

# Dienste-basierte Kopplung von virtueller und Präsenzlehre

Philipp Lehsten, Andreas Thiele, René Zilz,  
Enrico Dressler, Raphael Zender, Ulrike Lucke, Djamshid Tavangarian  
Universität Rostock, Institut für Informatik, Lehrstuhl für Rechnerarchitektur  
vorname.nachname@uni-rostock.de

**Abstract:** Der Beitrag zeigt, wie eine Service-Orientierte Architektur die Kluft zwischen verschiedenen Formen des rechnergestützten Lehrens und Lernens überbrücken helfen kann. Der grundlegende Ansatz ist eine bidirektionale Distribution von Diensten verschiedener eLearning-Umgebungen, so dass die Dienste der jeweils anderen Umgebung neben den eigenen verwendet werden können. Dies wird exemplarisch verifiziert durch die Kopplung von Lehrveranstaltungen in einer modernen Medienwerkstatt mit Werkzeugen der rechnergestützten Präsenzlehre (Stud.IP) sowie des virtuellen Lernens (Second Life). An diesem Beispiel wird der durch das Konzept entstehende Mehrwert aufgezeigt, und es werden Möglichkeiten des Einsatzes und der Erweiterung des Systems diskutiert.

## 1 Typische eLearning-Szenarien und Ansätze zu deren Verknüpfung

Die *rechnergestützte Präsenzlehre* ist die derzeit am weitesten verbreitete Form des eLearning [LT07]. Dies beinhaltet den Austausch bzw. die Ergänzung von Vortrag und Tafelanschrieb durch digitale Medien (i. Allg. Folien), die den Lernenden anschließend auf einfache Weise für die Nachbereitung der Veranstaltung zur Verfügung gestellt werden können. Meist erfordert das den manuellen Upload von Foliensätzen oder Skripten in eine web-basierte Lernplattform durch den Lehrenden. Eine vergleichsweise neue Entwicklung ist die (z. T. automatisierte) Archivierung und Publikation von Vorlesungsmitschnitten [Hür08], die neben den Foliensätzen auch ein Video des Dozenten umfasst und zusätzlich die gezielte Navigation in den Aufzeichnungen erlaubt. Die genannten Mechanismen sind jedoch nur asynchron nutzbar, d.h. von der Veranstaltung zeitlich entkoppelt, was Interaktionen mit dem Dozenten auf die Präsenzlehre beschränkt.

Beim *Online* bzw. *Virtual Learning* werden reale Prozesse der Präsenz nachgebildet und ggf. angereichert um spezifische Elemente für die Kommunikation und Kollaboration (wie Chat, Forum oder Shared Whiteboard). Oft kommen hierfür Lehr-/Lern-Management-Systeme (LLMS) zum Einsatz, wie etwa für Tele-Akademien. Auch virtuelle 3D-Welten bergen aufgrund der nachempfundenen sozialen Präsenz von Lehrenden und Lernenden ein Innovationspotential für das eLearning [ML07]. Diese Mechanismen können zwar in sich synchron sein, d. h. es kann eine zeitliche Kopplung von individuellen Lernaktivitäten erfolgen, aber die Umgebung ist dennoch abgekoppelt von realen Präsenzszenerien.

Nicht unerwähnt bleiben soll darüber hinaus das *Offline Learning*. Hier kommt speziell aufbereitete Software für das Selbstlernen zum Einsatz. Die kontinuierliche Interaktion

zwischen Lernendem und Lehrendem wird jedoch nicht unterstützt. Aus didaktischer Sicht ist das nur für ausgewählte Anwendungsfälle sinnvoll.

Ein didaktischer Ansatz zur Kopplung von virtueller und Präsenzlehre ist das *Blended Learning*. Hier erfolgt eine vorherige Unterteilung des Lernprozesses in Phasen mit jeweils eigener Gestaltungsweise. I. Allg. werden Präsenzphasen für die Vor- und Nachbereitung in enger tutorieller Anleitung genutzt, zwischen die intensive eLearning-Phasen für eher informelle Szenarien gelagert werden. Hier erfolgt eine vorgegebene Trennung von synchronen und asynchronen Prozessen – ein Kompromiss zwischen technischen Möglichkeiten und pädagogischen Notwendigkeiten.

Aus technischer Sicht möglich und für den Lernenden durchaus wünschenswert wäre jedoch ein systematischer Ansatz zur nahtlosen Verzahnung von virtueller und Präsenzlehre, bei dem die Nutzer (d. h. Lernende und Lehrende) ad hoc über konkrete Modalitäten entscheiden können. So könnten in einer Lernumgebung etwa synchrone Szenarien zu Präsenzzeiten (für die Interaktion mit dem Dozenten) und asynchrone Szenarien außerhalb der Lehrveranstaltung (für deren individuelle oder kollaborative Vor- und Nachbereitung) transparent vereint werden. Voraussetzungen hierfür sind die flexible Kombination der dafür eingesetzten Werkzeuge/Plattformen bzw. der damit verbundenen Prozesse. Dies sollte auch über die Grenzen einzelner Bildungsanbieter hinweg möglich sein, was neben einem erhöhten Komfort und einer größeren Reichweite auch die Mobilität und Chancengleichheit von Lehrenden wie Lernenden fördern hilft.

## 2 SOA als Architekturkonzept für modernes eLearning

Bei den o. g. eLearning-Szenarien kommen verschiedene Werkzeuge bzw. Infrastrukturen zum Einsatz. Das Spektrum reicht von lokaler Lernsoftware über netzbasierte LLMS bis hin zu Online Communitys oder virtuellen Welten. Prinzipiell gibt es die folgenden architektonischen Ansätze für deren Verzahnung:

- Client-Server-Architektur (siehe Bild 1a):

Bei diesem klassischen Ansatz stellt ein Client seine Anfrage (*request*) an einen vordefinierten Server, der nach Bearbeitung des Auftrags seine Antwort (*response*) zurücksendet. Clients und Server können beliebig im Netzwerk verteilt sein; auch kann ein Rechner für verschiedene Aufgaben wahlweise die Rolle von Client oder Server annehmen. Viele Internet-Dienste (z. B. Mail, Web) sind in der Art organisiert. Als problematisch stellen sich drei Aspekte heraus. Erstens sind bei jedem Client Kenntnisse über die für konkrete Aufträge zu kontaktierenden Server nötig. Zweitens wird die so definierte Menge von Servern schnell zum Flaschenhals, was sich sowohl auf die Leistung als auch auf die Ausfalltoleranz des Systems negativ auswirken kann. Drittens geht die Aktivität immer vom anfragenden Client aus. Daher ist das Client-Server-Prinzip nicht für Anwendungen geeignet, bei denen eine hohe Aktualität von Informationen bzw. Synchronität von Prozessen gefordert ist.

- Publisher-Subscriber-Architektur (siehe Bild 1b):

In diesem Modell registrieren sich die Interessenten für bestimmte Informationen zunächst bei dem dafür zuständigen Anbieter (*subscribe*). Sobald neue Informationen verfügbar sind, benachrichtigt dieser alle dafür registrierten Interessenten in einer

Sammelnachricht (*publish*). Eine typische Anwendung dieses Prinzips stellen Newsletter dar, über die der Empfänger zeitnah mit neuesten Informationen zu den ihn interessierenden Themen versorgt wird.

Da die Aktivität hier stets von dem benachrichtigenden Publisher ausgeht, wird das Aktualitäts- bzw. Synchronitäts-Defizit der Client-Server-Architektur umgangen. Auch kann die im Netzwerk verfügbare Bandbreite durch Multicasts effizient genutzt werden. Die ersten beiden der o. g. Nachteile bleiben jedoch weiter bestehen.

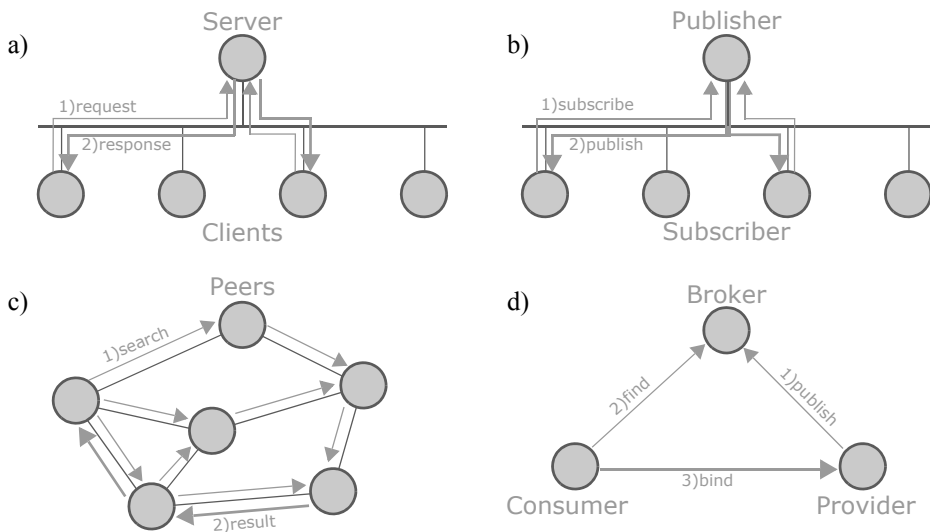
- Peer-to-Peer-Architektur (siehe Bild 1c):

Dieses Konzept stellt eine Loslösung von der Trennung in Auftraggeber und Auftragnehmer dar. Suchanfragen werden durch eine Gemeinschaft von gleichberechtigten, autonomen Peers weitergeleitet (*search*), und gefundene Ergebnisse kommen auf dem gleichen Wege zurück (*result*). Solche Architekturen wurden v. a. durch das File Sharing bekannt, haben aber aufgrund des kooperativen Ansatzes auch im eLearning an Popularität gewonnen – z. B. für die verteilte Verwaltung von Metadaten.

Der zentrale Vorteil dieses Ansatzes ist das Fehlen von Flaschenhälsen und damit die große Skalierbarkeit und Robustheit des Systems. Interessenten können am Netzwerk partizipieren, sobald sie eine beliebige Einstiegsadresse kennen. Allerdings sind gerade bei Anwendungen im eLearning-Bereich häufig explizite Rollenverteilungen vorhanden, die hier ignoriert würden. Auch garantieren die potentiell unzuverlässigen Verbindungen zwischen den Peers keine Vollständigkeit der Suchergebnisse. Zudem kann das Fehlen einer zentralen Kontrolle zu mangelnder Akzeptanz in hierarchischen Organisationsstrukturen (wie etwa der typischen Hochschule) führen.

- Broker-Architektur (siehe Bild 1d):

Die Entkopplung von Auftraggeber und Auftragnehmer kann auch durch einen Broker erfolgen. Er nimmt Dienstbeschreibungen von deren Anbietern entgegen (*publish*) und kann aus dieser Datenbasis bei der Suchanfrage eines Dienstnutzers



**Bild 1** Architekturkonzepte für die Kopplung von Werkzeugen bzw. Infrastrukturen

(*find*) die in Frage kommenden Dienste ermitteln. Erst danach erfolgt zur Laufzeit die Kopplung zwischen Dienstanbieter und -nutzer (*bind*). Das entspricht einer Synthese von Merkmalen der oben genannten Architekturen. Ein viel zitiertes Beispiel dafür sind individuelle Buchungen auf Basis diverser Flug-, Hotel- und Mietwagenanbieter. Die Broker-Architektur spielt ihre Vorteile v. a. in dynamischen Umgebungen aus, da der Broker als einzig feste Instanz bekannt sein muss. Ihr wird ein hohes Potenzial v. a. angesichts ihrer Agilität zugesprochen: einerseits erlaubt das redundante Angebot von verwandten Diensten die sofortige Reaktion auf Ausfälle oder Veränderungen im System, andererseits sind Erweiterungen um neue Anbieter oder Nutzer von Diensten zur Laufzeit und völlig transparent realisierbar. Das schließt auch die Aggregation von bestehenden Diensten zu neuen, höherwertigen Diensten mit ein.

Aus den genannten Gründen findet das Broker-Konzept (unter dem Schlagwort der Service-Orientierten Architektur, SOA) derzeit vielfache Beachtung – nicht nur für ausgewählte eLearning-Systeme [All05][Krö06] und eScience-Plattformen [KW06], sondern generell für universitäre [LT07][Fre07] bzw. betriebliche Infrastrukturen [Bur02]. Aus technischer Sicht erfordert das Broker-Modell (über prozedurale oder Objekt-orientierte Ansätze der zuvor genannten Architekturen hinaus) einen Dienste-basierten Ansatz, der gekennzeichnet ist durch:

- eine lose Kopplung dank minimaler Abhängigkeiten; d. h. Interaktionen in einer heterogenen Umgebung erfolgen unabhängig von konkreten Implementierungen
- eine Inhalts- statt Ressourcen-orientierte Adressierung; d. h. der Request eines Dienstanwenders beinhaltet nur *was*, und nicht *wo* gesucht wird
- eine späte Bindung (*late binding*); d. h. der tatsächlich genutzte Dienst wird erst zur Laufzeit ermittelt und kontaktiert
- eine Spiegelung realer Aktivitäten; d. h. die Architektur korreliert mit unterstützten (Geschäfts-)Prozessen, also Handlungen von Mensch zu Mensch

Die Prinzipien sind hervorragend mit den in Abschnitt 1 erarbeiteten Anforderungen an Mechanismen zur Kopplung verschiedener eLearning-Szenarien vereinbar: Nutzer bestimmen, *wann* und *wo* sie *was* tun möchten, und die Infrastruktur unterstützt die Handlungen transparent durch Auswahl und Kombination geeigneter Dienste. Nachfolgend wird anhand eines Prototyps die praktische Umsetzung dieser Idee geschildert.

## 3 Systemarchitektur und Realisierung

### 3.1 Ausgangslage

Hochgradig rechnergestützte Präsenzlehre findet i. Allg. in speziell ausgestatteten Räumen statt, wie der in Bild 2 gezeigten Medienwerkstatt. Diese bieten neben der herkömmlichen Anzeige von Vortragsfolien zahlreiche weitere Funktionen. Die hier verfügbaren Geräte bieten eine Reihe von Diensten an, wie z. B.:

- Input und Output von Audio- und Video-Daten (z. B. angeboten durch Kameras, Beamer, Mikrofone, Lautsprecher, PC und Laptop)
- Steuerung der Signalverteilung (über Kreuzschienenverteiler und ein drahtloses Panel zur Mediensteuerung)



1. Kameras
2. Leinwände & Beamer
3. Mikrofone
4. Lautsprecher
5. Notebook/PC
6. Mediensteuerung
7. weitere Medientechnik

**Bild 2** Szenario und technische Ausstattung für die rechnergestützte Präsenzlehre in einer typischen Medienwerkstatt

- Aufzeichnung und Wiedergabe von Video- und Audio-Daten
- Herstellung einer Videokonferenzverbindung
- Zugang zum Universitätsnetzwerk und Internet

Die lose Kopplung dieser Dienste ermöglicht eine Vielzahl von Gerätekombinationen (z.B. Kamera-Beamer- oder PC-Lautsprecher-Verbindung). Typische Szenarien aus Lehre und Forschung auf Basis dieser Geräte und ihrer Dienste sind:

- Präsentation von Vortragsfolien und Multimediainhalten
- Aufzeichnung von Veranstaltungen (Video, Audio und Vortragsfolien)
- Durchführung von Videokonferenzen

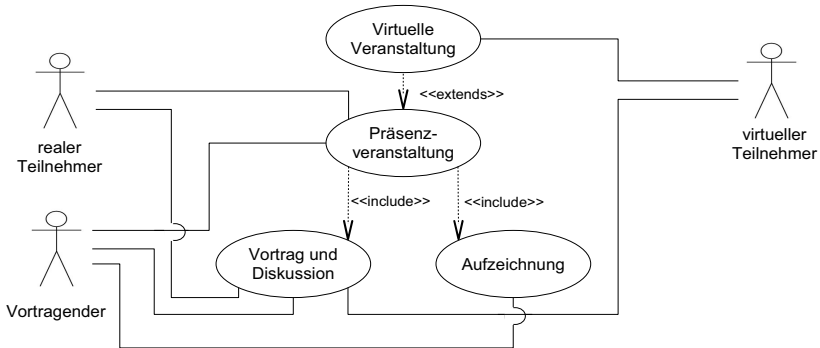
Der Einsatz einer derartigen Medienwerkstatt, aber auch von entsprechend ausgestatteten Hörsälen oder Speziallaboren, ermöglicht eine deutliche Bereicherung der universitären Lehre. Durch die Nutzung von Video-Streams oder Videokonferenzen besteht darüber hinaus die Möglichkeit einer räumlichen Entkopplung der Teilnehmer. Allerdings existiert bei herkömmlichen Video-Streams i. d. R. kein Rückkanal, mit dem externe Teilnehmer bei Live-Veranstaltungen in die Diskussion eingebunden werden könnten. Ein Rückkanal wird durch Videokonferenzsysteme angeboten; jedoch sind diese oft sehr komplexe und proprietäre Systeme, die nicht bei allen Teilnehmern vorhanden bzw. interoperabel sind. Die räumliche Entkopplung ist ein wesentliches Merkmal des Virtual Learning. Es liegt daher nahe, die in diesem Bereich eingesetzten Plattformen mit denen der Präsenzlehre zu koppeln, um einerseits eine räumliche Erweiterung der Präsenzlehre und andererseits einen engeren Realitätsbezug der virtuellen Lehre zu erreichen. Davon könnten neben der Präsenzlehre auch rechnergestützte Exkursionen (sog. Field Trips) profitieren.

Im Folgenden wird ein prototypisches System vorgestellt, das die Dienste-basierte Verbindung beider Lehr-/Lernparadigmen anhand der Kopplung einer Medienwerkstatt mit der virtuellen 3D-Welt *Second Life* [Lin08a] und dem LLMS *Stud.IP* [dat08] realisiert.

### 3.2 Unterstützte Nutzungsmöglichkeiten

Die Anwendungsfälle bei der Kopplung beider Lehr/Lernparadigmen sind in Bild 3 dargestellt. Die *Präsenzveranstaltung* mit dem herkömmlichen Anwendungsfall *Vortrag und Diskussion* und dem inzwischen weit verbreiteten Anwendungsfall der *Aufzeichnung* stellt

den Ausgangspunkt dieser Arbeit dar. Dies kann eine gewöhnliche Vorlesung sein. Sie wird durch eine *virtuelle Veranstaltung* erweitert. Dies kann z. B. eine verbundene Veranstaltung in einer virtuellen 3D-Welt sein, die eine Teilnahme von externen Studenten an der Vorlesung ermöglicht, oder die zur Präsenzveranstaltung gehörende Live-Repräsentation in einem LLMS (z.B. Live-Stream innerhalb der konventionellen Materialsammlung zur Vorlesung).



**Bild 3** Anwendungsfälle der erweiterten Präsenzveranstaltung

Bei einer Live-Veranstaltung wird durch den entwickelten Prototyp die synchrone Beteiligung externer, sog. virtueller Teilnehmer sowohl am Vortrag (passiv) als auch an der Diskussion (aktiv) ermöglicht. Durch mehrere Audio-/Videodienste (z. B. Vortragsfolien, Kamerabild vom Vortragenden oder vom Publikum, eingespielte Medien) kann die Veranstaltung aus unterschiedlichen Blickwinkeln verfolgt werden (z.B. Vortragsfolien während der Vorlesung und Kamerabild des realen Auditoriums zur Diskussion). Bei aufgezeichneten Veranstaltungen ist dagegen nur eine passive, asynchrone Teilnahme möglich. Die damit verbundene Beschränkung der Signalquellen besteht bei einer Live-Veranstaltung nicht.

Der bei der Entwicklung des Prototypen verfolgte, Dienste-basierte Ansatz erlaubt eine flexible Integration von synchronen und asynchronen Szenarien in verschiedene virtuelle Lernumgebungen. Dadurch ist z. B. der spontane Wechsel zwischen verschiedenen Bildquellen (Vortragsfolien, Auditorium, Vortragender, DVD) möglich, ohne die jeweilige Umgebung daran anpassen zu müssen. Die dafür entwickelte Architektur wird im Folgenden erläutert.

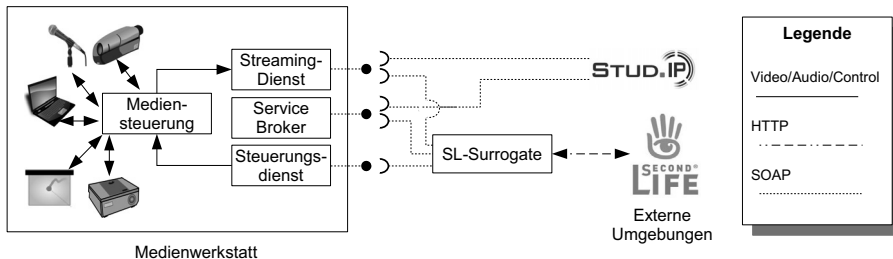
### 3.3 Architektur des Prototypen

Der entwickelte Prototyp setzt sich aus einer Reihe von Einzelkomponenten zusammen. Diese Komponenten und ihre Verbindungen sind in Bild 4 übersichtsartig dargestellt. Sie können drei Schichten zugeordnet werden:

- **SOA-Anbindung der Gerätesteuerung in der Präsenz-Umgebung:** Hier sind die proprietären Mechanismen der Medientechnik abzugreifen und für die Nutzung über ein Netzwerk von Diensten aufzubereiten.

- **Dienstenetzwerk:** Diese Ebene stellt den Kern des Dienste-basierten Ansatzes dar. Neben den Diensten selbst (Ein-/Ausgabe von Audio-/Video-Signalen; Steuerdienste) wird ein Service Broker als zentrales Element einer SOA bereitgestellt.
- **SOA-Anbindung der Lehr-/Lernumgebungen:** Diverse Plattformen können entweder direkt (wenn eine Modifikation der Zugriffsmechanismen möglich ist) oder über einen sog. Surrogate an das Dienstenetzwerk angebunden werden.

Die Architekturskizze zeigt klar, dass dank der Abstraktion durch die Diensteschnittstelle ohne strukturelle Änderungen am System nahezu beliebige Typen, Anbieter oder Nutzer von Diensten integriert werden können. In den folgenden vier Abschnitten werden die Lösungen für die einzelnen Ebenen näher erläutert.



**Bild 4** Grobarchitektur des Systems

### 3.4 Anbindung der Mediensteuerung

Die in der Medienwerkstatt verfügbaren Gerätedienste werden im Regelfall nur für die Präsenzlehre und nicht im Rahmen einer SOA angeboten, genutzt und verwaltet. Hier übernimmt eine zentrale Mediensteuerung die Kontrolle der einzelnen Datenströme. Dies ist ein übliches Verfahren bei derartigen Umgebungen. Die Mediensteuerung legt z. B. fest, dass das vom Notebook angebotene Videosignal zum Videoeingang des Beamers und das Audio-Signal des Mikrofons zum Audio-Eingang der Lautsprecher geleitet wird. In unserer Medienwerkstatt kommt dafür ein spezieller Controller zum Einsatz. Die Steuerung der Kommunikation zwischen den einzelnen Geräten und dem NI-3100 erfolgt netzbasiert über Kommandos des Internet Control System Protocols (ICSP). Um diese Schnittstelle plattformunabhängig in die Architektur zu integrieren, wird das ICMP durch eine Java-basierte Gegenstelle bedient, die eine Portierung aller Bedienfunktionen nach Java erlaubt. Diese Funktionen wurden als Steuerungsdienst in der *Web Service*-Technologie [Mel07] implementiert.

Damit neben dem weichenstellenden Steuerungsdienst auch multimediale Inhalte in externe Umgebungen übertragen werden können, sind weitere Dienste notwendig. In der Architekturdarstellung (Bild 4) ist beispielhaft ein Streaming-Dienst enthalten. Dieser bereitet audiovisuelle Inhalte (z. B. von Kamera und Mikrophon) in Form eines Streams auf und bietet diesen zur Nutzung an. Neben dem Live-Streaming hat der Dienst Zugriff auf archivierte Vorlesungsaufzeichnungen; die technische Umsetzung dieser Unterscheidung ist für die Konsumenten des Dienstes jedoch transparent. Auch der Streaming-Dienst steht in Form eines Web Services für die externe Nutzung zur Verfügung. Die Verteilung des Streams an die Clients in externen Umgebungen wird intern durch einen Helix Server

Unlimited [Rea08] realisiert. Dieser bietet im Vergleich zu anderen Lösungen eine große Menge unterstützter Datenformate (z.B. Real Media, Windows Media oder QuickTime) und gleichzeitiger Streaming Clients. Die Zuteilung der Video- und Audio-Quellen für den Streaming Server erfolgt über den Steuerungsdienst und ist daher auch von außerhalb der Medienwerkstatt durchführbar.

Weitere Dienste aus der Umgebung der Präsenzlehre sind denkbar, aber im Prototypen derzeit noch nicht implementiert (z. B. Schwenk/Zoom der Kameras, Regelung der Mikrofone). Auch können weitere Räumlichkeiten bzw. die von deren Medientechnik angebotenen Dienste völlig transparent integriert werden.

Der folgende Abschnitt befasst sich mit der Realisierung der SOA für das Angebot und die externe Nutzung der Dienste aus der Medienwerkstatt.

### 3.5 Propagation als Netzwerk von Diensten

Die zentrale Komponente einer SOA ist der bereits erwähnte *Service Broker*. Die in der Medienwerkstatt verfügbaren Web Services stehen zwar auch ohne einen Broker zur externen Nutzung zur Verfügung, allerdings erhöht der Einsatz eines Brokers deutlich die Flexibilität bei der Dienstsuche, denn die Consumer müssen kein apriorisches Wissen zur Adressierung jedes verfügbaren Dienstes besitzen. Weiterhin kann über die Eignung eines Dienstes bereits auf Basis seiner Beschreibung im Broker entschieden werden.

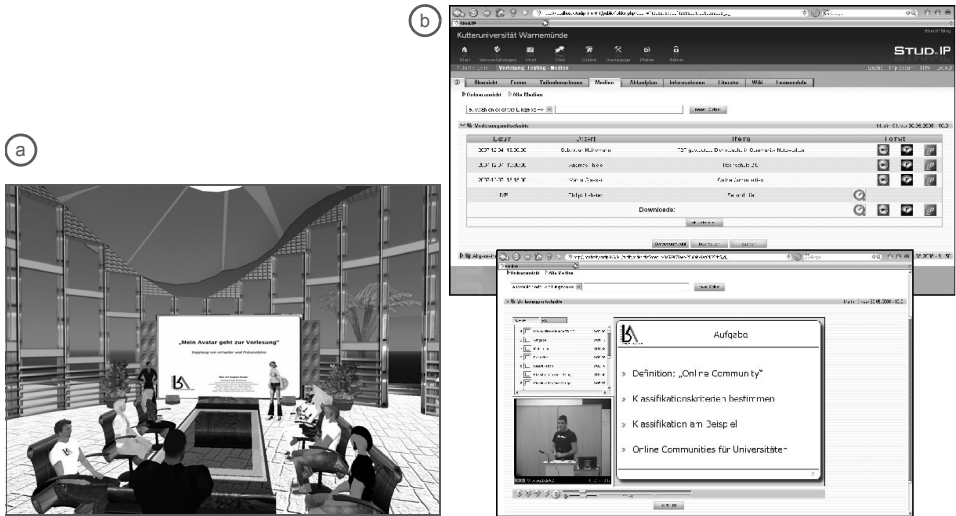
Im vorliegenden Prototypen werden die verfügbaren Dienste bei einem WS-Inspect-basierten Broker [Bal01] registriert. Dieser stellt den zentralen Anlaufpunkt für externe Consumer bei der *Suche* nach Diensten der Medienwerkstatt dar. Die Dienstinformationen werden durch WS-Inspect als einfache XML-basierte Liste gespeichert, die durch HTTP abgerufen werden kann. Dadurch ist eine einfache Integration der Brokeranfrage in verschiedenste Consumer-Systeme möglich.

Die Mechanismen zur *Nutzung* der verfügbare Dienste sind dienstspezifisch. Der Steuerungsdienst stellt z. B. eine Beschreibung auf Basis der Web Service Description Language (WSDL) [Mel07] zur Verfügung, mit der automatisch Schnittstellen für verschiedene Plattformen generiert werden können. Der im Prototyp genutzte Streaming-Dienst wird jedoch über das Real-Time Streaming Protocol (RTSP) [Sch98] genutzt. RTSP ist ein weit verbreitetes Protokoll zur Steuerung von Video-Streams, welches auch im Second Life nutzbar ist. Diese Spezifika der Nutzung müssen bei der Client-seitigen Anbindung an das Dienstenetzwerk, wie sie in den beiden folgenden Abschnitten beschrieben ist, Berücksichtigung finden.

### 3.6 Anbindung von Second Life

Einige virtuelle Lehr-/Lernumgebungen sind bereits in der Lage, Web Services nativ zu nutzen, so dass eine Integration einfach zu realisieren ist. Second Life bietet diese Möglichkeit nicht. Auch eine Anpassung des Systems ist aufgrund der Kapselung beim Anbieter nicht möglich. Für den vorliegenden Prototyp wurde daher ein *SL-Surrogate* entwickelt: ein stellvertretender Consumer, der per HTTP mit Second Life kommuniziert. Die Linden Scripting Language (LSL) ermöglicht in der virtuellen Welt eine Auswertung der Dienstbeschreibungen. Die Auswahl eines Video-Streams (z. B. der Vortragsfolien





**Bild 5** Integration der Präsenzlehre in die virtuelle 3D-Welt *Second Life* und in die Lernplattform *Stud.IP*

wie in Bild 5 a) und die Bedienung des Steuerungsdienstes sind an 3D-Bedienelemente der virtuellen Umgebung gekoppelt und somit aus *Second Life* heraus intuitiv durchführbar.

Der Rückkanal zur Integration des virtuellen Publikums in die Präsenzveranstaltung ist im Prototypen zunächst nur durch den in der Medienwerkstatt vorhandenen *Second Life Client* gegeben. Sinnvoll wäre hierfür eine Umsetzung z. B. des *VoiceChats* von *Second Life* auf die Audio-Kanäle der Medienwerkstatt, ebenfalls auf Basis der *SOA*. Eine solche Dienste-basierte Lösung erfordert jedoch eine intensivere Auseinandersetzung mit der *Client Software* und konnte daher bislang noch nicht realisiert werden. Weitere Details zur Integration der Mediendienste in *Second Life* können einem separaten Beitrag entnommen werden [Lin08b].

### 3.7 Anbindung von Stud.IP

Neben der virtuellen Welt wurde auch eine Anbindung des Systems an eine klassische Lernplattform für die rechnergestützte Präsenzlehre implementiert. Im Gegensatz zu *Second Life* steht *Stud.IP* quelloffen zur Verfügung. Dadurch konnte die Dienstnutzung und -bereitstellung hier direkt integriert werden. Dafür wurde die Ordnerstruktur im Dateibereich um Mechanismen für Video-Streams erweitert und der Bereich in "Medien" umbenannt. Bild 5 b) zeigt die Präsentation von Live- und archivierten Veranstaltungen mit Hilfe einer Auswahlliste und eines integrierten Players in der Browser-Oberfläche von *Stud.IP*. Details zu dieser *AJAX*-basierten Integration werden ebenfalls in einem separaten Beitrag behandelt [Glä08].

## 4 Evaluationsszenario

Zur Evaluation des Prototyps wurde dieser zunächst im Rahmen einer speziellen Lehrveranstaltung mit Second Life-Kopplung getestet.

Dem Publikum in der Medienwerkstatt wurde ein herkömmlicher Vortrag mit entsprechende Vortragsfolien auf einer Leinwand präsentiert. Eine zweite Leinwand zeigte das Publikum im Second Life. Diesem wurden automatisch synchron die Folien präsentiert und es hörte die Stimme des Vortragenden, der durch seinen Avatar in der virtuellen Umgebung präsent war. Für die anschließende Diskussion wurde das Kamerabild des realen Publikums anstelle der Folien in Second Life übertragen. Somit konnten sich reales und virtuelles Publikum gegenseitig wahrnehmen. Fragen aus der virtuellen Umgebung wurden vom realen Publikum gesehen (TextChat) oder gehört (VoiceChat). Letzteres fand jedoch nur einmal statt, da der neu eingeführte VoiceChat von Second Life noch nicht intensiv genutzt wird. Fragen aus der Medienwerkstatt wurden immer durch den Vortragenden in das Second Life vermittelt, da statt mobiler Mikrofone nur das Headset des Vortragenden genutzt wurde.

Der Test hat gezeigt, dass die beschriebene Kopplung nicht nur funktioniert, sondern auch vom virtuellen und realen Publikum akzeptiert wird. Eine intensivere Nutzung des VoiceChats im Second Life würde die Kopplung jedoch noch intensivieren. Im Folgenden wird eine Bewertung des vorgestellten Prototyps vorgenommen, und es werden mögliche Perspektiven für das eLearning diskutiert.

## 5 Bewertung und Erweiterungsmöglichkeiten

Die beschriebene Dienste-basierte Kopplung von Szenarien der Präsenzlehre mit verschiedenen virtuellen Lernumgebungen geht weit über bisherige Ansätze hinaus. So gibt es bereits Vernetzungen zwischen Second Life und traditionellen LLMS [KL06], die sich jedoch auf das Setzen von Querverweisen (Links) bzw. eine gemeinsame Datenbank beschränken. Es handelt sich dabei um keinen systematischen Ansatz zur Verzahnung von synchronen und asynchronen Prozessen beider Lernwelten. Auch die Durchführung von Lehrveranstaltungen im Second Life wird schon intensiv praktiziert [ML07]. Allerdings handelt es sich hierbei um Veranstaltungen, die explizit für diese virtuelle Umgebung konzipiert und durchgeführt werden.

Das hier vorgestellte System ermöglicht durch die konsequente Nutzung einer Service-Orientierten Architektur eine flexible und systematische Kopplung von Plattformen und Werkzeugen für die Präsenz- und die virtuelle Lehre. Dafür kommt eine Abstraktion der verfügbaren Dienste über eine SOA-Zwischenschicht zum Einsatz. Verschiedene Umgebungen können Dienste in die SOA einspeisen oder verfügbare Dienste nutzen. Durch das implementierte System können Lehrende und Lernende erstmals ad hoc über die konkrete Ausgestaltung des Bildungsprozesses (jenseits vordefinierter Phasen wie beim Blended Learning oder vordefinierter Nutzungsumgebungen wie bei entkoppelter Präsenz- oder virtueller Lehre) entscheiden und ihre Entscheidung auch noch während der Veranstaltung anpassen. Weiterhin ermöglicht die entstandene Unabhängigkeit von konkreten Umgebungen eine transparente Vereinigung asynchroner und synchroner

Lernszenarien. Diese ist über die Grenzen einzelner Bildungsanbieter hinweg möglich; dank der transparenten Kapselung durch die SOA muss hierfür nicht in administrative Verantwortungsbereiche der einzelnen Institutionen eingegriffen werden.

Dennoch kann der Prototyp noch an einigen Stellen erweitert und verbessert werden. So fehlt derzeit noch der Dienste-basierte Audio-Rückkanal aus der virtuellen in die Präsenz-umgebung. Dieser ist derzeit nicht zwingend notwendig – entsprechende Clients bzw. Browser erfüllen diese Aufgabe zufriedenstellend – aber im Sinne erhöhter Flexibilität und Systematik wünschenswert. Auch eine unmittelbare Integration von SOA-Mechanismen in virtuelle Welten wie Second Life würde die Flexibilität des Ansatzes deutlich steigern, ebenso wie eine Erweiterung der virtuellen Umgebungen um weitere Medien (z. B. HTML und PDF).

Zusätzliche Probleme existieren auf der Seite der SOA. Beispielsweise wurde durch den SOA-Hype in den vergangenen Jahren eine Reihe weitestgehend inkompatibler Technologien entwickelt. Im Prototyp werden Web Services verwendet. Damit sind jedoch alternative und dennoch verbreitete Technologien ausgegrenzt (z.B. Jini, UPnP und Bonjour). Durch einen Mechanismus zur Überbrückung einzelner SOA-Inseln kann das Spektrum der nutzbaren Dienste für die Lehre deutlich erweitert werden [Zen08].

Weiterhin kann die Interaktion der Lehrenden und Lernenden mit den vorgestellten Lernsystemen erheblich intuitiver gestaltet werden als dies derzeit der Fall ist. Dafür erscheint die Verschmelzung des vorgestellten Ansatzes mit Prinzipien und Technologien der Selbstorganisation oder des Pervasive Computing vielversprechend [LT08].

Aus nicht-technischer Sicht bleibt darüber hinaus zu untersuchen, ob durch eine derartig enge Verzahnung von virtueller und Präsenzlehre weiterhin eigene didaktische Strategien für die beteiligten Lehr- bzw. Lernformen benötigt werden, ob die traditionelle Didaktik der Präsenzlehre auch hier anwendbar ist, oder ob wiederum ein völlig neuer didaktischer Ansatz erforderlich wird. Die Klärung dieser Frage wird umfangreiche Praxistests mit intensiver Begleitung aus pädagogischer Perspektive erfordern.

## Literatur

- [All05] C. Allison, R. Nicoll, M. Bateman, B. Ling: „From Web To Grid: Evolution of a Learning Environment“, 4th IASTED International Conference on Web Based Education (WBE'05), 2005. <http://www.elegi.org/>
- [Bal01] K. Ballinger, P. Brittenham, A. Malhotra, W. A. Nagy, S. Pharies: „Web Services Inspection Language (WS-Inspection) 1.0“, GXA Specification, November 2001. <http://www.serviceoriented.org/ws-inspect.html>
- [Bur02] W. Burr: „Service-Engineering bei technischen Dienstleistungen – eine ökonomische Analyse der Modularisierung, Leistungstiefengestaltung und Systembündelung“. Neue betriebswirtschaftliche Forschung, Bd. 286, Deutscher Universitäts-Verlag, 2002.
- [dat08] data-quest GmbH: „Stud.IP Portalseite“, letztes Update: 2008. <http://www.studip.de/>

- [Fre07] P. Freudenstein, F. Majer, A. Maurer, D. Ried, W. Juling: „Wiederverwendungsorientierte Dienste für Universitäten“, Workshop 'Pervasive University' im Rahmen der 37. GI-Jahrestagung, September 2007.
- [Glä08] M. Gläser, R. Zender, U. Lucke, D. Tavangarian: „Service-basierte Integration dynamischer, interaktiver Medien in die Lernplattform Stud.IP“, Die 6. e-Learning Fachtagung Informatik (DeLFI 2008).
- [Hür08] W. Hürst: „Automatic lecture recording for lightweight content production“, in M. Pagani (Hrsg.): „Encyclopedia of Multimedia Technology and Networking“, 2nd edition, 2008.
- [KL06] J. Kemp, D. Livingstone: „Putting a Second Life 'Metaverse' Skin on Learning Management Systems“, Second Life Education Workshop at SL Community Convention, August 2006.
- [KW06] D. Kuroпка, M. Weske: „Die Adaptive Services Grid Plattform: Motivation, Potential, Funktionsweise und Anwendungsszenarien“, Emisa Forum, Nr. 26/1, Januar 2006. <http://asg-platform.org/>
- [Lin08a] Linden Lab: „Second Life: Official site of the 3D online virtual world“, letztes Update: 2008. <http://secondlife.com/>
- [Lin08b] S. Lindemann, T. Reichelt, R. Zender, U. Lucke, D. Tavangarian: „eLearning im Second Life: Mehr als nur eine digitale Kopie der Realität“, eingereicht zum Workshop E-Learning 2.0 im Rahmen der DeLFI 2008.
- [LT07] U. Lucke, D. Tavangarian: „Aktueller Stand und Perspektiven der eLearning-Infrastruktur an deutschen Hochschulen“, Die 5. e-Learning Fachtagung Informatik (DeLFI 2007), September 2007.
- [LT08] U. Lucke, D. Tavangarian: „Mehr als Mobiles Lernen – Innovative Infrastrukturen im Dienste für Pervasive Learning“. Zeitschrift für e-Learning, Heft IV/2007, S. 29-41, Studienverlag, vorauss. Februar 2007.
- [Krö06] R. Kröger, U. Lucke, M. Schmid, D. Tavangarian: „Web Services for the Integration of XML-based Content into Learning Platforms: a Three-level Model“, 2nd Int. Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2006), November 2006
- [Mel07] I. Melzer: „Service-orientierte Architekturen mit Web Services : Konzepte – Standards – Praxis“, 2. Auflage, Spektrum Akademischer Verlag, 2007.
- [ML07] A. Müller, M. Leidl: „Virtuelle (Lern-)Welten: Second Life in der Lehre“, Portal e-teching.org, Dezember 2007.  
[http://www.e-teaching.org/didaktik/gestaltung/vr/SL\\_lehre\\_langtext\\_071207\\_end.pdf](http://www.e-teaching.org/didaktik/gestaltung/vr/SL_lehre_langtext_071207_end.pdf)
- [Rea08] Real Networks Inc.: „Media Servers“, letztes Update: 2008.  
[http://www.realnworks.com/products/media\\_delivery.html](http://www.realnworks.com/products/media_delivery.html)
- [Sch98] H. Schulzrinne, A. Rao, R. Lanphier: „RFC2326: Real Time Streaming Protocol (RTSP)“, IETF Network Working Group, April 1998.
- [Zen08] R. Zender, E. Dressler, U. Lucke, D. Tavangarian: „Meta-Service Organization for a Pervasive University“, 4th IEEE Int. Workshop on Pervasive Learning (PerEL) im Rahmen der 6th IEEE Int. Conference on Pervasive Computing and Communication (PerCom), März 2008.