

# Lernfeldübergreifender Zugang zu Betriebssystemen und zur Netzwerktechnik mit den Methoden der objektorientierten Programmierertechnik für die IT Berufe

Dietmar Johlen

Oskar-von-Miller-Schule Kassel  
Weserstr. 7  
34125 Kassel  
d.johlen@ovm-kassel.de

**Abstract:** Die Administration von Betriebssystemen (OS) und der Betrieb von Netzwerken stellen wichtige Handlungsfelder für Auszubildende der IT Berufe dar. Im Gegensatz zur Programmierertechnik wurden für diese Gebiete bisher erst wenige handlungsorientierte Unterrichtskonzeptionen nach dem Lernfeld-Konzept vorgestellt.

Im Rahmen dieser Arbeit wird ein Zugang zu Betriebssystemen und zur Netzwerktechnik über die Methoden der objektorientierten Programmierertechnik (OOP) vorgestellt. Es wird gezeigt, wie mit diesem Ansatz lernfeldübergreifender Unterricht gefördert wird. Anhand eines Referenzszenarios wird eine Handlungssystematik vorgeschlagen.

## 1 Einleitung

Der Ausbildungsberuf Fachinformatiker gliedert sich in die Schwerpunkte Anwendungsentwicklung und Systemintegration. Neben der Programmierertechnik (Lernfeld „Entwickeln und Bereitstellen von Anwendungssystemen“) bilden Betriebssysteme und Netzwerktechnik (Lernfeld „Vernetzte IT Systeme“) weitere Schwerpunkte der Ausbildung. Während im Bereich Programmierertechnik durch die Organisation des Unterrichts nach dem Vorgehensmodell des objektorientierten Softwareentwicklungsprozesses (OOSE) in Verbindung mit der Metasprache UML handlungsorientierte Konkretisierungen des betreffenden Lernfeldes vorliegen (z.B. [Jo02]), steht eine entsprechende Ausarbeitung für das Lernfeld „Vernetzte IT-Systeme“ in Anlehnung an den Lernfeldansatz noch aus. Im schulischen Alltag werden hier die Inhalte häufig fachsystematisch in eigenständigen Unterrichtslektionen unabhängig voneinander vermittelt. Typische Themen sind z.B. Subnetting zur Strukturierung von IP Adressräumen, Ressourcenfreigabe in Netzwerken und Benutzerverwaltung. Das Ausfüllen der vom Betriebssystem für diese Anwendungen bereitgestellten Masken erschwert häufig ein tieferes Verständnis der zugrunde liegenden Vorgänge. Da häufig eine Trial and Error Vorgehensweise zum „Ziel“ führt, wird ein solches tieferes Verständnis nicht immer honoriert. Damit ist die Gefahr groß, dass so erlernte Kenntnisse in komplexen Anforderungssituationen der Praxis nicht hinreichend dekontextualisiert werden können.

Die Arbeit stellt einen Ansatz vor, der das überaus erfolgreiche Konzept der Objektorientierung in Verbindung mit der Methodik der OOSE im Lernfeld „Vernetzte IT Systeme“ auf Betriebssysteme und Netzwerktechnik anwendet. Diese Übertragung ist mehr als eine rein formale Analogie. Es muss vielmehr festgehalten werden, dass Betriebssysteme, z.B. Windows 2000, eine objektbasierte Architektur aufweisen.

Damit kann z.B. die Benutzerverwaltung auf eines der Kernkonzepte der OOP, der Kommunikation zwischen Objekten, abgebildet werden. Die Begriffe Klasse, Objekt, Methode, Attribut sind übertragbar. Die Beziehungen der Klassen untereinander lassen sich übersichtlich in einem Klassendiagramm darstellen. Sequenzdiagramme eignen sich, um dynamische Vorgänge, z.B. das Anlegen eines neuen Benutzers, zu verdeutlichen.

Die Arbeit setzt sich zum Ziel, die gestalterischen Aspekte des Lernfeldes „Vernetzte IT-Systeme“ in Verbindung mit der Vermittlung von Grundprinzipien der Informatik herauszuarbeiten. Zu diesem Zweck wird ein Referenzszenario für die lernfeldübergreifende Unterrichtsgestaltung vorgeschlagen. Ausgewählte Beispiele, die sich aus dem Referenzszenario ergeben, illustrieren den vorgeschlagenen Ansatz.

Die methodisch-didaktische Konzeption verlangt nicht zwingend spezifische Vorkenntnisse. Sie eignet sich daher besonders als Einführungsveranstaltung zu Beginn der Ausbildung<sup>1</sup>. Grundlegende Kenntnisse der OOSE und zur Modellierungssprache UML lassen sich sinnvoll im Unterricht parallel erarbeiten. So wird die OOP in ihrem Kern als Kommunikation zwischen Objekten für die Schüler erfassbar. Im Rahmen dieses Zugangs wird die strukturierte Programmieretechnik als Untermenge der OOP behandelt. Ebenso verhält es sich mit Vorkenntnissen im Bereich Betriebssysteme und Netzwerktechnik. Es wird die Zielsetzung verfolgt, das Verhalten von Betriebssystemen und Netzwerken aus UML Diagrammen zu entnehmen bzw. in ihnen zu formulieren.

Das folgende Kapitel erläutert kurz das Lernfeld-Konzept und beschreibt die Anforderung an eine Ausgestaltung eines Lernfeldes zu konkreten Lernsituationen unter Einbeziehung des Bildungs- und Qualifizierungsauftrags der Berufsschule. Kapitel 3 stellt ein mögliches Referenzszenario für den Unterricht im Lernfeld „Vernetzte IT Systeme“ vor. Auf dieser Grundlage werden im Kapitel 4 typische Anwendungsfälle im Bereich der Administration von Betriebssystemen diskutiert. Konkret wird die Automatisierung der Mitarbeiterverwaltung durch die Sprache VBScript im Verzeichnisdienst Active Directory von Windows 2000 vorgestellt. Kapitel 5 zeigt die engen Bezüge zwischen Betriebssystem und Netzwerkarchitektur auf, die sich aus dem gewählten Referenzszenario ergeben. Hier wird exemplarisch die Veranschaulichung der Kommunikation zwischen zwei Rechnern (Client-Server-Kommunikation) über ein Netzwerk mit Hilfe eines Sequenzdiagramms herausgegriffen. Die Arbeit schließt mit einer kurzen Zusammenfassung und gibt einen Ausblick auf weitere Unterrichtsvorhaben.

---

<sup>1</sup> An der Oskar-von-Miller-Schule in Kassel wird dieser Zugang mit Fachinformatikern im 1. Ausbildungsjahr und Studierenden der Fachschule für Technik mit dem Schwerpunkt Computersystem- und Netzwerktechnik im ersten Semester beschritten.

Anhand dieser einleitenden Bemerkungen werden die folgenden Thesen für einen zeitgemäßen Unterricht im Lernfeld „Vernetzte IT Systeme“ für Unterricht zu Betriebssystemen und zur Netzwerktechnik aufgestellt:

- Die Methoden der OOP fördern die Planung und Dokumentation von Betriebssystemen und Netzwerken.
- Das Paradigma der OOP ist geeignet, als verbindende, konzeptionelle Klammer um Programmier-, Betriebssystem- und Netzwerktechnik zu dienen.
- Die Verbindung von Unterricht durch ein lernfeldübergreifendes Beispielszenario verkürzt die Einarbeitungszeit der Schüler nach jedem Zyklus im Blockunterricht.
- Lernfeldübergreifender Unterricht, der in ein Beispielszenario eingebettet ist, stärkt das Verständnis grundlegender Prinzipien der Informatik.
- Lernfeldübergreifender Unterricht, der die Verknüpfung zwischen benachbarten Lernfeldern aufzeigt, unterstützt die Teamstrukturen in der Schule.

## **2 Das Lernfeld-Konzept - Reaktion auf das Transferproblem**

Das Lernfeld-Konzept ist als ein möglicher Reformansatz zur Überwindung des Transferproblems anzusehen. Beim Transferproblem handelt es sich um die Schwierigkeiten, die bei der Übertragung fachsystematisch erworbener Kenntnisse auf praxisrelevante Anwendungssituationen auftreten. Durch das Lernfeld-Konzept soll die Vermittlung von Handlungskompetenz in der beruflichen Bildung gefördert und handlungsorientierter Unterricht curricular abgesichert werden. Das Lernfeld-Konzept geht von Handlungsfeldern der Praxis aus, die von beruflicher, gesellschaftlicher und persönlicher Relevanz sind. Diese Handlungsfelder werden durch eine didaktische Aufbereitung und Reflexion in Lernfelder für die Schule überführt. Hierbei ist der Bildungsauftrag der Berufsschule zu berücksichtigen. Er verlangt die Förderung der Persönlichkeitsentwicklung hin zu sozialer Verantwortung in Verbindung mit der ganz konkreten Qualifizierung für eine berufliche Tätigkeit, die auf dem Arbeitsmarkt nachgefragt wird (Qualifizierungsauftrag der Berufsschule). Die Verbindung dieser beiden Ziele der beruflichen Bildung schafft die „Herausbildung beruflicher Identität als Komponente von sozialer Integration“ [Ba00]. Der Bildungsauftrag wirkt als Brücke zwischen Handlungsfeld und Lernfeld. Auf dieser Grundlage ist es Aufgabe der Schulen, die vom Rahmenlehrplan vorgegebenen Lernfelder in Form von Lernsituationen auszugestalten und zu konkretisieren [BS98]. So soll die Umsetzung von erlernten Inhalten in praktischen Anwendungssituationen, in anderen Worten „Vom Wissen zum Können“ [SI00], erleichtert werden.

Die im Mai 1996 verabschiedete Handreichung zur Erstellung von Rahmenlehrplänen nach Lernfeldern [KMK96] bildet auch die Grundlage für den Rahmenlehrplan des Ausbildungsberufs Fachinformatiker [RLP97].

Die Verlagerung von Teilen der curricularen Gestaltung auf die Schulen bei der Ausgestaltung der Lernfelder hat tiefgreifende Konsequenzen. Die Absage an eine fachsystematische Organisation des Unterrichts und die Auflösung des Fächerkanons hinterlässt eine Lücke, die erst noch durch eine zu erarbeitende handlungssystematische Strukturierung gefüllt werden muss. Dies gilt insbesondere für das vorgestellte Lernfeld. Neue fachwissenschaftliche Inhalte müssen integrativ im handlungsorientierten Kontext

übernommen werden. Lehrer müssen sich mit den Handlungsfeldern der Praxis vertraut machen. Hierfür schafft eine enge Lernortkooperation zwischen der Schule und den Ausbildungsbetrieben die Grundlage. Der Unterricht im Lernfeld erfolgt in Lehrerteams. Die Herausarbeitung geeigneter Unterrichtsszenarien, die lernfeldübergreifenden Unterricht fördern, trägt zur Teambildung bei. Zur Bewertung dieser im Team erarbeiteten Lernsituation eignen sich besonders Erschließungsfragen zum Lernfeld-Konzept (z.B. [BS98] und [SI00]).

Der Rahmenlehrplan für den Ausbildungsberuf Fachinformatiker enthält keine methodischen Vorgaben für den Unterricht. Gleichwohl wird betont, dass handlungsorientierte Unterrichtsformen „angemessen berücksichtigt“ werden sollen [RLP97, S. 2]. Für das Lernfeld „Vernetzte IT Systeme“ fordert der Rahmenlehrplan [RLP97, S. 10]:

*Die Schülerinnen und Schüler sollen vernetzte IT-Systeme in Einzel- oder Teamarbeit unter Berücksichtigung von Kundenanforderungen und Beachtung gesetzlicher und sicherheitstechnischer Bestimmungen planen, Komponenten begründet auswählen, installieren, konfigurieren, in Betrieb nehmen, dokumentieren, präsentieren und handhaben.*

### **3 Vorstellung eines Referenzszenarios für das Lernfeld „Vernetzte IT Systeme“**

Im Folgenden wird ein mögliches Referenzszenario für den Unterricht im Lernfeld „Vernetzte IT Systeme“ ansatzweise vorgestellt. Das Referenzszenario versucht eine notwendige Mindestkomplexität für die im Unterricht behandelten Lernsituationen zu definieren. Beispielsweise wird der Mehraufwand einer Unterteilung von IP Adressräumen (Subnetting) erst rechtfertigbar, wenn hierdurch ein Mehrwert im Rahmen einer übergeordneten Netzwerkstruktur erkennbar wird.

Das Referenzmodell erleichtert durch seine narrative Einbettung die methodisch-didaktische Aufbereitung handlungssystematischer Vorgehensweisen, z.B. bei der Erstellung von Richtlinien zur Benutzerverwaltung, die konkrete Anforderungen der Beispielfirma aufgreift. Die Inhalte zu Betriebssystemen und zur Netzwerktechnik werden hierdurch in Beziehung gesetzt.

**Referenzszenario:** Die Beispielfirma ist ein international operierendes Unternehmen mit mehreren Standorten im In- und Ausland. Die Betriebssystem- und Netzwerkinfrastruktur ist in der Lage, im Rahmen der Firmenentwicklung anstehende Zukäufe von Unternehmen zu integrieren und abgestoßene Unternehmensteile aus der Gesamtstruktur auszugliedern. Die im Zuge der technischen Weiterentwicklung notwendige Migration zu neuen Betriebssystemversionen und Netzwerkkomponenten knüpft evolutionär an die bestehenden Strukturen an. Diese Vorgänge werden auf der Grundlage der Dokumentation der bestehenden Betriebssystem- und Netzwerkstrukturen geplant und durchgeführt.

Durch die Erarbeitung unternehmensweiter Richtlinien zur Benutzerverwaltung, Ressourcenverwaltung, Netzwerkstrukturierung und Datensicherheit wird die Wartbarkeit und Erweiterbarkeit der IT Infrastruktur unterstützt. Für einzelne Client Rechner erlauben benutzerprofiltypische Betriebssysteminstallationen eine vereinfachte Administration. Auf der Serverebene werden für die unterschiedlichen Anforderungen optimierte Netzwerkbetriebssysteme eingesetzt, deren Konfiguration und Administration weitgehend automatisiert wird. Durch eine serverseitige Speicherung der Benutzerprofile findet ein Benutzer im gesamten Unternehmen die gewohnte Arbeitsumgebung vor.

Die Arbeit eines Systemadministrators setzt damit beim Modellieren von Teilen der IT Infrastruktur an und verlangt in besonderer Weise gestalterische Tätigkeiten. In [Sc01] wird die Bedeutung dieses Aspekts für die Didaktik der Informatik hervorgehoben, die hier in gleicher Weise gelten. Die gestalterische Komponente der Modellierung verlangt mit ihrem Dreiklang Planen-Urteilen-Entscheiden von den Schülern Verantwortung zu übernehmen, wodurch in besonderem Maße die vom Bildungsauftrag unterstrichene Sozialkompetenz gefördert wird [Ka02]. Im Hinblick auf den Bildungsauftrag der Berufsschule sind Arbeiten zur Didaktik der Informatik für allgemeinbildende Schulen anwendbar (siehe z.B. [Hu01]).

Es bietet sich im Unterricht an, das Vorgehen zur Planung und Implementierung einer unternehmensweiten IT Infrastruktur am auf die Situation in der Schule angepassten Vorgehensmodell [Jo02] der OOP anzulehnen. Die darin enthaltenen Phasen sind jedoch auch in diesem Zusammenhang kein Selbstzweck. Es gilt weiterhin, dass „Modellierung und Strukturierung“ so eingesetzt werden, dass sie einerseits nicht zum „philosophischen Exkurs“ überhöht werden oder im anderen Extrem „spezifische Eigenheiten der verwendeten Programmiersprache (im Rahmen dieser Arbeit heißt das: spezifische Eigenheiten des verwendeten Betriebssystems) in den Mittelpunkt des Unterrichts rücken“ ([Hu99], S. 24f).

#### **4 Automatisierung der Betriebssystemadministration**

Im Rahmen des zuvor beschriebenen Szenarios ergeben sich typische Tätigkeiten für Systemadministratoren. Eine Auswahl ist in Abb. 1 in der Form einzelner Anwendungsfälle in einem Anwendungsfalldiagramm zusammengefasst.

Aus der Analyse der Anforderungsbeschreibung an eine IT Infrastruktur einer Firma modellieren die Schüler mit Blick auf die Betriebssystemseite eine Domänenstruktur. Ein Teil dieses Modells ist in Abb. 2 vereinfacht wiedergegeben. In diesem Kapitel wird vorgestellt, wie eine Domäne durch die Einführung von Organisationseinheiten (OU) den Anforderungen der jeweiligen Geschäftsprozesse angepasst werden kann. Innerhalb einer OU werden durch die Zusammenfassung von Benutzern zu Gruppen die Voraussetzungen geschaffen, um Teile von Administrationsaufgaben an diese Gruppen zu delegieren.

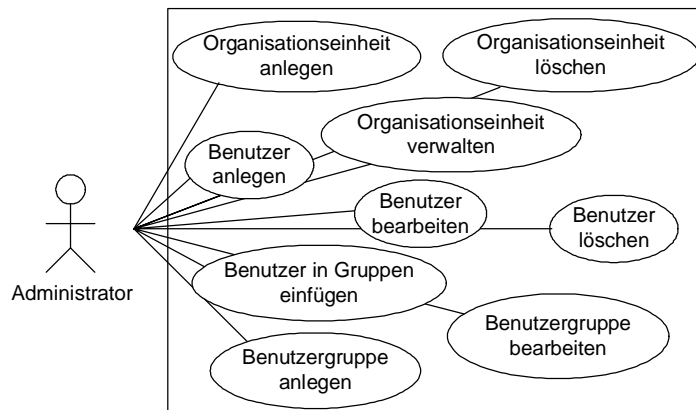


Abb. 1: Typische Aufgaben eines Systemadministrators, die als Anwendungsfälle in einem Anwendungsfalldiagramm dargestellt sind.

Die konkrete Umsetzung dieser Administrationsaufgaben wird für den Verzeichnisdienst Active Directory<sup>2</sup> (AD) des Betriebssystems Windows 2000 am zugehörigen Objektmodell demonstriert. An die Stelle der im Unterricht häufig üblichen „Abarbeitung“ dieser Tätigkeiten über Assistenten und Masken der graphischen Benutzeroberfläche (GUI) des OS tritt das unmittelbare Erfahren der „eigentlichen Vorgänge“, durch den direkten Umgang mit den zugrunde liegenden Objekten des OS. Der Schlüssel für dieses tiefere Verständnis liegt in der Anwendung der Methoden der OOP.

---

<sup>2</sup> Der Verzeichnisdienst Active Directory ermöglicht z.B. die Veröffentlichung von Ressourcen im Netzwerk, die Benutzerverwaltung und ist zudem die Grundlage für Programme wie Web- und Datenbankserver.

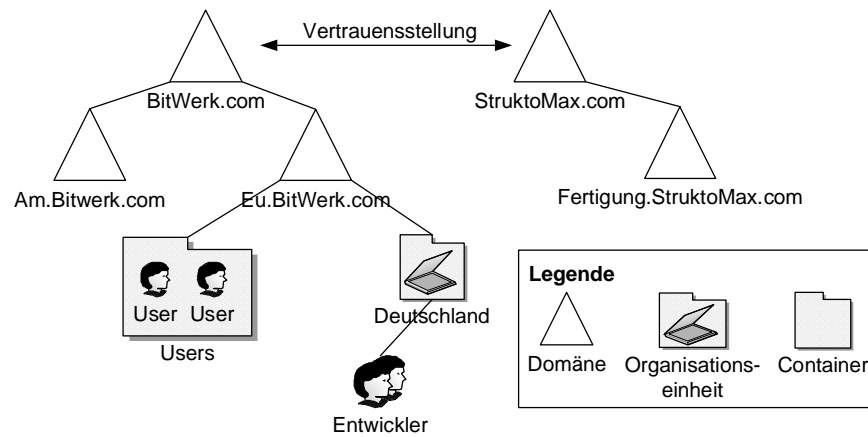


Abb. 2: Ausschnitt der Domänenstruktur der Beispielfirma BitWerk.com. Die Domäne Eu.BitWerk.com der europäischen Niederlassung wird durch eine Organisationseinheit (OU) Deutschland weiter strukturiert. Innerhalb der OU Deutschland werden Mitarbeiter der Firma, die Entwickler sind, in einer Gruppe zusammengefasst.

Der Verzeichnisdienst organisiert Objekte in einer hierarchischen Struktur. In dieser Struktur werden zwei Arten von Objekten unterschieden: Container und Nichtcontainer. Ausgehend von einem Wurzelcontainer, enthalten Container wieder Container und Nichtcontainer. Ein Nichtcontainer enthält keine weiteren Objekte. Ein Beispiel für ein Container Objekt ist eine Organisationseinheit. Ein Benutzer stellt ein Nichtcontainer-Objekt dar. Für den Zugriff auf diese Objekte in der Sprache VBScript stellt das Active Directory Service Interface (ADSI) Interfaceklassen zur Verfügung. Für die hier diskutierten Beispiele sind von Bedeutung: IADsContainer, IADs, IADsGroup und IADsMembers. Eine ausführliche Darstellung befindet sich z.B. in [Lo00] und [Sc00]. Die folgenden Beispiele sollen den Zugang zur Betriebssystemadministration über die OOP verdeutlichen.

Zum Anlegen eines Benutzers in der Domäne Eu.BitWerk.com wird mit der Methode GetObject() eine Verbindung zu dem standardmäßig bei der Windows Installation angelegten Users Container hergestellt. Diese Verbindung wird mittels ADSI im VBScript durch die Instanz adsBenutzerContainer der Interfaceklasse IADsContainer dargestellt. Als Argument dient der eindeutige AD Pfadname zu der Ressource Users, die über das Lightweight Directory Access Protocol (LDAP) angesprochen wird.

```
Set adsBenutzerContainer = GetObject("LDAP://cn=Users,dc=Eu,dc=BitWerk,dc=com")
```

Der AD Pfad ergibt sich aus der Domänenstruktur (siehe Abb. 2) ausgehend von der Wurzel bis zur gewünschten Ressource. In diesem Fall über die Domain Components (dc) com, BitWerk und EU bis zum Benutzer Container mit dem Common Name (cn) Users.

Für IADsContainer steht die Methode Create() bereit, um Objekte zu erzeugen. Der folgende Aufruf zeigt die Erzeugung eines Benutzer Objekts adsBenutzer (Argument "user") mit dem Namen Sandra Schulz.

```
Set adsBenutzer = adsBenutzerContainer.Create("user", "cn=Sandra Schulz")
adsBenutzer.SetInfo
```

Mit dem Aufruf adsBenutzer.SetInfo wird dieser Vorgang im AD Verzeichnisdienst bekannt gemacht. Hier wird durch das Präfix „ads“ das Objekt adsBenutzer als ADSI Objekt gekennzeichnet.

Aus Administrationszwecken wird für die Mitarbeiter der deutschen Niederlassung von BitWerk.com eine eigene OU eingerichtet.

```
adsContainer = GetObject("LDAP://dc=EU,dc=BitWerk,dc=com")
Set adsDeutschlandOU = adsContainer.Create("organizationalUnit",
"ou=Deutschland")
adsDeutschlandOU.SetInfo
```

Mit der GetObject() Methode wird die Verbindung zur Domäne EU.BitWerk.com hergestellt, in der die OU Deutschland über den Methodenaufruf Create() eingerichtet wird. Die Create Methode erkennt am Argument "organizationalUnit", dass ein OU Objekt angelegt werden soll.

Der folgende Befehl legt in der OU Deutschland eine Gruppe mit dem Namen Entwickler an:

```
Set adsGruppe = adsDeutschlandOU.Create("group", "cn=Entwickler")
adsGruppe.SetInfo
```

Da es sich bei der Benutzerin Sandra Schulz um eine Entwicklerin handelt, die in Deutschland arbeitet, wird sie dieser Gruppe hinzugefügt:

```
adsGruppe.Add("LDAP://cn=Sandra Schulz,cn=Users,dc=EU,dc=BitWerk,
dc=com")
```

Ein weiteres Beispiel für die dem Betriebssystem zugrunde liegende Objektstruktur ist der Umgang mit dem Component Object Model<sup>3</sup> (COM). Ein Anwendungsprogramm wird als Instanz der Klasse Word.Application dargestellt. Das Programm lässt sich vollständig über den Aufruf von Methoden und das Setzen von Attributen steuern:

```
Set meinWord = CreateObject("Word.Application")
meinWord.Visible = True
Set neuesDok = meinWord.Documents.Add
meinWord.Selection.TypeText "Betriebssysteme verstehen mit Hilfe der OOP!"
```

---

<sup>3</sup> Die zuvor dargestellte Betriebssystemverwaltung beruht auf der Abbildung des Verzeichnisdienstes auf COM Klassen.



Mit diesem VBScript wird Microsoft Word mit einem neuen Dokument geöffnet und der Text „Betriebssysteme verstehen mit Hilfe der OOP!“ wird in der ersten Zeile ausgegeben.

## **5 Zugang zur Netzwerktechnik**

Die zuvor beschriebenen gestalterischen Aspekte der Planung, Weiterentwicklung und Administration von Betriebssystemen sind in gleicher Weise auf den Entwurf der zugehörigen Netzwerke übertragbar. Als Beispiel sei die Planung der Zonen und Knoten einer Domain Name Service (DNS) Hierarchie genannt, die in engem Zusammenhang zu der in Abb. 2 gezeigten Domänenstruktur des Betriebssystems steht.

Die Methoden der OOP, insbesondere die Notation UML, ist hilfreich, um z.B. Vorgänge bei der Kommunikation zwischen Rechnern in einer einheitlichen Schreibweise skalierbar zu visualisieren. Abb. 3 zeigt in einem Sequenzdiagramm den Datenaustausch zwischen zwei Anwendungen über das TCP/IP Protokoll. Unter Skalierbarkeit wird hier verstanden, dass in der nächsten Detaillierungsstufe z.B. die Kommunikation im Protokoll Stack mit in das Diagramm einbezogen wird.

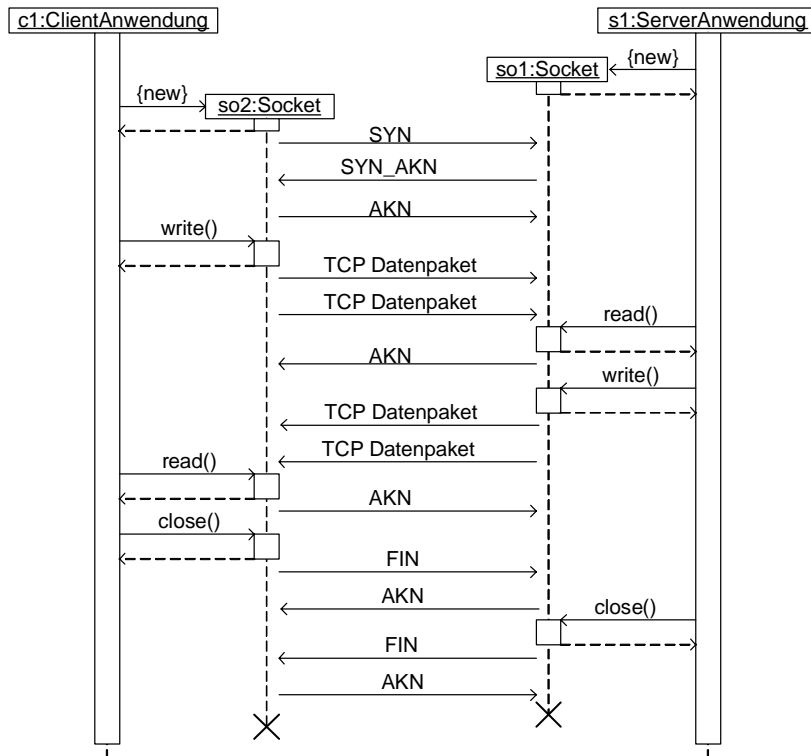


Abb. 3: Sequenzdiagramm für die Kommunikation zwischen zwei Anwendungen über das TCP/IP Protokoll. Mit read() und write() Methoden werden Datenpakete auf Socket Objekte geschrieben, die die IP und Port Nummer des jeweiligen Rechners symbolisieren.

## 6 Zusammenfassung und Ausblick

Das vorgestellte Beispiel zur Automatisierung der Administration komplexer Betriebssystemstrukturen belegt die Anwendbarkeit der Methoden der OOP. Für Planung und Dokumentation eignet sich das Vorgehensmodell der OOSE. Der vorgestellte lernfeldübergreifende Zugang zu Betriebssystemen und zur Netzwerktechnik hat insbesondere die Fachinformatiker mit der Fachrichtung Systemintegration im Blick, für die der Rahmenlehrplan verstärkt Inhalte zu Betriebssystemen und zur Netzwerktechnik vorsieht. Auf diese Weise lassen sich die Methoden der OOP auch diesen Schülern verstärkt näher bringen. Diese Überlegung trifft auch für Studierende der Fachschule für Technik mit dem Schwerpunkt Computersystem- und Netzwerktechnik zu. Erste Erfahrungen mit dem Zugang zu Betriebssystemen über die OOP belegen diese These. Der Einsatz eines lernfeldübergreifenden Beispielszenarios verspricht zudem die Abkürzung der Wiedereinführungsphasen zu Beginn einer Blockwoche. Zudem ist die gemeinsame Absprache eines Beispielszenarios eine wichtige Voraussetzung im Prozess der

Teambildung an den Schulen. Die weitere Vorbereitung auf eine lernfeldübergreifende Pilotunterrichtsreihe in Anlehnung an das vorgestellte Referenzszenario beinhaltet als nächste Schritte die Veranschaulichung der komplexen Betriebssystemstruktur und ihrer Abbildung im ADSI in Form eines Klassendiagramms. Die Anwendungsfälle der Betriebssystemadministration werden auf dieser Grundlage durch Sequenzdiagramme modelliert.

## Literaturverzeichnis

- [Ba00] Bader, R.: Stand der wissenschaftlichen Forschung zum Lernfeld-Konzept. (<http://www.uni-magdeburg.de/ibbp/bp/downloads/Lernfeld-Konzept.pdf>), 2000.
- [BS98] Bader, R.; Schäfer, B.: Lernfelder gestalten – Vom komplexen Handlungsfeld zur didaktisch strukturierten Lernsituation. In: Die berufsbildende Schule (BbSch), 50 (1998), 7-8.
- [Hu99] Hubwieser, P.: Modellieren in der Schulinformatik. In: Log In, Vol. 19, Nr.1, 1999, S. 24-29.
- [Hu01] Hubwieser, P.: Didaktik der Informatik: Grundlagen, Konzepte, Beispiele. 1. korrigierter Nachdruck. Springer, Berlin, 2001.
- [Jo02] Johlen, D.: Methodik der OOSE für Fachinformatiker nach dem Lernfeldansatz unter Einbeziehung der Lehrerfortbildung. In: Forschungsbeiträge zur „Didaktik der Informatik“ – Theorie, Praxis, Evaluation, Proceedings 1. GI-Workshop DDI'02, Witten-Bommerholz, S. 55-64, 10.-11. Okt. 2002.
- [Ka02] Kath, F.M.: Paradigmenwechsel auch in der Fachdidaktik – Wunsch oder Realität? In: Die berufsbildende Schule (BbSch), 54 (2002), 4.
- [Lo00] Lowe-Norris, A.: Windows 2000 Active Directory. O'Reilly Verlag, Farnham, 2000.
- [KMK96] KMK – Sekretariat der Ständigen Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland: Handreichungen für die Erarbeitung von Rahmenlehrplänen der Kultusministerkonferenz für den berufsbezogenen Unterricht in der Berufsschule und ihre Abstimmung mit Ausbildungsordnungen des Bundes für anerkannte Ausbildungsberufe. Bonn, 1996.
- [RLP97] KMK – Sekretariat der Ständigen Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland: Rahmenlehrplan für den Ausbildungsberuf Fachinformatiker/Fachinformatikerin. (Beschluss der Kultusministerkonferenz vom 25. April 1997), 1997.
- [Sc00] Schwichtenberg, Holger: Windows-Scripting: Automatisierte Systemadministration mit Visual Basic/VBSript und COM/.NET. Verlag Addison-Wesley, Boston, 2., aktualisierte Auflage, 2000.
- [Sc01] Schulte, C.: Vom Modellieren zum Gestalten – Objektorientierung als Impuls für einen neuen Informatikunterricht. In: informatica didactica 3, 2001 (<http://ddi.cs.uni-potsdam.de/InformaticaDidactica>).

[S100] Sloane, P.F.E.: Lernfelder und Unterrichtsgestaltung. In: Die berufsbildende Schule (BbSch), 52 (2000), 3.