheitskritischen Anwendungen. So wird beispielsweise vermittelt, wie ein Betriebssystem die Ressourcen eines Rechners schützt und wo hierbei die Stärken und Schwächen verschiedener Betriebssystementwürfe liegen. Im Zeitalter des *Internet-of-Things* wird dies zunehmend bedeutender, um die Infrastruktur einer modernen Gesellschaft zu schützen.

Die Vorlesungen vermitteln – wie in der Standardliteratur [TB14, SGG09] – theoretische Konzepte und praktische Lösungsansätze. So wird in der Vorlesung *Betriebssysteme* schrittweise ein Lehrbetriebssystem entwickelt, anhand dessen grundlegende Entwurfskonzepte für Betriebssysteme erläutert werden, wobei jeder Meilenstein einem Vorlesungskapitel entspricht. Dieser Ansatz verbindet Forschung und Lehre auf direktem Weg, da ein direkter Bezug zu Forschungsprojekten (z. B. HermitCore³ [LPB16]) des Institutes for Automation of Complex Power Systems hergestellt wird.

Ein zentrales Problem solcher Lehrveranstaltungen stellt der unterschiedliche Wissensstand der Studierenden dar. Je nach privat oder schulisch entwickelten Vorkenntnissen, sind einige Studierende unter- oder überfordert. Dieses Problem wird in der Vorlesung *Betriebssysteme* durch die Alltagspräsenz von Betriebssystemen noch einmal verstärkt. Durch PCs, Laptops und Smartphones gehören sie zur meistbenutzten Software im Alltag. Eine Diskussion über das Für und Wider eines Betriebssystems wird von den Nutzern häufig emotional geführt. Ein Wechsel und die damit verbundene Einarbeitung wird meistens gescheut.

In der Forschung und der Lehre werden häufig UNIX-basierte Betriebssysteme als Arbeitsbasis gewählt, da diese meist offener und besser dokumentiert sind. Zudem besitzen sie im Bereich der eingebetteten und Server-Systeme eine hohe Akzeptanz. Allerdings ist der Wissensstand bei den Studierenden äußerst unterschiedlich und kann durch (studienbegleitende) Praktika kaum angeglichen werden, da diese andere Themenbereiche adressieren.

Die Ziele, welche das Projekt FLIPPEDOS hatte, können wie folgt zusammengefasst werden:

- Die Komplexität der Systemsoftware auf das nötige reduzieren und die grundlegende Prinzipien theoretisch aber auch praktisch zu vermitteln.

¹<http://www.gi4.rwth-aachen.de>

²<http://www.bs.rwth-aachen.de>

³<http://www.hermitcore.org>

- ❑ Unterschiedliche Kenntnisstände der Studierenden schnell anzugleichen, um eine Über- bzw. Unterforderung zu verhindern.
- ❑ Erlernen der kooperativen Entwicklung großer Softwaresysteme mit Maßnahmen zur Qualitätssicherung.
- ❑ Förderung des selbstständigen Arbeitens und des eigenständigen Denkens

2 Entwickelte Maßnahmen zur Lehrinnovation

Die Vorlesung Betriebssysteme basiert größtenteils auf dem Betriebssystem UNIX. Je nach Arbeitsweise der Studierenden ist beispielsweise die Installation einer virtuellen Maschine nötig, um dort ein UNIX-artiges Betriebssystem zu installieren und die praktischen Beispiele zu testen. Das Institute for Automation of Complex Power Systems hatte hierzu Leitfäden angeboten, um die Installation zu erleichtern. Erfahrungsgemäß ist diese Hürde sehr hoch und schreckt ab. Zudem sollte im Rahmen von FLIPPEDOS das Vermitteln und Testen einzelner Betriebssysteme-Konzepte in interaktive Lehrmaterialien integriert werden. Hierbei sollten die Lehrinhalte auf jedem Endgerät (Tablet, Laptop, PC) zu verwenden sein. So wurde angedacht, für die Vorlesungsinhalte anstatt dem klassischen, starren PDF-Format ein flexibles, interaktives eBook-Format zu verwenden (z. B. ePUB3). Die Lehrinhalte haben zum Ziel, die Durchführung eines Flipped Classroom Konzeptes zu ermöglichen. In der Vorhabensbeschreibung wurde bereits erwähnt, dass zur Erstellung der eBooks das Werkzeug AsciiDoc⁴ zu evaluieren ist. Bei der Evaluation wurde festgestellt, dass auch eBooks einen zu starren Aufbau besitzen. HTML5-basierte Webseite sind deutlich flexibler und können auch mit AsciiDoc schneller und komfortable erstellt werden. Daher wurden die entwickelten Lehrinhalte als Web-Seite den Studierenden zur Verfügung gestellt. AsciiDoc erwies sich hierbei als einfaches und flexibles Werkzeug, welches auch mathematische Formeln im \LaTeX -Format akzeptiert. Hierdurch war das Übertragen von existierenden Lerninhalten, die mit Hilfe von \LaTeX verfasst wurden, zum neuen Lernformat einfach zu realisieren.

Die Verwendung HTML-basierter Lehrmaterialien ermöglicht die Integration von Editoren. So können die Studierenden Beispielanwendungen direkt im Web-Browser editieren und testen. Hierdurch wurde der Einsatz von Spezialsoftware auf studentischer Hardware vermieden, was die Einstiegshürde des Unterrichtsfaches senkt. Allerdings stellt ein Browser keine ausreichende Infrastruktur zur Verfügung, um systemnahe Anwendungen zu compilieren und zu testen. Um dies zu ermöglichen, übertragen die Web-Seiten die Beispielanwendungen zu einem Server, der sie automatisiert compiliert und testet. Allerdings stellt ein klassischer Server einen Flaschenhals dar. Für die Vorlesung ist es wichtig, dass der angebotene Dienst mit der Anzahl der Nutzer skaliert. So kann die Anzahl der Studierende zwischen den Jahrgängen variieren und die Nutzerzahl wird zu Klausurzeiten deutlich ansteigen. Aus diesem Grund muss die

⁴<http://www.methods.co.nz/asciidoc/>

Rechenleistung des Server-Infrastruktur mit der Anzahl der Nutzer hoch- aber auch herunterskalierbaren.

Solche Problemstellungen werden mit Hilfe von Techniken aus dem Bereich des Cloud Computings gelöst. Cloud-Lösungen sind vor allem von Google und Amazon⁵ bekannt, die ihre Infrastruktur den Kunden als Dienstleistung (*Infrastructure as a Service*) zur Verfügung stellen. Eine ähnliche Lösung existiert an der RWTH Aachen, welche Open-Stack⁶ als Basis verwendet. Um skalierbare Dienste auf solchen Plattformen anzubieten, werden sogenannte Container erstellt, welche die benötigten Komponenten enthalten und über die Rechnerinfrastruktur automatisiert verteilt werden. Im Falle von FLIP-PEDOS werden solche Container mit Hilfe von Docker⁷ erstellt und mit Kubernetes⁸ verteilt. Die Container beinhalten alle Komponenten (z. B. Compiler), die zum Testen der Anwendung benötigt werden. Somit müssen die Studierenden nicht länger die Software auf ihren Rechnern installieren, sondern verwenden indirekt die Container, welche vom Dozenten bereitgestellt wurden. Dies vereinfacht auch die Arbeit des Dozenten, da alle Studierende das gleiche *Setup* / den gleichen Aufbau verwenden.

Die Abbildung 1 verdeutlicht die entwickelte Lösung graphisch. Für die Darstellung wurden die Logos der einzelnen Komponenten verwendet. Der Web-Server *Apache* läuft gegenüber dem Dienst, der die Anwendungen baut und testet, auf einem anderen System. Über den Web-Browser der Studierende werden die Zugriffe auf das System koordiniert. Alle *build*-Anfragen werden über einen sogenannte *Load Balancer* auf die Rechner verteilt, welche die Anfragen bearbeiten. Graphisch werden diese durch einen Wal dargestellt, der mehrere Container transportiert. Der Wal repräsentiert hierbei das Logo der verwendeten Container-Technologie.

Der gewählte Lösungsansatz ist nahezu beliebig skalierbar, so dass die Lösung auch für größere Kurse geeignet ist. Bei der Entwicklung der Lösung wurde großer Wert auf Sicherheit gelegt. So wird jegliche Kommunikation zwischen der Cloud-Infrastruktur, Web-Server und dem Web-Browser des Studierenden verschlüsselt durchgeführt. Zudem werden alle Tests in einem abgesicherten Bereich bearbeitet, so dass auch „böse“ Teilnehmer nicht die Möglichkeit haben, das System zu stören. Absolute Sicherheit kann zwar nicht garantiert werden, aber das Sicherheitsrisiko wurde auf ein Minimum reduziert. Zudem werden keine personenbezogenen Daten erhoben und protokolliert, um die Gefahr des Missbrauches auszuschließen.

Durch die Komplexität der entwickelten Lösung, hat sich der Entwicklungsprozess verzögert und wird erstmalig im vollen Umfang im aktuellen Semester (WS 2018/2019) verwendet. Beispielsweise sind interaktive Folien unter <https://os.rwth-aachen.de/slides/> zu erreichen. Einzelne Themen werden im umgekehrten Unterricht (Flipped Classroom) behandelt, wo die Vorbereitungsunterlagen entsprechend erstellt wurden. Die Web-Seite beinhaltet Schnittstellen zur Cloud-Infrastruktur, in der Beispiele com-

⁵Beispielsweise <https://aws.amazon.com/de/>

⁶<http://www.openstack.org>

⁷<https://www.docker.com>

⁸<https://kubernetes.io>

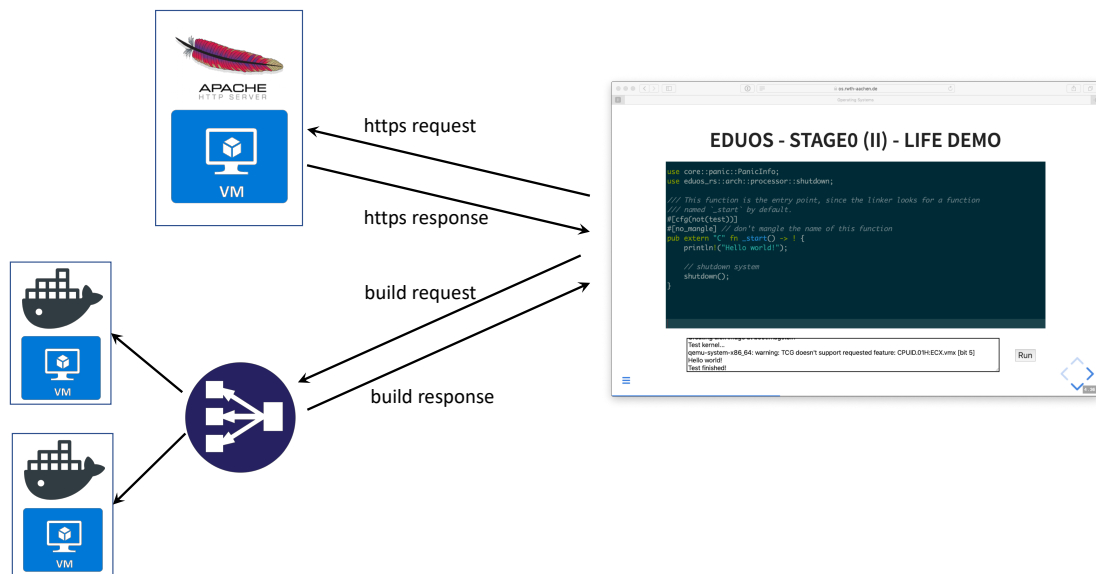


Abbildung 1: Schematischer Aufbau der entwickelten Lösung

piliert und getestet werden. Die Web-Seite ist direkt aus dem RWTH-Netz erreichbar. Alle anderen Nutzer müssen sich mit dem Login **rwth** und dem Passwort **os4ever#** authentifizieren.

Für weiterführende Themen wurde das Lehrbetriebssystem *eduOS* weiterentwickelt und die Einstiegshürde reduziert. So wird nun als Programmiersprache *Rust* anstatt *C* verwendet, die mehr von den technischen Details abstrahiert und somit einfacher zu verwenden ist. Die Studierenden können nun Ergänzungen zu *eduOS* entwickeln und als sogenannte *Pull Requests* in einem webbasierten Online-Dienst zur Softwareentwicklung einreichen. *Pull Requests* stellen bei OpenSource-Projekten Anfragen zur Integration neuer Komponenten dar. Diese werden begutachtet, um eine fortlaufend hohe Qualität des Software-Projekts zu gewährleisten, und nach positiver Begutachtung aufgenommen. Im Rahmen der Lehrveranstaltung werden die Einreichungen bewertet und die besten Studierenden prämiert. Hierdurch wird vermittelt, wie große Software-Projekte realisiert werden.

3 Finanzierung und Nachhaltigkeit von FLIPPEDOS

Das Projekt wurde wie geplant finanziert und durchgeführt. Die Tabelle 1 listet im Detail die verausgabten Mitteln auf. Für die Durchführung des Vorhabens FLIPPEDOS wurde ein Server-System angeschafft, um die Cloud-Infrastruktur der RWTH Aachen zu erweitern.

Position	Kosten
Server für die Cloud-Infrastruktur	10 943,24 €
Laptops & Tablets	6 561,66 €
Studentische Hilfskräfte	20 824,73 €
Wissenschaftliche Hilfskraft	10 696,47 €
Gesamtsumme	49 026,10 €

Tabelle 1: Verausgabete Finanzmitteln für das Projekt FLIPPEDOS

Für die Überarbeitung der Lehrmaterialien, den Aufbau des Infrastrukturdienstes, die Erweiterung des Werkzeugs AsciiDoc, um die Web-Seiten für das Testen der praktischen Beispiele zu erstellen, wurden studentische Hilfskräfte aus dem Bereich *Technische Informatik* sowie wissenschaftliche Mitarbeiter des Institutes for Automation of Complex Power Systems eingesetzt.

Zur Entwicklung und zum Testen der neuen Lehrmaterialien, sowie zur Durchführung der Präsenzstunden wurden Laptops und Tablets angeschafft. Eine klassische *Arbeitsplatzrechner-Infrastruktur* ist hierfür nicht geeignet, da sie dem Umfeld eines Hörsaals nicht entspricht.

Nach Projektabschluss soll die Pflege der entwickelten Lösungen weiterhin mit studentischen Hilfskräften gewährleistet werden. Der Arbeitsumfang wird deutlich geringer sein als bei der Entwicklung des Lehrkonzept, sodass dies aus den Haushaltsmitteln zu gewährleisten ist. Es ist auch zu erwarten, dass die Cloud-Infrastruktur weiterhin aus Haushaltsmitteln zu finanzieren sind, da das Institute for Automation of Complex Power Systems in diesem Bereich forscht und lehrt. Nur bei Erweiterung des Konzeptes für andere Vorlesungen müsste überprüft werden, ob dies mit der aktuellen finanziellen Ausstattung zu stemmen ist.

4 Erste Erfahrungen mit dem neuen Lernformat

Die neuen Lehrinhalte konnten noch nicht vollständig evaluiert werden, da sie voll umfänglich erst im aktuellen Wintersemester zum Einsatz kommen. Teilbereiche wurden aber schon in den vorherigen Semestern getestet und verwendet, sodass schon jetzt ersichtlich ist, dass die Studierende das Konzept gut annehmen.

Im ursprünglichen Antrag wurde angestrebt, die Lehrmaterialien als eBook zur Verfügung zu stellen und den plattformunabhängigen Editor *Atom* für die Vorlesung zu erweitern. Es hat sich herausgestellt, dass eBooks zu starr sind und Web-Seiten eine höhere Flexibilität anbieten. Durch die Integration eines Quellcode-Editors innerhalb der Web-Seiten wurde die Arbeiten am Editor *Atom* überflüssig. Das Bauen und Testen der Programme wurde wie im Antrag beschrieben auf einer Cloud-Infrastruktur realisiert. Die

entwickelte Infrastruktur ist erweiterbar auf alle Kurse, die eine Software-Infrastruktur (zumindest unterstützend) benötigen. Dies können Programmierkurse aber auch mathematische bzw. kommunikationstechnische Kurse sein, die MatLab oder ähnliche Werkzeuge zur Visualisierung verwenden. Die Einstiegshürde wird durch die Cloud-Infrastruktur reduziert. Hierdurch kann der Dozent schneller zum inhaltlichen Kern des Kurses kommen und diesen eingehender behandeln. Zudem könnte das Konzept für größere Praktika ausgebaut werden, da die Cloud-Dienste mit den Bedürfnissen mitwachsen können. Hierdurch besteht die Möglichkeit, die Größe der Rechenräume zu verkleinern, da mit Hilfe des *Flipped Classroom* Konzeptes die Anzahl der Präsenztermine reduzierbar sind.

Die Lehrmaterialien wurden in englischer Sprache verfasst, um die Internationalisierung der RWTH Aachen zu erleichtern. Dieser Wechsel ist bei den Studierenden gut angekommen.

Literatur

- [LPB16] LANKES, Stefan ; PICKARTZ, Simon ; BREITBART, Jens: HermitCore – A Unikernel for Extreme Scale Computing. In: *Proceedings of the International Workshop on Runtime and Operating Systems for Supercomputers (ROSS 2016), held in conjunction with 25th International ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC 2016)*. Kyoto, Japan, jun 2016
- [SGG09] SILBERSCHATZ, Abraham ; GALVIN, Peter B. ; GAGNE, Greg: *Operating System Concepts*. 8th edition. John Wiley & Sons., 2009
- [TB14] TANENBAUM, Andrew S. ; BOS, Herbert: *Modern Operating Systems*. 4th edition. Pearson Education International, 2014