

CSCL unplugged – Nomadisches kooperatives Lernen

Matthias Hellweg, Stephan Lukosch und Martin Rasel
Fakultät für Mathematik und Informatik
FernUniversität in Hagen
58084 Hagen

{matthias.hellweg, stephan.lukosch, martin.rasel}@fernuni-hagen.de

Abstract: Die FernUniversität in Hagen verwendet die webbasierte kollaborative Lernplattform CURE, um verschiedene kollaborative Lernszenarien zu unterstützen, z.B. Seminare oder Praktika. In diesen Szenarien bilden Studierende auf Aufforderung der Lehrenden Lerngruppen und arbeiten gemeinsam an einer Aufgabe. Durch Befragungen von Studierenden, die CURE im Rahmen von Lehrveranstaltungen genutzt haben, wurde deutlich, dass eine nomadische Nutzungsmöglichkeit von CURE von großem Interesse ist, so dass auch Zeiten der Mobilität zum Lernen genutzt werden können. In diesem Artikel beschreiben wir, wie wir CURE für eine nomadische Nutzung erweitert haben und welche Anforderungen dabei zu berücksichtigen waren.

1 Einleitung

Die klassische Lehre und Weiterbildung im Rahmen von Vorlesungen, Seminaren oder Praktika wird heutzutage in zunehmendem Maße durch rechnergestützte Lehrmethoden unterstützt. Wenngleich diese den sogenannten *Face-to-Face-Unterricht* sicherlich nicht vollständig ersetzen können, so sind sie doch eine wertvolle Ergänzung, die besonders im Hinblick auf selbstbestimmte Weiterbildung und kooperatives Lernen großes Potential besitzt. Insbesondere im Fernstudium ermöglicht die rechnergestützte Lehre es Studierenden, gemeinsam zu lernen. Aus diesem Grund wurde an der FernUniversität in Hagen die webbasierte Plattform *CURE* [HSH⁺03] zur Unterstützung des rechnergestützten kollaborativen bzw. kooperativen Lernens entwickelt, die seit mittlerweile zwei Jahren erfolgreich im Einsatz ist.

Durch Feedback der Studierenden, welche CURE im Rahmen von Lehrveranstaltungen, wie z.B. Seminaren oder Praktika genutzt haben, wurde deutlich, dass großes Interesse daran besteht, CURE auch nomadisch zu nutzen. Fernstudierende, die sich ja gerade für eine orts- und zeitflexible Variante des Studiums entschieden haben, beschäftigen sich zum überwiegenden Teil zuhause mit dem Lernstoff. Hier kann schon aus Kostengründen nicht davon ausgegangen werden, dass eine ständig verfügbare Internetverbindung besteht. Oft werden auch Reisezeiten (z.B. im Zug oder Flugzeug) zum Lernen genutzt. Die nomadische Verfügbarkeit von Inhalten würde daher das Nutzungsspektrum von CURE in der Lehre deutlich erweitern.

In diesem Artikel beschreiben wir, wie wir CURE für eine nomadische Nutzung erweitert haben und welche Anforderungen dabei zu berücksichtigen waren. Im folgenden Abschnitt stellen wir kurz die kooperative Lernplattform CURE vor, bevor wir durch einige kurze Szenarien die Notwendigkeit, die Vorteile und die Anforderungen einer nomadischen Nutzung beschreiben. Dann stellen wir im Abschnitt 4 unser Lösungskonzept vor und gehen auf die Realisierung dieser Konzepte ein. Im Abschnitt 5 vergleichen wir unsere Lösung mit anderen Ansätzen, bevor wir mit einer Zusammenfassung und einem Ausblick auf zukünftige Arbeiten schließen.

2 Ein kurzer Einblick in CURE

CURE [HSH⁺03] basiert auf der Metapher der virtuellen Räume [GR03] [PSBWW98], die häufig zur Strukturierung von Kooperation eingesetzt wird. Hierbei dient ein Raum der Repräsentation eines virtuellen Orts für die Zusammenarbeit. In CURE können Räume Inhaltsseiten, Kommunikationskanäle wie z.B. Chat oder Diskussionsforen und Nutzer enthalten. In einem Raum können Nutzer auf alle Seiten zugreifen. Veränderungen dieser Seiten sind für alle Nutzer im Raum sichtbar. Zur Definition von Zugriffsrechten auf Räume wird das Konzept des virtuellen Schlüssels [HHS⁺04] benutzt. Jeder Schlüssel definiert die Rechte des Inhabers für den zugehörigen Raum (z.B. Rechte zum Betreten des Raums, zum Erzeugen eines Unterraums, zum Editieren von Seiten oder zur Kommunikation im Raum).

Nutzer mit entsprechenden Rechten können einen Raum betreten, auf seine Kommunikationskanäle zugreifen und an der Zusammenarbeit im Raum teilnehmen. Ebenso können sie Seiten erzeugen und editieren. Die Seiten können entweder direkt per einfacher Wiki-Syntax [LC01] editiert werden oder sie können binäre Dokumente oder Artefakte enthalten. Die Wiki-Syntax unterstützt Links zu anderen Seiten, Räumen, externen URLs und E-Mail-Adressen. Alle Artefakte werden durch den CURE-Server für den gemeinsamen Zugriff gespeichert. So bleiben die Inhalte eines Raumes erhalten, auch wenn keine Nutzer sich aktuell in diesem aufhalten. Nutzer können später zurückkommen und weiterarbeiten.

Abb. 1 zeigt einen typischen Raum in CURE. Ein Raum enthält Seiten (A), die von Nutzern mit ausreichenden Editierrechten manipuliert werden können (B). Wenn Seiten verändert werden, werden alle Versionen einer Seite gespeichert (C). Der Raum stellt zwei Kommunikationskanäle, d.h. eine Mailbox (D) und einen Chat (E), zur Verfügung. Die raumbasierte Mailbox kann zum Senden von E-Mails an den Raum verwendet werden, die dann von Nutzern mit Kommunikationsrechten für diesen Raum empfangen werden.

Durch miteinander verbundene Räume können strukturierte Lernumgebungen erstellt werden. Mittels eines Plenarraums für die gesamte Gruppe und entsprechenden Räumen für Untergruppen kann z.B. die Kommunikation und die gemeinsame Arbeit in einem Kurs oder einer Organisation unterstützt werden. Ausgehend von dem Plenarraum können Nutzer zu den verbundenen Unterräumen navigieren (F).

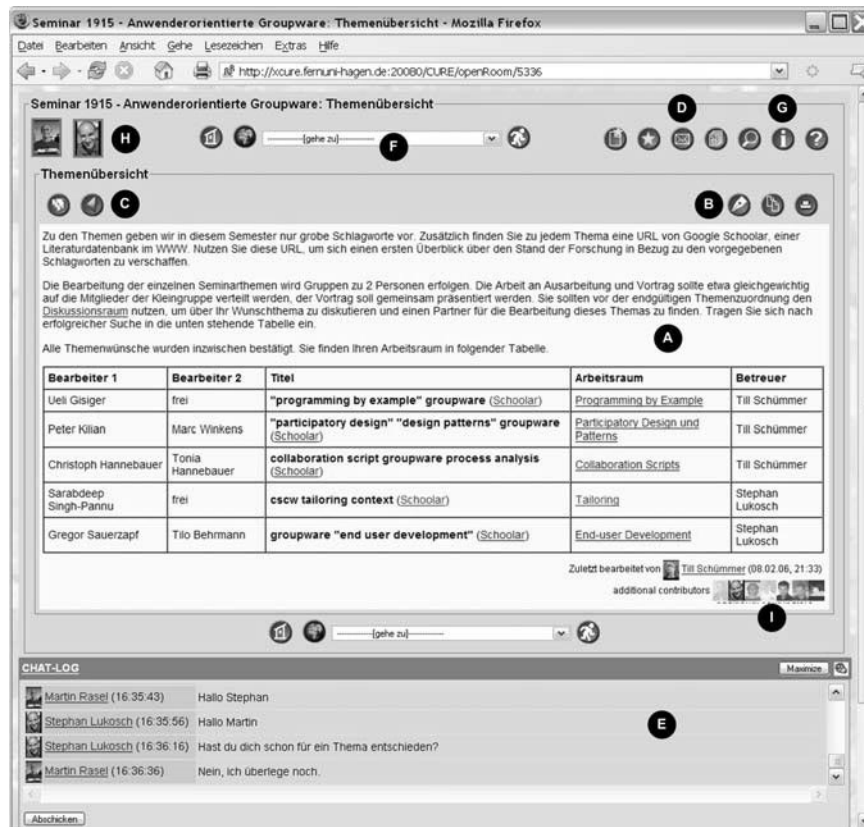


Abbildung 1: Ein Raum in CURE

Zur Koordination zwischen den Nutzern eines Raums unterstützt CURE verschiedene Typen von Awareness-Information. Die Raumeigenschaften zeigen, wer Zugriff auf den Raum hat (G). Es wird angezeigt, wer gerade im Raum anwesend ist (H). Über den Chat (E) können diese Nutzer direkt miteinander kommunizieren. Es wird angezeigt, wer die aktuelle Seite bisher bearbeitet hat (I) und schließlich informiert ein automatisch verschickter Mail-Report alle Nutzer, welche Änderungen im Raum seit dem letzten Report geschehen sind.

3 Problemanalyse

In diesem Abschnitt werden wir anhand von zwei Szenarien aus der Praxis an der FernUniversität in Hagen die Anforderungen für eine nomadische Nutzung von CURE ableiten. Bei den Szenarien handelt es sich um die Durchführung eines Seminars in CURE (siehe Abschnitt 3.1) und eine studentische Arbeitsgruppe in CURE

(siehe Abschnitt 3.2). In diesem Zusammenhang werden wir auch die Vorteile einer nomadischen Nutzung verdeutlichen.

3.1 Seminar in CURE

Der Seminarteilnehmer Klaus hat gerade die Teilnahmebestätigung für ein Online-Seminar erhalten. Für dieses Seminar wurde vom Kursbetreuer in CURE ein Raum eingerichtet, auf den alle Seminarteilnehmer Zugriff haben (wie z.B. in Abb. 1). In diesem Raum wurden einige Textseiten erstellt, die das Thema des Seminars grob umreißen und Hinweise zum Ablauf geben. Weiterhin wurden Unterräume für einzelne Arbeitsgruppen erstellt, die sich jeweils mit der Bearbeitung eines spezifischen Themas beschäftigen sollen. Dort wurden einige Binärseiten abgelegt, die Veröffentlichungen zum Thema der jeweiligen Arbeitsgruppe im PDF-Format enthalten. Die Teilnehmer sollen ihre Arbeitsbereiche zur Diskussion untereinander nutzen, Zwischenergebnisse festhalten, und die fertigen Seminarbeiträge einstellen.

Klaus möchte sich gerne auf das Thema vorbereiten, jedoch muss er in Kürze zu einer beruflichen Reise aufbrechen. Obwohl er weiß, dass er während der Reise einige Stunden Zeit haben wird, bleibt ihm nur die Möglichkeit, die PDF-Dateien herunterzuladen, um wenigstens diese studieren zu können. Unter Verwendung eines Offline-Clients für CURE mit vollwertiger Darstellung und Editierbarkeit der Daten könnte Klaus die Inhalte des Seminarraums vollständig vom CURE-Server herunterladen und während der Reise auf seinem Laptop lesen. Es wäre ihm auch möglich, sofort erste Gedanken in Form neuer CURE-Seiten im Raum seiner Arbeitsgruppe festzuhalten, und diese bei seiner Ankunft auf den Server zu laden.

3.2 Studentische Arbeitsgruppe in CURE

Claudia möchte CURE nutzen, um zum Austausch mit anderen Studierenden eine strukturierte Übersicht des Bachelorstudiums Informatik an der FernUniversität in Hagen anzulegen. Sie plant, einen Raum zu erzeugen, der für jeden von ihr absolvierten Kurs einen Unterraum enthält. Diese sollen wiederum Dokumente zur Klausurvorbereitung oder Prüfungsprotokolle als Text- oder Binärseiten enthalten. Als Inhaltsverzeichnis soll pro Raum eine Textseite mit Links auf die anderen Seiten eingebunden werden. Im Hauptraum soll ein Verzeichnis aller Unterräume verfügbar sein. Sie stellt sich vor, dass andere Studierende dieses Konstrukt nutzen, um ihrerseits Material einzustellen, und unter Verwendung der Kommunikationswege der einzelnen Kursräume (Mail, Chat usw.) die Inhalte zu diskutieren.

Die Erstellung dieser komplexen Struktur ist mit der Weboberfläche von CURE durchaus möglich. Jedoch weiß Claudia, dass hierzu vielfältige Interaktionen mit dem zentralen CURE-Server nötig sein werden, und sie eine lange Online-Zeit einplanen muss. Da sie eine sehr verschachtelte Struktur mit vielen Unterräumen

erzeugen möchte, muss sie mehrfach den Raumkontext wechseln, was leicht unübersichtlich werden kann. Mittels eines Offline-Clients würde sich dieses Projekt wesentlich einfacher durchführen lassen, da zu jedem Zeitpunkt der gesamte Inhalt des Arbeitsbereichs sichtbar ist und ohne Kosten für die Internetverbindung in Ruhe alle Inhalte erstellt werden könnten.

3.3 Anforderungsanalyse

Die beschriebenen Szenarien zeigen, dass eine nomadische Nutzungsmöglichkeit von CURE als Ergänzung zu den bestehenden Diensten unter verschiedenen Ansatzpunkten nützlich ist. Durch die Möglichkeit, CURE-Inhalte ohne bestehende Internetverbindung betrachten und bearbeiten zu können, lassen sich Zeiten der Mobilität effektiv nutzen. Lernen und Arbeiten mit CURE würde so prinzipiell überall möglich.

CURE als webbasierte CSCL-Plattform ist so ausgelegt, dass es von jedem beliebigen Internetzugang unter Verwendung gebräuchlicher Webbrowser nutzbar ist. Diese nicht zu unterschätzende Flexibilität wird jedoch durch Einschnitte in der Benutzungsschnittstelle erkauft. Jede Interaktion muss erst zum Server übertragen und dort verarbeitet werden, bevor die Reaktion beim Nutzer angezeigt werden kann. Eine lokal laufende Applikation zur Nutzung und Bearbeitung von CURE-Inhalten kann durch eine Verringerung der Antwortzeit den Nutzungskomfort steigern. Weiterhin kann ein solcher lokaler Client vielfältige Möglichkeiten ausschöpfen, um die Nutzerinteraktion komfortabler und intuitiver zu gestalten, die einer webbasierten Plattform naturgemäß nicht zur Verfügung stehen.

Folgende Anforderungen an den lokalen Client und die Serveranbindung wurden von uns vor dem Hintergrund der geschilderten Szenarien identifiziert:

- A1:** Es muss eine Kommunikationsschnittstelle zwischen dem CURE-Portal und dem lokalen Client geschaffen werden, die es erlaubt, auf die Daten in CURE sowohl lesend als auch schreibend zuzugreifen.
- A2:** In CURE wird der Zugriff auf Räume und ihren Inhalt durch virtuelle Schlüssel reguliert. Die somit geltenden Zugriffsrechte müssen auch bei der Nutzung des lokalen Clients berücksichtigt werden und es muss eine Authentifizierung des Nutzers durchgeführt werden.
- A3:** Nutzer müssen die Möglichkeit haben, diejenigen Inhalte auszuwählen, die lokal verfügbar sein sollen. Für eine zeitlich flexible Nutzung müssen die lokalen Daten persistent sein. Neue und geänderte Inhalte oder auch Veränderungen der Konfiguration müssen mit CURE synchronisiert werden können. Dies beinhaltet auch die Auflösung von Konflikten durch parallele Bearbeitung von Inhalten.
- A4:** Eine intuitiv nutzbare Benutzungsschnittstelle soll bei der Erstellung und Be-

arbeitung von CURE-Inhalten die Bearbeitung im Vergleich zur Webschnittstelle vereinfachen.

4 Lösungskonzept

Zur Lösung der gestellten Anforderungen wurde auf Seite des CURE-Servers eine Kommunikationsschnittstelle (*offlineCURE-Server*) erstellt, die parallel zum CURE-Weboberfläche Zugriff auf die Inhalte von CURE bietet. Der lokale Client (*offlineCURE-Client*) wurde als Java-Applikation realisiert, um Unabhängigkeit von den verschiedenen Betriebssystemen der Studierenden zu erreichen. Die folgenden vier Abschnitte illustrieren, wie die einzelnen Anforderungen (A1-A4) erfüllt werden.

4.1 Kommunikationsschnittstelle (A1)

Die Kommunikation zwischen einem offlineCURE-Client und dem offlineCURE-Server wird durch eine sichere SSL-Verbindung gewährleistet. Nachdem eine Verbindung aufgebaut wurde, muss sich der Client zunächst über die Zugangsdaten des CURE-Nutzers authentifizieren, bevor Daten ausgelesen oder verändert werden können. Erfolgreich angemeldeten Clients wird jeweils ein eigener Prozess auf dem Server zugeordnet, der die übermittelten Befehle interpretiert und ausführt.

Zur Kommunikation mit dem offlineCURE-Server wird eine Skriptsprache verwendet. Diese Skriptsprache erlaubt es, alle lesenden bzw. schreibenden Funktionen von CURE anzusprechen, um auf CURE-Inhalte zugreifen zu können. Unter anderem ist es möglich, Raum- oder Seiteninhalte auszulesen, Seiten zu verändern und zu synchronisieren, neue Seiten oder Räume zu erzeugen oder Raumeigenschaften zu verändern. Um Nebenläufigkeiten bei der Synchronisierung entgegenzuwirken, benutzt die offlineCURE-Schnittstelle die internen Objekt-Verwaltungsprozesse von CURE. Diese arbeiten mittels Transaktionen auf einer Datenbank, so dass Inkonsistenzen vermieden werden.

4.2 Zugriffsbeschränkung gemäß der Nutzerrechte in CURE (A2)

Jeder Nutzer in CURE hat unterschiedliche Rechte. Diese werden in Form virtueller Schlüssel repräsentiert, über die für jeden Raum die erlaubten Aktionen des Nutzers festgelegt sind. Die Einhaltung dieser Zugriffsrechte wird bei der Verwendung von offlineCURE durch die Kommunikationsschnittstelle geregelt. Jeder Befehl, der vom Client über die Schnittstelle abgesetzt wird, ist eindeutig einem bestimmten Zielraum zugeordnet. Hierdurch ist es dem offlineCURE-Server möglich, vor der Ausführung zu überprüfen, ob der Nutzer über ausreichende Rechte verfügt. Falls

ihm bestimmte Rechte fehlen, wird die Operation abgebrochen und eine Fehlermeldung mit Fehlercode an den Client übermittelt. Auf diese Weise ist sichergestellt, dass über den Client ausschließlich jene Aktionen ausgeführt werden, die ein Nutzer auch über die Weboberfläche von CURE vornehmen kann.

4.3 Auswahl und Synchronisierung der lokalen Daten (A3)

Zur Auswahl der lokal zu replizierenden Inhalte im Client wird die rekursive Struktur der CURE-Daten genutzt. Alle CURE-Räume und die enthaltenen Seiten sind über den zentralen Zugangsraum erreichbar und jeweils eindeutig einem Raum zugeordnet. Sie lassen sich daher übersichtlich in einer Baumstruktur darstellen, aus welcher der Nutzer die gewünschten Inhalte selektiv auswählen kann. Um den Datentransfer beim Aufbau der Inhaltsübersicht möglichst gering zu halten, werden vom offlineCURE-Server jeweils nur die notwendigen Informationen für die aktuell angezeigten Rekursionstiefe bezogen und weitere Daten bei Bedarf nachgeladen.

Beim Download der ausgewählten Inhalte werden diese vom Server bezogen und lokal gespeichert, wobei die Strukturierung nach Räumen und Seiten identisch zu CURE vorgenommen wird. Der Client kann nach der Datenreplikation auf unbestimmte Zeit vom Netz getrennt werden, und Inhalte können lokal betrachtet oder verändert werden. Die Originaldaten auf dem Server werden nicht gesperrt, so dass es anderen Nutzern möglich ist, diese per offlineCURE oder direkt in CURE ebenfalls zu bearbeiten (*optimistische Replikation* [SS05]). Da die Änderungen nicht automatisch zu den anderen Kopien propagiert werden, müssen eventuell auftretende Konflikte bei der Synchronisierung mit dem Server berücksichtigt werden, die zum gewünschten Zeitpunkt vom Nutzer angestoßen wird.

Neue und geänderte Inhalte werden bei der Synchronisierung durch eine Kombination von Zeitstempeln auf dem Server und *Modified Bits* [SS05] auf dem Client erkannt, die bei Bearbeitung gesetzt werden. Die Richtung des notwendigen Datentransfers und eventuell zu berücksichtigende Konflikte werden dem Nutzer in einer Baumstruktur analog zur initialen Datenauswahl angezeigt, so dass die zu synchronisierenden Inhalte gezielt ausgewählt werden können.

Zur Konflikterkennung bedient sich der offlineCURE-Server der CURE-intern verwendeten Versionierungsfunktionen. Auf diese Weise fügen sich neue und geänderte Seiten, die vom offlineCURE-Client übertragen werden, nahtlos in das bestehende Datenmodell des CSCL-Portals ein. Treten durch parallele Bearbeitung von Inhalten Konflikte bei der Synchronisierung auf, so werden die kollidierenden Seitenversionen blockweise nach identischen und unterschiedlichen Textbereichen aufbereitet, und dem Nutzer zur manuellen Zusammenführung im offlineCURE-Client übersichtlich gegenübergestellt (siehe Abbildung 2). Eine Schnittstelle im Programm ermöglicht hier die zukünftige Integration von weiteren Verfahren zur Konfliktlösung.

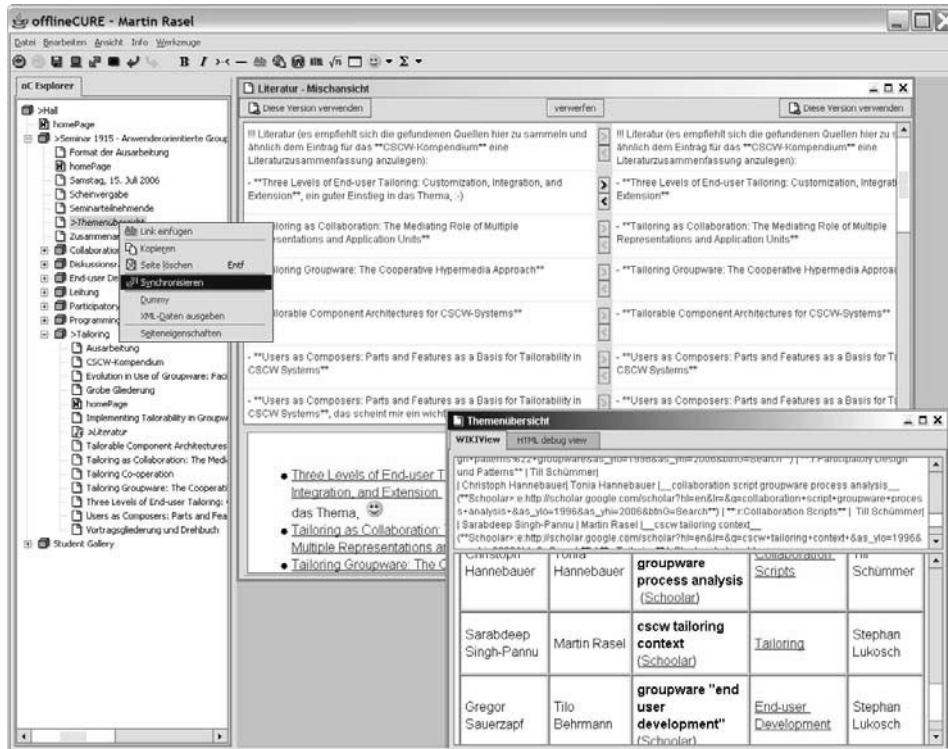


Abbildung 2: offlineCURE Benutzungsschnittstelle

4.4 Benutzungsschnittstelle (A4)

Eine Anforderung bei der Erstellung von offlineCURE war es, dem Nutzer eine leicht verwendbare Benutzungsschnittstelle sowie eine übersichtliche Darstellung der CURE-Inhalte zur Verfügung zu stellen. Letzteres wird durch die Anzeige der lokal vorhandenen Inhalte in Form einer Baumansicht erreicht (links in Abbildung 2). Der Detaillierungsgrad kann durch Ein- und Ausblenden von Baumzweigen jederzeit frei bestimmt werden, um die Verwaltung komplexer Strukturen aus Räumen, Unterräumen und Inhaltsseiten zu erleichtern. Diese Baumansicht zeigt zu jedem lokal vorhandenen Inhalt auch dessen Konsistenzstatus im Vergleich zum CURE-Server an. Ebenfalls an dieser Übersicht greifen kontextsensitive Menüs an, die je nach Typ des ausgewählten Objekts (z.B. Raum, Textseite, Binärdaten) geeignete Aktionen zur Bearbeitung anbieten. Weitere Elemente zur Nutzerinteraktion sind statische Menüs, eine Toolbar mit häufig verwendeten Aktionen, sowie Tastatur-Shortcuts. Dem Nutzer steht somit eine reichhaltige Schnittstelle zur komfortablen Erstellung und Bearbeitung von Inhalten zur Verfügung.

Abbildung 2 zeigt weiterhin mehrere geöffnete Fenster, in denen CURE-Inhalte offline bearbeitet werden (eine Misch-Ansicht zur Auflösung eines Versionskonflikts

sowie eine Textseite mit einer komplexen Tabelle). Diese Fenster sind jeweils unterteilt in einen editierbaren Bereich und einen Previewbereich. Während der Bearbeitung von Seiteninhalten in der Wiki-Syntax von CURE wird der gerenderte Inhalt in Echtzeit angezeigt. Die Umsetzung der Wiki-Formatierung geschieht hierbei analog zum Rendering im Web-Interface von CURE und schließt auch funktionsfähige Hyperlinks auf lokal verfügbare Inhalte sowie Internetadressen ein. Unterstützung bei der Verwendung der Wiki-Syntax wird durch Toolbar-Schaltflächen zum Einfügen häufig verwendeter Wiki-Tags, Smilies und \LaTeX -Formeln gegeben. Wenn gewünscht, lassen sich sowohl der editierbare Fensterbereich als auch der Previewbereich selektiv ein und ausblenden, um offlineCURE z.B. ausschließlich zur Offline-Betrachtung von CURE-Inhalten zu verwenden (analog zum Webbrowser), oder das gesamte Fenster zur Editierung zur Verfügung zu haben.

5 Verwandte Arbeiten

In diesem Abschnitt werden vorhandene Systeme vorgestellt, die nomadischen und mobilen Zugriff auf Inhalte oder gemeinsame Ressourcen ermöglichen. Lernplattformen, wie z.B. *Moodle* [Moo06] oder *WebCT* [Web06], werden hier nicht betrachtet, da sie keine Unterstützung für einen nomadischen Zugriff bieten.

BSCW [AM99] und *BSCl* [Sta02] sind webbasierte Kooperations- bzw. Lernportale. BSCW erlaubt Dokumente lokal zu speichern, sie zu verändern und sie wieder auf dem Server abzuspeichern. Mittels *BSCWeasel* [BSC06] kann auf einen BSCW-Server zugegriffen werden, ohne die Webschnittstelle zu nutzen. Zur Zeit ist es jedoch nicht möglich, mittels BSCWeasel offline zu arbeiten. Des Weiteren bietet BSCW im Allgemeinen keine automatische Versionsverwaltung und keine Möglichkeit unterschiedliche Versionen bei einem Konflikt zu mischen.

Letzteres trifft auch auf *KOLUMBUS* [HK03] zu, das zwar den Im- bzw. Export von Daten ermöglicht, aber ebenfalls keine Verfahren zur Konfliktlösung bereitstellt.

Mittels *Blackboard Backpack* [Bla06] können Studierende die Inhalte der *Blackboard* Lernplattform nomadisch nutzen. Die Nutzung beschränkt sich allerdings auf einen lesenden Zugriff der geladenen Inhalte.

FirstClass [Fir06] ist eine zu CURE ähnliche kommerzielle Groupware. Im Vergleich zu CURE und offlineCURE bietet FirstClass keine Versionskontrolle für Dokumente. Stattdessen werden Dokumente während der Bearbeitung durch einen Nutzer gesperrt, und können zwischenzeitlich von anderen nur lesend geöffnet werden. Somit werden unsere Anforderungen an die Synchronisation und Konfliktbehandlung von FirstClass nicht erfüllt.

eBag [BCG⁺05] stellt eine *digitale Schultasche* dar, in der sich z.B. Bilder, Videos, Musik und Textdokumente für das Lernen in der Schule befinden können. eBag unterstützt nomadisches Verhalten dadurch, dass Schüler sich zwischen stationären Terminals, sogenannten *digitalen Oasen*, bewegen können. Außerhalb dieser Oasen, von den Entwicklern als offline bezeichnet, können Nutzer auf die persönliche

Schultasche über einen Webbrowser zugreifen. Allerdings unterstützt auch eBag keine Versionierung der gespeicherten Daten.

Bleckmann et al. [BSH05] beschreiben, wie eine Gruppe von Studierenden synchron und kooperativ gemeinsam lernen können, ohne mit der Lernplattform verbunden zu sein. Dazu müssen die Studierenden untereinander ein Ad-hoc-Netzwerk aufbauen. So wird zwar eine Entkoppelung von der Lernplattform erreicht, aber die Studierenden müssen sich dennoch an einem Ort treffen und können nicht individuelle Zeiten der Mobilität nutzen.

Mit *Personal Wikis*, wie z.B. *WikiWriter* [Wik06b], *EclipseWiki* [Ecl06] und *WikidPad* [Wik06a], können die Inhalte eines lokalen Wikis verwaltet, erstellt und editiert werden. Sie setzen auf einer lokalen Datenbank oder lokal gespeicherten Dateien auf. Sie besitzen zwar Funktionen, die für nomadisches Bearbeiten nötig sind, können aber zur Kollaboration nicht eingesetzt werden, da sie nur auf lokalen Daten arbeiten und keine Interaktion zwischen mehreren Anwendern unterstützen.

SimpleWikiEditMode [Sim06] unterscheidet sich von den obigen Personal Wikis und benutzt *Emacs*, um Inhalte eines auch über das Web erreichbaren Wikis lokal zu bearbeiten. Allerdings bietet SimpleWikiEditMode keine Unterstützung zur Auswahl der Inhalte, zum Auflösen von Konflikten oder zur Ermittlung lokal geänderter Seiten.

Der *Wikipedia Editor* [PLo06] ist ein *Eclipse Plugin*, das die Nutzern die Möglichkeit bietet, Wikipedia-Artikel lokal zu bearbeiten. Die Artikel können lokal gespeichert und verändert werden. Allerdings bietet das Plugin keine Unterstützung zur Synchronisierung, d.h. zur Konflikterkennung und -auflösung. Stattdessen werden lokal vorhandene Texte einfach durch die aktuellen Texte überschrieben.

Keines der hier betrachteten Systeme erfüllt die von uns identifizierten Anforderungen für nomadisches kooperatives Lernen. offlineCURE stellt im Vergleich zu diesen Systemen eine deutliche Weiterentwicklung dar, da es Studierenden erlaubt, Zeiten der Mobilität zum Lernen zu nutzen und die in dieser Zeit erzielten Ergebnisse in den kooperativen Lernprozess durch Synchronisierung der lokalen Inhalte wieder einzubringen.

6 Zusammenfassung und Ausblick

Befragungen von Studierenden in Anschluss an Lehrveranstaltungen, die unter der Verwendung von CURE durchgeführt worden sind, haben gezeigt, dass eine nomadische Nutzung von CURE von großem Interesse ist. Eine solche Möglichkeit würde es den Studierenden erlauben, auch in Zeiten der Mobilität zu lernen.

Wir haben in diesem Artikel beschrieben, wie wir die webbasierte kollaborative Lernplattform CURE um offlineCURE erweitert haben, um eine nomadische Nutzung zu ermöglichen. offlineCURE nutzt eine Kommunikationsschnittstelle im CURE-Server, um auf die Inhalte sowohl lesend als auch schreibend zu zugreifen.

Dabei berücksichtigt offlineCURE die geltenden Zugriffsrechte, die in CURE durch virtuelle Schlüssel festgelegt werden. Nutzer von offlineCURE haben die Möglichkeit, von allen für Sie erreichbaren Inhalten diejenigen auszuwählen, die lokal verfügbar gemacht werden sollen. Die so verfügbaren lokalen Daten können betrachtet, editiert und mit dem Server synchronisiert werden. Versionskonflikte, die durch die nomadische Nutzung vermehrt auftreten können, können im Vergleich zur Web-Schnittstelle übersichtlich gelöst werden. Des Weiteren vereinfacht offlineCURE den Aufbau und die Bearbeitung komplexer Strukturen.

Erste Erfahrungen beim Einsatz von offlineCURE haben gezeigt, dass alle von uns ermittelten Anforderungen erfüllt werden. Nutzer von offlineCURE haben berichtet, dass die ständige Verfügbarkeit der Inhalte von CURE es ihnen erleichtert, ihr Lernen zu organisieren. Ebenso wurde berichtet, dass es Ihnen wesentlich leichter gefallen ist, Inhalte zu editieren oder neue zu erzeugen. Dies zeigt, dass eine Nutzung von offlineCURE auch bei einer permanenten Netzwerkanbindung sinnvoll erscheint, da offlineCURE komfortableres Arbeiten ermöglicht.

In Zukunft wollen wir offlineCURE in eingeschränkter Funktionalität für kleinere mobile Endgeräte, wie z.B. PDAs, zur Verfügung stellen. Ebenso wollen wir offlineCURE in mehreren Lehrveranstaltungen einsetzen. Dabei wollen wir evaluieren, ob durch die nomadische Nutzungsmöglichkeit die Nutzung von CURE gesteigert wird und der Lernerfolg der Studierenden, die offlineCURE nutzen, sich im Vergleich zu anderen Studierenden verbessert.

Literatur

- [AM99] W. Appelt und P. Mambrey. Experiences with the BSCW Shared Workspace System as the Backbone of a Virtual Learning Environment for Students. In *Proceedings of ED-MEDIA99*, 1999.
- [BCG⁺05] Christina Brodersen, Bent G. Christensen, Kaj Grønbaek, Christian Dindler und Balasuthas Sundararajah. eBag: a ubiquitous Web infrastructure for nomadic learning. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, Seiten 298–306, Chiba, Japan, 2005. ACM Press.
- [Bla06] Blackboard Inc. Blackboard Backpack. <http://backpack.blackboard.com/>, März 2006.
- [BSC06] BSCWeasel. Erweiterbarer Rich Client für das BSCW Groupware System. <http://www.bscweasel.de/>, März 2006.
- [BSH05] Peter Bleckmann, René Sprotte und Thorsten Hampel. Interaktive Lernobjekte in einem mobilen Einsatzumfeld. In Jörg M. Haake, Ulrike Lucke und Djamshid Tavangarian, Hrsg., *DeLFI 2005: Die 3. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI)*, Seiten 339–350, Rostock, Germany, September 2005. GI.
- [Ecl06] EclipseWiki. Eclipse Wiki Editor Plugin. <http://eclipsewiki.sourceforge.net/>, März 2006.

- [Fir06] FirstClass. Product Homepage. <http://www.firstclass.com>, März 2006.
- [GR03] Saul Greenberg und Mark Roseman. Using a Room Metaphor to Ease Transitions in Groupware. In M. Ackermann, V. Pipek und V. Wulf, Hrsg., *Sharing Expertise: Beyond Knowledge Management*, Seiten 203–256. MIT Press, Cambridge, MA, USA, Januar 2003.
- [HHS⁺04] J. M. Haake, A. Haake, T. Schümmer, M. Bourimi und B. Landgraf. End-User Controlled Group Formation and Access Rights Management in a Shared Workspace System. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, Seiten 554–563, Chicago, Illinois, USA, November 6-10 2004. ACM Press.
- [HK03] Thomas Herrmann und Andrea Kienle. KOLUMBUS: Context-oriented communication support in a collaborative learning environment. In T. J. van Weert und K. S. Munro, Hrsg., *Informatics and the Digital Society – Social, Ethical, and Cognitive Issues*, Seiten 251–260. Kluwer, 2003.
- [HSH⁺03] J. M. Haake, T. Schümmer, A. Haake, M. Bourimi und B. Landgraf. Two-level tailoring support for CSCL. In *Groupware: Design, Implementation, and Use. Proceedings of the 9th International Workshop (CRIWG 2003)*, LNCS 2806, Seiten 74–82, Heidelberg, September 2003. Springer.
- [LC01] Bo Leuf und Ward Cunningham. *The WIKI way*. Addison-Wesley, Boston, MA, USA, 2001.
- [Moo06] Moodle. A Free, Open Source Course Management System for Online Learning. <http://moodle.org/>, März 2006.
- [PLo06] PLog4U. Eclipse Wikipedia Plugin. http://www.plog4u.de/index.php/Eclipse_Wikipedia_Plugin, März 2006.
- [PSBWW98] Hans-Rüdiger Pfister, Christian Schuckmann, Jennifer Beck-Wilson und Martin Wessner. The Metaphor of Virtual Rooms in the Cooperative Learning Environment CLear. In *Proceedings of CoBuild'98*, LNCS 1370, Seiten 107–113, Heidelberg, 1998. Springer.
- [Sim06] SimpleWikiEditMode. Product Homepage. <http://www.emacswiki.org/cgi-bin/wiki/SimpleWikiEditMode>, März 2006.
- [SS05] Yasushi Saito und Marc Shapiro. Optimistic replication. *ACM Computing Surveys*, 37(1):42–81, 2005.
- [Sta02] Gerry Stahl. Groupware Goes to School. In Jörg M. Haake und José A. Pino, Hrsg., *Groupware: Design, Implementation, and Use, 8th International Workshop, CRIWG 2002*, LNCS 2440, Seiten 7–24, La Serena, Chile, September 2002. Springer-Verlag Berlin Heidelberg.
- [Web06] WebCT Inc. Product homepage. <http://www.webct.com>, März 2006.
- [Wik06a] WikidPad. Product Homepage. <http://www.jhorman.org/wikidPad/>, März 2006.
- [Wik06b] WikiWriter. A stand-alone, shareware PersonalWiki for Windows. <http://hytext.com/index.htm>, März 2006.