



**Technische  
Universität  
Darmstadt**

# **ISO/OSI-Referenzmodell**

von

Florian Becher und Jonas Steitz

<b>Seminar</b>	Fachdidaktisches Seminar im Hauptstudium Informatik im Schulunterricht
<b>Fachgebiet</b>	Fachdidaktik der Informatik
<b>Dozent</b>	Dr. Ulrike Brandt, Gerhard Röhner
<b>Semester</b>	WS 2006/2007
<b>Vortragsdatum</b>	16.01.2007
<b>Verfasser</b>	Florian Becher, Wirtschaftsinformatik Jonas Steitz, Wirtschaftsinformatik

# I. Inhaltsverzeichnis

I.	Inhaltsverzeichnis.....	2
II.	Abbildungsverzeichnis.....	3
1	Einleitung .....	4
1.1	Motivation der Arbeit .....	4
1.2	Ziel der Arbeit.....	4
1.3	Aufbau der Arbeit .....	4
2	Warum ISO/OSI-Referenzmodell als Wahlthema .....	5
2.1	Didaktische Begründung für die Behandlung .....	5
2.2	Allgemeine Aspekte und Lebensweltbezug des Stoffs.....	8
2.3	Verbindung zu anderen Themen und notwendiges Grundwissen.....	9
3	Das ISO/OSI-Referenzmodell – Grundlagen.....	10
3.1	Anforderungen an Netzwerke und Netzwerkarchitekturen .....	10
3.2	Das Konzept von Schichten- und Referenzmodellen .....	11
3.3	Aufbau des OSI-Referenzmodells.....	12
3.4	Abgrenzung von TCP/IP- und OSI -Referenzmodell .....	15
4	Umsetzung im Informatikunterricht.....	17
4.1	Herangehensweise und Voraussetzungen.....	17
4.2	Rahmenbedingungen und Lehr- und Lernziele .....	18
4.3	Unterrichtsformen, -Inhalt und -Umfang .....	19
4.4	Aufteilung der Unterrichtszeit und Gestaltung der Einheiten.....	20
4.5	Programmierprojekt als Vertiefungsmöglichkeit .....	20
4.6	Wissenskontrolle .....	24
4.7	Bewertung des Konzeptes .....	25
5	Zusammenfassung und Fazit .....	25
III.	Literaturverzeichnis.....	27
IV.	Anlage .....	28

## II. Abbildungsverzeichnis

Abbildung 1: Unterrichtsinhalte und Aufgaben des Wahlthemas Rechnernetze.....	6
Abbildung 2: Beispielhafte Visualisierung eines Drei-Schichtenmodells. ....	11
Abbildung 3: Darstellung aller Schichten des OSI-Referenzmodells. ....	13
Abbildung 4: Vergleich der Schichtenaufteilung von OSI und TCP/IP.....	16
Abbildung 5: Sendevorgang auf Schicht 2 mit Hilfe von Schicht 1. ....	22
Abbildung 6: Programmoberfläche eines „Netzwerk-Roulette“.....	24

# **1 Einleitung**

## **1.1 Motivation der Arbeit**

Diese Arbeit entstand im Rahmen eines Seminars zur Fachdidaktik der Informatik. Als Motivation dieser Arbeit steht die Überlegung, warum die Behandlung des ISO/OSI-Referenzmodell im Schulunterricht sinnvoll und notwendig ist.

Das ISO/OSI-Referenzmodell ist Bestandteil des Themengebietes Rechnernetze welches als Wahlthema im Kurshalbjahr 13/2 behandelt werden kann. Das ISO/OSI-Referenzmodell kann also nicht als solches alleine betrachtet werden. Es bietet sich aber an, das Modell als Rahmen für das Wahlthema Rechnernetze heranzuziehen, womit sich die Frage stellt, wie man anhand des Modells das Wahlthema lehren könnte. Hierfür sollen der hessische Lehrplan selbst, die Fachliteratur, der Lebensweltbezug des vermittelten Stoffs sowie Verbindungen zu anderen schulischen Aktivitäten untersucht und abschließend Vorschläge zur Unterrichtsgestaltung erarbeitet werden.

Die Ergebnisse der Diskussion mit den Kommilitonen, die sich an die Präsentation des Seminarthemas anschloss, sind bereits in diese Arbeit integriert und lieferten wertvolle Anregungen.

## **1.2 Ziel der Arbeit**

Die Arbeit verfolgt zwei Ziele. Zum einen soll eine didaktische Begründung für die Behandlung des ISO/OSI-Referenzmodell gegeben werden und damit verbunden auf den Stellenwert des ISO/OSI-Referenzmodell selbst beziehungsweise des kompletten Wahlthemas Rechnernetze eingegangen werden.

Im Anschluss sollen die notwendigen Grundlagen für die Umsetzung des Themas im Kurshalbjahr 13/2 Grund- und Leistungskurs erläutert sowie konkrete Umsetzungsvorschläge erarbeitet werden. Diese beinhalten ein Grobkonzept für die Unterrichtsgestaltung (Didaktik) sowie mögliche Arbeitsmethoden und Organisationsformen.

## **1.3 Aufbau der Arbeit**

Die Arbeit beginnt in Kapitel 2 mit der didaktischen Begründung des ISO/OSI-Referenzmodell, in der Lehrplan, die Fachliteratur sowie ein allgemeiner Lebenswelt-

bezug untersucht werden. Anschließend führt Kapitel 3 in das zugrunde liegende Fachwissen des ISO/OSI-Referenzmodells ein, erläutert Schichtenmodelle und schließt mit einer Abgrenzung zum TCP/IP-Referenzmodell. Kapitel 4 als Hauptteil der Arbeit widmet sich der konkreten Umsetzung im Informatikunterricht. Dabei steht die Gestaltung der Unterrichtseinheiten im Vordergrund und damit verbunden die angestrebten Lernziele. Zusätzlich werden Anregungen für weitere Vertiefungsmöglichkeiten gegeben.

## 2 Warum ISO/OSI-Referenzmodell als Wahlthema

Dieses Kapitel befasst sich mit der didaktischen Begründung für die Behandlung des ISO/OSI-Referenzmodells als Wahlthema in der Sekundarstufe 13/2. Ausgehend von einer Untersuchung des hessischen Lehrplans Informatik und einer Analyse der einschlägigen Literatur zur Didaktik der Informatik soll die Attraktivität des Wahlthemas aufgezeigt werden. Als besonderer Schwerpunkt wird dabei der Lebensweltbezug des Wahlthemas herausgearbeitet und dabei die Wertigkeit des Wahlthemas für die kommenden Jahre prognostiziert. Das Kapitel schließt mit einer Übersicht über verwandte Themen und diskutiert Ergänzungen und Überschneidungen.

### 2.1 Didaktische Begründung für die Behandlung

#### *Hessischer Lehrplan Informatik*

Im hessischen Lehrplan Informatik [LehHes] ist das Thema ISO/OSI-Referenzmodell ein Bestandteil des Wahlthemas Rechnernetze für das Kurshalbjahr 13/2 und repräsentiert das Modul Kommunikationssysteme. Eine Übersicht über alle Module des Wahlthemas zeigt die folgende Abbildung 1.

Unterrichtsmodul	Inhalte
Kommunikationssysteme	ISO/OSI-Referenzmodell
Prinzipien und Protokolle	CSMA/CD TCP und IP, UDP
Dienste und Anwendungen	Kommunikation über Sockets und Ports POP3, SMTP, HTTP, FTP
Netzwerke	Topologien: Stern, Bus, Ring Arten und Medien: Ethernet (802.3, 802.11), BNC-Kabel, RJ45 (Twisted-Pair)-Kabel, Lichtwellenleiter, Funk Komponenten: Hubs und Switches
Routing	Router, NAT

Sicherheit	Prinzip des Firewalls
Weitere Dienste und Anwendungen	News, Timeserver, Proxies
Server	SQL-Server

Abbildung 1: Unterrichtsinhalte und Aufgaben des Wahlthemas Rechnernetze.

Der erste Kontakt mit dem Thema Rechnernetze findet bereits im Halbjahr 11/1 statt, in der das Internet und die damit verbundenen Netzwerktechnologien besprochen werden.

Betrachtet man die didaktischen Begründungen, die im hessischen Lehrplan Informatik beschrieben sind, so müssen Argumente für das Thema Rechnernetze und Argumente für das ISO/OSI-Referenzmodell unterschieden werden.

Als Argument für die Behandlung des Wahlthemas Rechnernetze wird lediglich erwähnt, dass Rechnernetze für die heutigen Informations- und Kommunikationssysteme die Grundlage bilden, womit der Stellenwert von Rechnernetzen nur implizit angesprochen wird. In der Begründung für das Wahlthema werden besonders technische und sicherheitsrelevante Aspekte von Netzwerken hervorgehoben, was einen aktuellen Trend der Informationstechnologie antizipiert.

Das ISO/OSI-Modell per se wird in der Begründung des Wahlthemas gleich zweimal erwähnt. Der Lehrplan betont zunächst den besonderen Zugpferdcharakter des ISO/OSI-Referenzmodells, welches als Grundlage und Rahmen für alle anderen Inhalte des Wahlthemas herangezogen werden kann („Das ISO/OSI Referenzmodell liefert für die Auseinandersetzung mit dem Thema Rechnernetze einen sinnvollen Rahmen“). Dem ISO/OSI-Referenzmodell wird somit ein besonderer Stellenwert im Vergleich zu den anderen Inhalten des Wahlthemas zugesprochen. Weiterhin wird erwähnt, dass sich das ISO/OSI-Referenzmodell gut dazu eignet, die verschiedenen Probleme und Lösungen der technischen Aspekte von Netzwerkstrukturen darzustellen.

#### *Fachliteratur zur Didaktik der Informatik*

Als Grundlage für die Analyse der Fachliteratur dienten Werke von Baumann [BaRü96], Eberle [EbFr96], Hubwieser [HuPe01] und Humbert [HuLu05], auf die im Folgenden einzeln eingegangen wird. Dabei werden Erwähnungen des Themas Rechnernetze und des ISO/OSI-Referenzmodell gesondert betrachtet.

Rüdeger Baumann schlägt für die Oberstufe sechs Themenkomplexe für die Informatik vor, von denen der dritte den Titel „Datenbanksysteme und Rechnernetze“ trägt. Die Rechnernetze untergliedert er weiter in Computer- und Telekommunikationsnetze. Inhaltlich werden die Themen Computervernetzung, Schichtenmodell (Protokolle, Architektur, ISO/OSI-Referenzmodell), Multimedia im Netz, Hypertextsysteme und Internet-Dienste bearbeitet. Bereits bei Baumann wird dem ISO/OSI-Referenzmodell eine besondere Stellung zugesprochen („Eine Schlüsselstellung nimmt das Referenzmodell für offene Systeme (...), kurz: OSI-Referenzmodell genannt, ein“).

Als didaktische Begründung sieht Baumann die Notwendigkeit des Themenblocks im Wandel zur global vernetzten Informationsgesellschaft und betont dabei die Zunahme der verteilten Anwendungssysteme. Auch sicherheitsrelevante Aspekte werden erwähnt („Äußerst aktuell und wichtig ist ein Thema, das nur im Informatikunterricht behandelt werden kann, nämlich Sicherheit im Datennetz.“).

Franz Eberle geht in seinem Werk zur Didaktik der Informatik nicht explizit auf den Bereich der Rechnernetze ein, umschreibt diesen aber als einen für die Schule relevanten Bereich. Eberle zitiert Steinbock und erwähnt somit implizit das ISO/OSI-Referenzmodell („Herstellerspezifischen Netzwerkarchitekturen steht das ISO/OSI-Referenzmodell als Rahmen für die Entwicklung international anerkannter Kommunikationsstandards gegenüber...“).

Zur didaktischen Begründung des Wahlthemas führt Eberle die wesentliche Bedeutung der komplexen Kommunikationssysteme im wirtschaftlichen, sozialen und politischen Leben an. Er erkennt die zunehmende Verflechtung von Computertechnik und Gesellschaft und möchte diese Vernetzung im Informatikunterricht aufzeigen. In einer weiteren Aussage, in der er die Gesellschaft für Informatik zitiert, werden Rechnernetze sogar mit dem Buchdruck Johannes Gutenbergs verglichen.

Peter Hubwieser wählte in seinem Werk einen anderen Ansatz als seine Vorgänger, er propagiert die Einführung der Informatik als Pflichtfach und wirkte bei der erfolgreichen Einführung des Pflichtfachs Informatik in Bayern mit. Die Inhalte der Informatik und besonders der Rechnernetze streuen sich bei Hubwieser über Sekundarstufe I und II. So werden in der Sekundarstufe I Aufbau und Funktionsweise von Computern und Computernetzen, Konzeption und Planung von lokalen Netzwerken, Rechte- und Verzeichnisstrukturen und Sicherungsstrategien behandelt, wodurch sich der Schüler bereits in einer frühen Phase mit Rechnernetzen auseinandersetzt. Eine Vertiefung des Wissens bietet das Wahlthema „Parallele Prozesse“ in der Sekundarstufe II, welches die Inhalte Kommunikation, Protokolle, einfache Schichtenmodelle, Dienste globaler Netze (Grundkurs) vertieft.

Das Schichtenmodell wird bei Hubwieser in das einfache und das vollständige Schichtenmodell differenziert, welches nur dem Leistungskurs vorbehalten ist. Das einfache Schichtenmodell unterscheidet sich vom vollständigen Modell in der Anzahl der Schichten, wodurch die Komplexität des Schichtenmodells auf Kosten des Detaillierungsgrades der Netzwerkkommunikation reduziert wird.

In seiner didaktischen Begründung greift Hubwieser den Kriterienkatalog zur Allgemeinbildung nach Heymann [siehe HeHa97] auf. Den Erwerb von Grundkenntnissen über die Funktionsweisen von Rechneranlagen und Netzen sieht er als Grundlage für einen kritischen Vernunftgebrauch. Eine weitere Begründung findet er in der Frage: „Für welche Welt bilden wir unsere Schüler aus?“. Er unterteilt dabei vier Bereiche, die *Informationsflut* (welche z.B. mit Hilfe von Rechnernetzen verarbeitet werden kann), den *Arbeitsmarkt* (z.B. das verteilte Arbeiten über Netze), die *wirtschaftliche Bedeutung* (Globalisierung) und die *Allgegenwart* (z.B. Anzahl der Mobilfunkteilnehmer).

Ludger Humberts Werk, welches im Vergleich zu den bereits erwähnten Werken wesentlich jünger ist, fasst die fachdidaktischen Empfehlungen und Entwicklungen der Schulinformatik zusammen. Humbert zeigt besonders den Wandel der Anforderungen und der zu behandelnden Inhalte auf und schlägt eine Neubewertung der Fachstruktur vor, bei der besonders der Interaktionsgedanke betont werden soll. Im Zuge des Wandels spricht er allgegenwärtigen Elementen, wie beispielsweise der RFID (Radio Frequency Identification) Technologie eine höhere Bedeutung zu und möchte sie in den Unterricht integrieren.

## **2.2 Allgemeine Aspekte und Lebensweltbezug des Stoffs**

Der Lebensweltbezug wurde bereits ansatzweise bei den didaktischen Begründungen der Fachliteratur angesprochen und soll nun vertieft werden, um den hohen Stellenwert des Wahlthemas aufzuzeigen. Ausgangspunkt dabei ist der „normale“ Alltag eines Menschen, wie er den Schüler erwarten könnte. Bei der Betrachtung des Alltags muss das Private und die Arbeit gesondert behandelt werden.

Im privaten Bereich gibt es bereits eine Vielzahl von bewussten und unbewussten Berührungen mit dem Thema Rechnernetze. Nahezu unsere gesamte Kommunikation, wie Telefonie und Mobilfunk werden über Rechnernetze (z.B. das Internet) abgewickelt. Technologien wie VoIP (Voice over IP), die Anrufe direkt über das Internet ermöglichen, verzeichnen ein starkes Wachstum. Auch die Verwendung des Me-

diums Internet an sich wächst zunehmend, genaue Zahlen und Tendenzen kann man von der Seite des statistischen Bundesamtes ([www.destatis.de](http://www.destatis.de)) beziehen.

Weiterhin wird eine Vielzahl administrativer Aufgaben die ein Bürger erledigen kann oder muss zunehmend freiwillig oder sogar obligatorisch über das Internet angeboten und abgewickelt. Hier sei beispielsweise auf die elektronische Lohnsteuererklärung (ELSTER) oder das geplante E-Government der Bundesregierung (Verwaltung innovativ, [www.staat-modern.de](http://www.staat-modern.de)) verwiesen.

Im beruflichen Alltag ist der Einzug von Rechnernetzen bereits weiter fortgeschritten als im privaten. Grund dafür ist hauptsächlich der Trend zur Globalisierung. Unternehmen arbeiten heute weltweit und benötigen für die entstehenden Kommunikationsbedürfnisse eine Vernetzung zwischen den eigenen Standorten, sowie zu den Kunden. Weitere Veränderungen in den heutigen Geschäftsprozessen sind ein wachsendes Outsourcing sowie Offshoring, die ebenfalls eine Vernetzung voraussetzen.

Der moderne Lohnabhängige sollte also über die nötigen Kenntnisse im Informatikbereich verfügen, um in der Arbeitswelt bestehen zu können. Dabei sind Kenntnisse über Rechnernetze besonders von Vorteil. Zusätzlich ist sowohl im privaten als auch im beruflichen Bereich das wachsende Bedürfnis nach Sicherheit und Datenschutz ein Thema mit steigender Bedeutung.

Im Informatikunterricht sollten bei der Behandlung von Rechnernetzen auch die gesellschaftlichen Auswirkungen der Anwendung von Informationstechnologien besprochen werden. Insbesondere sind hierbei technische Möglichkeiten als auch Grenzen (momentane oder permanente) aufzuzeigen, die dann auch zur Entmystifizierung der Technik beitragen können.

### ***2.3 Verbindung zu anderen Themen und notwendiges Grundwissen***

Es existieren mehrere Verbindungen zu anderen Disziplinen und Unterrichtsinhalten anderer Fächer, besonders seien genannt:

- Physikalischen Grundlagen der Netze, welche auch im Physikunterricht bearbeitet werden können
- Mediale Beeinflussung und die Qualität von Informationen, die mit der steigenden Informationsflut im Zusammenhang stehen
- E-Government, die elektronische Abwicklung der staatlichen Verwaltung

- E-Business, die elektronische Abwicklung von Geschäftsprozessen
- Kultureller Austausch und Konfliktpotentiale
- Risiken und Missbrauchsmöglichkeiten der neuen Technologien (Big Brother)

### **3 Das ISO/OSI-Referenzmodell – Grundlagen**

Im diesem Kapitel soll eine Einordnung des OSI-Referenzmodells in die Netzwerkarchitektur erfolgen sowie dessen Aufbau und Funktionsweise knapp erläutert werden. Dabei ist es nicht Ziel dieser Arbeit umfangreiches Modell- und Fachwissen zu vermitteln, sondern es geht vielmehr darum gezielt auf Basis der hier gegebenen Modellbeschreibung Möglichkeiten zur Umsetzung und Behandlung des Themas im Unterricht aufzuzeigen.

#### ***3.1 Anforderungen an Netzwerke und Netzwerkarchitekturen***

Heutige Netzwerkarchitekturen und moderne Netzwerke müssen einer Vielzahl von komplexen Anforderungen genügen. Die Spanne reicht dabei von einer Vielzahl physikalischer Komponenten und Barrieren bis hin zu den unterschiedlichsten Applikationen und Diensten, die auf den verschiedenen Betriebssystemen in unserer heutigen heterogenen Gerätelandschaft ausgeführt werden. Ein Modell zur Steuerung der gesamten Kommunikation muss also unabhängig von verschiedenen Übertragungsmedien und Systemplattformen ein allgemeines Vorgehen für einen Datenaustausch beschreiben. Die Komplexität und die gleichzeitig notwendige Flexibilität eines solchen Modells sind sehr groß und machen deshalb den Einsatz von Schichten- und Referenzmodellen notwendig.

Für heutige Netzwerke haben sich vor allem zwei verschiedene Netzwerkarchitekturen entwickelt. Das TCP/IP-Referenzmodell wobei TCP für Transmission Control Protocol und IP für Internet Protocol steht, welche die beiden Basisprotokolle des Modells darstellen. Interessant an diesem Modell ist, dass die Entwicklung mehr dem historischen Wachstum der Netze selbst sowie der aktiven Verwendung dieser Architektur geschuldet ist und weniger von theoretischen Überlegungen geprägt ist.

Auf einem gegenläufigen Vorgehen basiert das Open System Interconnection Reference Model (kurz OSI-Referenzmodell), welches zunächst theoretisch entworfen und erst später implementiert wurde. Es ist ein Sieben-Schichtenmodell, welches eine so genannte offene Kommunikation zwischen beliebigen Netzteilnehmern be-

schreibt und vereinheitliche Verfahren und Regeln zum Datenaustausch definiert. Dieses Modell wird seit 1979 entwickelt und ist inzwischen von der Internationalen Organisation für Normung (ISO) standardisiert. Im Vergleich zu TCP/IP, welches eine sehr große Verbreitung und Verwendung genießt, liegt die Bedeutung des OSI-Referenzmodells vor allem in der Theorie, die sehr sorgfältig definiert ist und bei der Analyse von Netzwerken, Netzwerkarchitekturen oder Kommunikationsproblemen aktiv eingesetzt wird.

### 3.2 Das Konzept von Schichten- und Referenzmodellen

Die Verwendung von Modellen ist in der Informatik ein wichtiges Werkzeug allgemeine Sachverhalte zu beschreiben und liefert einen Ausgangspunkt für die konkrete Implementierung am Computer. Bei der Modellbildung sind vor allem zwei wichtige Aspekte zu berücksichtigen. Zum einen sollten komplexe Sachverhalte möglichst einfach dargestellt werden, zum anderen sollte ein gewisser Grad an Allgemeingültigkeit erzielt werden. Hierfür können vor allem die Ideen und Konzepte von Schichten- wie auch Referenzmodellen Anwendung finden.

Schichtenmodelle beschreiben eine Modellklasse, mit deren Hilfe komplizierte Zusammenhänge vereinfacht dargestellt werden können. Dies geschieht – wie der Name selbst schon beschreibt – durch die Trennung und Gliederung einzelner Aufgaben oder Probleme in verschiedene Schichten. Diese Aufteilung erlaubt dann in einem nächsten Schritt alle notwendigen Aufgaben der Schichten getrennt voneinander zu analysieren, zu beschreiben und letztlich zu implementieren. Die so gewonnenen Funktionalitäten aller einzelnen Schichten ergeben dann zusammengesetzt eine Lösung für die ursprüngliche Problemstellung. Wichtig ist in diesem Zusammenhang, dass die Aufteilung der Schichten hierarchisch erfolgt und eine feste Reihenfolge, in der die Schichten abgearbeitet werden müssen, besteht. Bei einer aufsteigenden Nummerierung der Schichten von 1 bis N tauscht jede Schicht  $n$  also nur mit ihren direkten Nachbarn ( $n-1$ ,  $n+1$ ) über fest definierte Schnittstellen Daten aus und stellt nur der nächst höheren Schicht ( $n+1$ ) Funktionen zur Verfügung. Die folgende Grafik soll genau dies visualisieren.

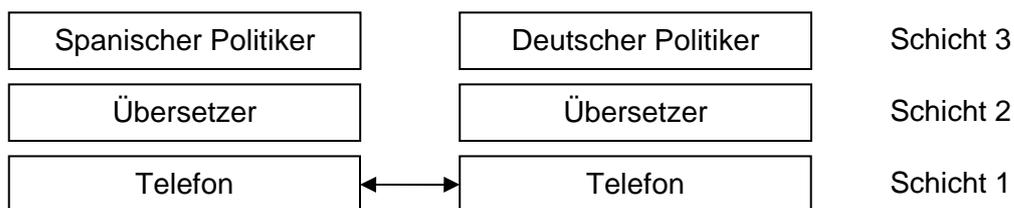


Abbildung 2: Beispielhafte Visualisierung eines Drei-Schichtenmodells.

In dem Beispiel benötigen ein spanischer und deutscher Politiker für ein Gespräch die Hilfe von Übersetzern, die in den Gesprächsablauf fest integriert sind. Die dargestellten Schichten müssen also immer in einer fest vorgegebenen Reihenfolge durchlaufen werden. Die eigentliche Gesprächslogik und die Kommunikation wird in Kapitel 3.3 genauer erläutert.

Wie eine effektive Trennung der Aufgaben erfolgen kann ist prinzipiell situationsabhängig, wichtig ist nur, dass die Gliederung in jeweils logischen Einheiten erfolgt. Aufgaben innerhalb einer Schicht müssen in sich abgeschlossen sein und die beschriebene Anforderung, nur mit den direkten Nachbarn zu interagieren, erfüllen. Die bekanntesten Schichtenmodelle sind das OSI- und TCP/IP-Referenzmodell. Diese Modelle bieten eine Beschreibung einer Netzwerkarchitektur bei gleichzeitiger Reduktion der Komplexität durch Verwendung von Schichtenmodellen und bilden insgesamt eine modellhafte Beschreibung vom Typ eines Referenzmodells.

Ein Referenzmodell beschreibt ein allgemeines Modell für eine Klasse von Sachverhalten, das vor allem zwei Ansprüchen genügen muss. Erstens muss es möglich sein auf Basis dieser generischen Modellbeschreibung ein spezielles Modell in Verbindung mit ganz bestimmten Sachverhalten zu planen und zweitens verschiedene Ausprägungen dann auch miteinander zu vergleichen. Ein Referenzmodell kann somit als ein Modell- bzw. Entwurfsmuster bezeichnet werden, welches die zu modellierenden Sachverhalte in einem idealtypischen Modell beschreibt.

### **3.3 Aufbau des OSI-Referenzmodells**

Wie bereits erwähnt beschreibt das OSI-Referenzmodell ein Schichtenmodell mit sieben Ebenen und ist in dem zugehörigen ISO Standard [IS7498-1] genau definiert. Umfangreiche Beschreibungen des Aufbaus sowie Analysen oder auch Vergleiche des OSI-Referenzmodells mit alternativen Modellen finden sich vielfach in der Fachliteratur zu Computernetzwerken, an dieser Stelle sei stellvertretend auf [DoCo04] und [PeDa04] verwiesen. Im Folgenden werden der Aufbau und die Funktionsweisen aller sieben Schichten grob beschrieben. Abbildung 3 visualisiert das OSI-Referenzmodell.

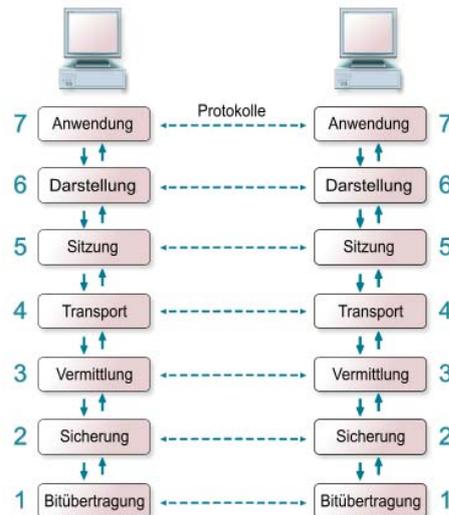


Abbildung 3: Darstellung aller Schichten des OSI-Referenzmodells.

Abbildung 3 zeigt die konkrete Aufteilung des OSI-Referenzmodells in die sieben Schichten mit den jeweiligen deutschen Bezeichnungen. Die Nummerierung aller Schichten erfolgt dabei vertikal von unten nach oben, basierend auf der Bitübertragungsschicht, gekennzeichnet als Ebene 1, bis hin zur Anwendungsschicht als höchste und 7. Ebene. Die vertikalen Pfeile zwischen den einzelnen Schichten symbolisieren die Interaktionsmöglichkeiten zwischen den einzelnen Schichten, die wie bereits oben gefordert hierarchisch erfolgt, jede Schicht also nur mit ihren direkten Nachbarn kommuniziert. Die technische Umsetzung bzw. die Implementierung einer Schicht bezeichnet man als Instanz. Dabei stellt jede Instanz Dienste zur Verfügung und nutzt dafür Dienste der direkt darunter liegenden Schicht. Insgesamt gilt deshalb für alle Schichten des Modells, dass der Abstraktionsgrad der einzelnen Ebenen von unten nach oben zunimmt. Dies bedeutet, dass die angebotenen Dienste einer Schicht für die jeweils nach oben folgende Schicht immer umfangreicher werden. Dies beruht auf der Tatsache, dass jede Schicht für eine bestimmte Aufgabe konzipiert ist und dafür die Dienste der direkt darunter liegenden Schicht, und somit indirekt auch die aller weiteren Vorgänger, in Anspruch nimmt. Die einzelnen Aufgaben der sieben Schichten sind im Folgenden knapp beschrieben, eine sehr gute und detaillierte Betrachtung der sieben Schichten findet sich in [TaN03].

- **Schicht 1 – Bitübertragungsschicht (engl. physical layer):** Die Bitübertragungsschicht ist die unterste Schicht. In dieser Schicht werden physikalische Verbindungen mit der Hilfe von mechanischen und elektrischen Elementen aktiviert bzw. deaktiviert sowie aufrechterhalten um Bits darüber zu übertragen.
- **Schicht 2 – Sicherungsschicht (engl. data link layer):** Mit Hilfe der Sicherungsschicht soll eine weitgehend fehlerfreie (also sichere) Übertragung gewähr-

leistet werden. Sie regelt außerdem den Zugriff auf das Übertragungsmedium selbst. Dazu dient das Aufteilen des Bitdatenstromes in Blöcke und das Hinzufügen von Folgenummern und Prüfsummen.

- **Schicht 3 – Vermittlungsschicht (engl. network layer):** Die Vermittlungsschicht ist für die Datenübertragung verantwortlich, die über das gesamte Kommunikationsnetz hinweg geht und schließt die Wegesuche (Routing) zwischen einzelnen Netzknoten mit ein.
- **Schicht 4 – Transportschicht (engl. transport layer):** Die Transportschicht ist die unterste Schicht, die eine vollständige Ende-zu-Ende Kommunikation zwischen Sender und Empfänger zur Verfügung stellt. Zu den Aufgaben dieser Schicht zählen die Segmentierung von Datenpaketen und die Stauvermeidung.
- **Schicht 5 – Sitzungsschicht (engl. session layer):** Die Sitzungsschicht stellt Dienste für einen organisierten und synchronisierten Datenaustausch zur Verfügung. Dies beinhaltet auch Fehlerprotokolle im Falle von allgemeinen Kommunikationsproblemen oder Verbindungsunterbrechungen.
- **Schicht 6 – Darstellungsschicht (engl. presentation layer):** Aufgabe der Darstellungsschicht ist es systemabhängige Daten in eine unabhängige Darstellungsform zu konvertieren und dadurch den syntaktisch korrekten Datenaustausch zwischen unterschiedlichen Systemen zu ermöglichen. Zusätzlich gehören die Datenkompression und die Verschlüsselung zu den Aufgaben dieser Schicht.
- **Schicht 7 – Anwendungsschicht (engl. application layer):** Die Anwendungsschicht ist die oberste der sieben hierarchischen Schichten. Sie stellt die Funktionalität den Anwendungen bereit, der eigentliche Anwendungsprozess liegt oberhalb dieser Schicht und ist nicht Bestandteil des OSI-Referenzmodells.

Ein Merksatz für alle OSI-Schichten in der Reihenfolge: Application, Presentation, Session, Transport, Network, Data link, Physical lautet zum Beispiel ***All People Seem To Need Data Processing.***

Für den eigentlichen Datenaustausch zwischen Sender und Empfänger ist es natürlich notwendig, dass beide Kommunikationspartner die Schichten (Protokolle) des OSI-Referenzmodells einhalten. Der tatsächliche Datenfluss innerhalb der Schichten erfolgt dann vertikal. Beim Sender durchlaufen die zu sendenden Daten die Schicht-

ten von oben nach unten also von Schicht 7 bis 1 und werden beim Empfänger von unten nach oben wieder schrittweise zusammengesetzt.

Die logische Kommunikation findet jedoch horizontal immer zwischen zwei identischen Schichten statt. Die ist in Abbildung 3 mit den horizontalen Pfeilen zwischen den einzelnen Schichten auf Sender- und Empfängerseite dargestellt und als Protokoll gekennzeichnet. Dabei beschreibt ein Protokoll prinzipiell einen Vorgang oder Prozess der nach bestimmten Regeln abläuft. Diese Regeln sind in dem OSI-Referenzmodell genau vorgegeben und bilden somit die Sprachgrundlage zwischen zwei Schichten der gleichen Ebene. Insbesondere ist es durch diese Kapselung der Kommunikation auch möglich einzelne Schichten gezielt auszutauschen, sofern dies beim Empfänger und beim Sender gleichzeitig erfolgt, der Rest des Modells bleibt davon dann unberührt.

In dem in Kapitel 3.2 gegebenen Kommunikationsbeispiel zwischen zwei Politikern, die keine gemeinsame Sprache sprechen, kann die eigentliche Verbindung über das Telefon beliebig ausgetauscht werden und beispielsweise durch ein Mobiltelefon ersetzt werden. Für den Inhalt des Gesprächs hat dies keine weitere Bedeutung. Auch die Übersetzer können für ihre Kommunikation untereinander eine beliebige Sprache wählen, dies ist ebenfalls für das eigentliche Gespräch unbedeutend.

Insgesamt beschreibt das OSI-Referenzmodell also einen allgemein gültigen Rahmen mit dessen Hilfe zwei beliebige Teilnehmer Daten austauschen können. Um einer Modellvermischung von OSI und TCP/IP entgegenzuwirken wollen wir im folgenden Kapitel beide Konzepte noch voneinander abgrenzen.

### **3.4 Abgrenzung von TCP/IP- und OSI -Referenzmodell**

In der Gegenüberstellung beider Modelle fällt sofort auf, dass sie sich vor allem in der Anzahl ihrer Schichten unterscheiden. Während das OSI-Modell in sieben Schichten gegliedert ist, unterscheidet man im TCP/IP-Referenzmodell lediglich vier Ebenen (Netzzugangsschicht, Internet-Schicht, Transportschicht, Anwendungsschicht). Vergleicht man nun weiter den Umfang und die Aufgaben der einzelnen Schichten aus beiden Modellen miteinander ergibt sich eine Zuordnung wie in der folgenden Grafik dargestellt.

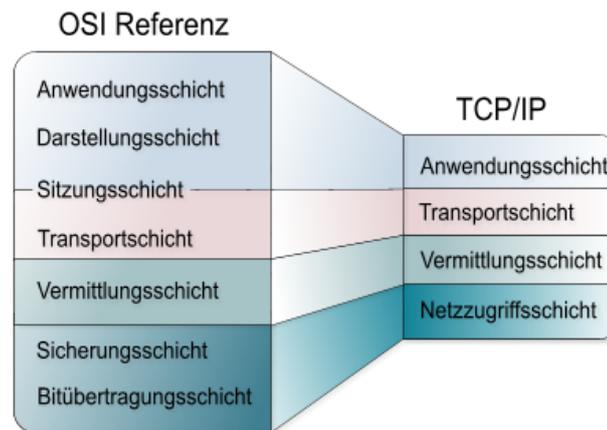


Abbildung 4: Vergleich der Schichtenaufteilung von OSI und TCP/IP.

Die Netzzugriffsschicht aus dem TCP/IP-Modell findet sein Gegenstück in der Bitübertragungs- und Sicherungsschicht des OSI-Modells. Eine Vermittlungsschicht existiert in beiden Modellen. Die Aufgabe der Transportschicht ordnet sich der Transport- und Sitzungsschicht zu und die Anwendungsschicht im TCP/IP-Modell umfasst die Aufgaben aus Anwendungs-, Darstellungsschicht und Sitzungsschicht im OSI-Modell. Es zeigt sich also, dass im TCP/IP-Referenzmodell keine so strenge Trennung der Aufgaben zwischen den Schichten besteht wie auf Seite des OSI-Referenzmodells, die vier Schichten überlappen sich also teilweise in ihren Aufgaben.

Ein weiterer wichtiger Punkt betrifft die Dokumentation beider Modelle. Das OSI-Referenzmodell als ISO Standard ist sehr gut und umfangreich dokumentiert. Es bildet einen offenen Standard und ein durchaus flexibles Modell mit der Möglichkeit es an bestimmte Gegebenheiten anpassen zu können. Die Dokumentation des TCP/IP-Modells ist weniger umfangreich und ist zudem auch erst nach der eigentlichen Implementierung erfolgt. Dafür sind die heutigen TCP/IP Implementierungen äußerst effizient und sehr stark verbreitet. Die gesamte Kommunikation im Internet wird üblicherweise über TCP/IP abgewickelt. Die effizienten und an die Gegebenheiten angepassten Implementierungen von TCP/IP machen es jedoch zu einem recht starren Modell.

Zusammengefasst kann man also festhalten, dass heute für die Kommunikation vor allem TCP/IP eingesetzt wird. Die sorgfältige Definition des OSI-Modells ist jedoch mit seinen theoretischen Überlegungen und Betrachtungen dem TCP/IP-Modell deutlich überlegen und besitzt folglich eine enorme Bedeutung als Referenzmodell.

## **4 Umsetzung im Informatikunterricht**

### ***4.1 Herangehensweise und Voraussetzungen***

Die hier vorgestellte Herangehensweise zur Umsetzung des Themenblocks ISO/OSI-Referenzmodell im Unterricht, folgt dem Vorschlag des hessischen Lehrplans, das Modell als zentrales Element zur Behandlung des Wahlthemas Rechnernetze zu verwenden. Mit Hilfe der theoretischen Grundlagen des Schichtenmodells können dann gezielt die weiteren Themenkomplexe des Wahlbereichs besprochen werden. Die Stoffvermittlung speziell für die Betrachtung des OSI-Modells sollte nach einer top-down Strategie erfolgen. Dabei muss zunächst eine Einordnung des Themenkomplexes in die Informatik selbst erfolgen und die Bedeutung und die Einsatzgebiete herausgearbeitet werden. Darauf aufbauend kann eine visualisierte Betrachtung des Modells (durch Medien unterstützte, wie zum Beispiel Grafiken oder Animationen) erfolgen, die dann in einer Beschreibung der einzelnen Modellschichten verfeinert werden kann.

An dieser Stelle sollte auch auf jeden Fall die offizielle Beschreibung der ISO-Norm [IS7498-1] in den Unterricht integriert werden. Eine vollständige Betrachtung der Dokumentation ist natürlich zu komplex und umfangreich, aber die Analyse und Diskussion von ausgewählten Inhalten soll in die Verwendung von solchen Dokumenten einführen und im Umgang mit solchen Materialien schulen. Speziell im Bereich der Informatik ist man häufig auf ähnlich umfangreiche Dokumentationen angewiesen. Diese beinhalten dann beispielsweise Detailinformationen für bestimmte Zusammenhänge und Schnittstellen oder etwa auch die Dokumentationen von Quellcode.

Neben der allgemeinen Herangehensweise ist auch die Definition von Voraussetzungen auf Schülerseite wichtig. Häufig sind bestimmte Themengebiete aufeinander aufbauend und setzen gezielt Kenntnisse voraus, damit überhaupt eine sinnvolle Betrachtung und Bearbeitung möglich ist. Das Wahlthema Rechnernetze sowie das ISO/OSI-Referenzmodell sind prinzipiell erst einmal nicht mit anderen Unterrichtseinheiten verknüpft und erfordern kein gezieltes Vorwissen. Die bereits erworbenen Grundkenntnisse aus dem vorhergehenden Informatikunterricht in der Mittel- und Oberstufe, die den Schülern als „Handwerkszeug“ bereits zur Verfügung stehen sind natürlich von Vorteil – speziell auch für die Umsetzung des unter 4.5 vorgestellten Programmierprojekts.

Diese Unabhängigkeit gegenüber anderen Themenkomplexen bietet somit eine einheitliche Startplattform für alle Schüler und erzeugt dadurch indirekt eine homogene Kursgruppe. Zusätzlich vorhandenes Fachwissen einzelner Schüler können außer-

dem gut in den Unterricht, beispielsweise durch Kurzreferate oder Ausarbeitungen, integriert werden. Im Gesamten ermöglicht das ISO/OSI-Referenzmodell einen logischen und an sich abgeschlossenen Einstieg in das Wahlthema.

## **4.2 Rahmenbedingungen und Lehr- und Lernziele**

Die allgemeinen Rahmenbedingungen für die Gestaltung eines Unterrichtskonzeptes sind elementar. Hierzu zählen unter anderen:

- Anzahl der Schüler im Kurs
- Grundlagen und Vorwissen der Schüler
- Die Kursform (Grund- und Leistungskurs)
- Organisatorische Elemente (Räume, Ausstattung und Hilfsmittel)
- Kursheterogenität
- Motivation der Schüler

Die hier beispielhaft genannten Punkte beeinflussen die Unterrichtsgestaltung natürlich direkt. Eine ganz detaillierte Planung eines Unterrichtsaufbaus und dessen Gestaltung sind vor diesem Hintergrund aber natürlich nicht möglich. Trotzdem wollen wir ein Grobkonzept vorschlagen mit Beispielen für den konkreten Unterrichtsinhalt. Dabei vernachlässigen wir gezielt nicht direkt beeinflussbare Faktoren wie Schüleranzahl und Motivationsaspekte. Was die organisatorischen Elemente betrifft, setzen wir das Vorhandensein eines Computerpools mit einer Entwicklungsumgebung einer höheren Programmiersprache (Delphi, Java) voraus sowie das Vorhandensein von Präsentationsgeräten.

Die Unterscheidung zwischen Grund- und Leistungskurs findet jedoch Berücksichtigung. Dabei sind die hier vorgeschlagenen Inhalte zunächst speziell für die Behandlung in einem Leistungskurs gedacht, können jedoch durch gezielte Reduktion des Umfangs oder Weglassen auf einen Grundkurs übertragen werden.

Die Rahmenbedingungen werden auch direkt von den Lehr- und Lernzielen beeinflusst, die wir für das Wahlthema auch auf Basis der noch folgenden Beschreibung der Unterrichtselemente nun angeben wollen. Zu den inhaltlichen Zielen zählen dem Namen des Wahlthemas entsprechend der Aufbau, die Verbreitung und die Bedeutung von heutigen sowie zukünftigen Rechnernetzen. Weiterhin sollen die Konzepte von Schichten- und Referenzmodellen, deren formale Bedeutung in der Informatik und die Funktionsweise sowie der Aufbau des ISO/OSI-Referenzmodells beschrieben und erklären werden können. Auch das effektive Arbeiten und Verwenden von

Fachliteratur in Form von Dokumentationen wird als Ziel gesetzt. Der Schüler soll eigenständig Arbeitsergebnisse hervorbringen, dokumentieren und präsentieren sowie abschließend in einem Programmierprojekt (siehe 4.6) das bereits erworbene Informatikfachwissen wiederholen und anwenden. Diese Punkte sollten dann auch Inhalt entsprechender Testate und Wissenskontrollen innerhalb des Halbjahrs sein. Diese können natürlich durch weitere behandelte Unterrichtseinheiten aus dem Komplex der Rechnernetze ergänzt werden, je nach gesetztem Schwerpunkt und verfügbarer Unterrichtszeit.

### **4.3 Unterrichtsformen, -Inhalt und -Umfang**

Die Wechselwirkungen zwischen Lehrer und Schüler sowie zwischen den Schülern untereinander prägen den Unterricht. Diese Beziehungen können anhand der zwei Aspekte Sozialform (Rahmen in dem der Unterricht abläuft) und Aktivitäten (Möglichkeiten der Schüler sich in den Unterricht einzubringen) genauer klassifiziert werden. Eine Übersicht über alle Sozialformen und Aktivitäten sowie deren Kombinationsmöglichkeiten findet sich in [ScSc04, S. 294]. Die wichtigsten und bekanntesten Unterrichtsformen sind der Frontalunterricht, Fragend-entwickelnder Unterricht, das freie Unterrichtsgespräch und individualisierte Instruktionen. Zur einfacheren Unterscheidung und besseren Zuordnung zu Gestaltung von Unterrichtseinheiten möchten wir die Unterrichtsformen in drei Gruppen mit ihrer jeweiligen Bedeutung gliedern.

- In der **Kursarbeit** erfolgt der Unterricht in der Gruppe, in der alle Schüler unabhängig von der tatsächlichen Unterrichtsform sich mit denselben Inhalten beschäftigen.
- In der **Projektarbeit** arbeitet ein Kurs gemeinschaftlich an einem Gesamtprojekt, dies kann aufgeteilt in kleinen Gruppen oder in einer großen Gruppe erfolgen.
- In der **Einzelarbeit** erhält jeder Schüler individuelle Instruktionen und Aufgaben, dies umfasst neben der Unterrichtszeit auch Heimaufgaben wie Referate oder Vorträge.

Die verschiedenen Unterrichtsformen müssen jetzt noch mit Inhalten gefüllt werden. Für die Behandlung des ISO/OSI-Referenzmodells bietet sich auch hier eine Gliederung in drei Aufgabenbereiche, die sich an den verschiedenen Unterrichtsformen orientieren, an. Der erste Bereich umfasst im Rahmen einer Kursarbeit zunächst eine Einführung in das Gesamtthema und den Unterrichtsstoff. Nach der Grundlagenvermittlung folgt eine Projektphase mit intensiver Projektarbeit. Nach Abschluss und auch bereits parallel zu der Projektphase können weitere Unterrichtsinhalte in den Unterricht integriert werden. Die Inhalte sollten dabei aus den vorgegebenen Einheiten

ten zum Thema Rechnernetze gewählt werden und in Form von Einzelarbeiten an die Schüler verteilt werden. Die Kombination dieser verschiedenen Arbeitsmethoden in Verbindung mit dem *Lernen im Kontext der Anwendung* werden auch die im Lehrplan formulierten Anforderungen an das Lernen selbst erfüllt – hier wird das Lernen als „aktiv-konstruktiver, selbstgesteuerter, situativer und sozialer Prozess“ [vgl. Leh-Hes] formuliert.

#### **4.4 Aufteilung der Unterrichtszeit und Gestaltung der Einheiten**

Die Aufteilung des Unterrichts sollte in Anlehnung an die in 4.3 vorgestellten Arbeits- und Unterrichtsformen erfolgen. Ausgangspunkt für den Unterricht bildet die Einführung in das Thema. Als Lehrer- oder Schülervortrag ausgestaltet, sollte das ISO/OSI-Referenzmodell Schicht für Schicht mit Beispielen behandelt werden. Bei der Behandlung bietet sich auch ein Rollenspiel an, in dem die einzelnen Schichten und deren Kommunikation durchgespielt werden. Der Einsatz von Visualisierungssoftware ist in der Einführungsphase ebenfalls sinnvoll und hilfreich.

Nach der Einführung können die weiteren Inhalte des Themas Rechnernetze besprochen werden, die im direkten Bezug mit dem ISO/OSI-Referenzmodell stehen. Dazu gehören besonders die Hardwarekomponenten eines Netzwerkes, verwendete Protokolle und das Routing. In der anschließenden Projektphase, die den Hauptteil der Unterrichtszeit einnehmen sollte, werden die Inhalte in einer Einzel- oder Teamarbeit vertieft. Der Ausgestaltung des Projektes ist das nächste Kapitel gewidmet.

#### **4.5 Programmierprojekt als Vertiefungsmöglichkeit**

Im Kontext der Behandlung des Wahlthemas Rechnernetze in Verbindung mit der Betrachtung des ISO/OSI-Referenzmodells bietet sich die Durchführung eines Rahmenprojektes zur Stoffvermittlung an. Die Schüler können dabei im Kontext der Anwendung lernen, Grundlagen wiederholen und Gelerntes umsetzen.

Ein Softwareentwicklungsprojekt in Kleingruppen erfordert ein Zusammenspiel vieler Kenntnisse. Besonders hervorzuheben sind dabei die Modellierung der Anwendung und der Einsatz entsprechender Modellierungssprachen wie UML (Unified Modeling Language) und die objektorientierte Programmierung in einer höheren Programmiersprache (Delphi, Java, C#). Weitere Anforderungen betreffen die sozialen Kompetenzen der Schüler, wie Arbeiten, Kommunizieren und Aufgabenteilung in einem

Team. Zu jedem Projekt gehört natürlich auch eine Präsentation der Ergebnisse, die kommunikative Fähigkeiten fördert.

Als Projektideen gibt es beispielsweise die Möglichkeit, gezielt einzelne Schichten des ISO/OSI-Referenzmodells zu implementieren oder auch die Entwicklung einer Netzwerkanwendung (Client-Client oder Client-Server basiert). Je nach verfügbarer Zeit (GK/LK) und vorhandenen Kenntnissen kann ein passendes Projekt gewählt werden.

Da die Planung eines solchen Projektes recht umfangreich ist und ein eigenständiges Thema darstellt, werden im Folgenden nur Vorschläge, Ideen und Quelltextfragmente für Projekte gegeben, eine vollständige Realisierung bleibt der Lehrkraft überlassen.

### *Netzwerkkomponenten als Ausgangspunkt der Entwicklung*

Für die konkrete Entwicklung einer netzwerkfähigen Anwendung wird eine Programmiersprache benötigt, die entweder bereits Netzwerkfunktionalitäten bereitstellt, oder um diese erweiterbar ist.

Die Programmiersprache Java bietet beispielsweise bereits die gewünschte Funktionalität. Mit Hilfe von Socket-Objekten kann eine Client-Server-Kommunikation über ein Netzwerk sehr schnell umgesetzt werden. Socket-Objekte basieren auf TCP/IP und benötigen zur Kommunikation lediglich eine IP-Adresse und eine Port-Nummer.

Ein konkreter Vorschlag zur Umsetzung findet sich im Anhang dieser Arbeit. Er besteht aus den drei Klassen *JavaClient*, *JavaServer* und *Connection*. Der *JavaServer* ist bereits fähig, mehrere Verbindungen gleichzeitig einzugehen. Einmal gestartet, wartet er auf eingehende Client-Verbindungen. Sobald eine Verbindung angefragt wird, wird ein Objekt der Klasse *Connection* erzeugt. In dieser Klasse ist der serverseitige Programmcode festgehalten, der clientseitige Programmcode befindet sich in der Klasse *JavaClient*. Das Programmbeispiel ermöglicht die Abfrage von Serverdaten. Auf Grundlage dieses einfachen und schnell zu implementierenden Beispiels sind viele verschiedene Netzwerkanwendungen umsetzbar, im Folgenden werden mehrere Anregungen gegeben.

### *Implementierung einzelner Schichten / Schichtaspekte des OSI-Referenzmodells*

Da der hier beschriebene Server bereits auf TCP/IP und damit auf Schicht 4 aufsetzt (also die Anwendung selbst darstellt), kann die Implementierung der ersten vier

Schichten nur über eine Emulation erfolgen. Denkbar wäre es, dass mit Hilfe der beschriebenen Funktionalität von Java ein Netzwerkkartenemulator dem Schüler als Ausgangspunkt für seine Implementierung vorgegeben wird. Je nach Schicht, die zu implementieren ist, muss die Vorgabe passende Funktionalitäten bereitstellen.

- **Bitübertragungsschicht:** Die Bitübertragungsschicht sollte genau zwei Funktionen bereitstellen, nämlich *void sendeBit(boolean bit)* und *boolean empfangenBit()*.
- **Sicherungsschicht:** Hier geht es um die Aufteilung eines Datenstroms in Blöcke, eine anschließende Nummerierung und das Hinzufügen von Prüfsummen. Dabei muss sowohl eine geeignete Blockgröße gefunden werden (zum Beispiel ein Byte), eine Nummerierungsmethode überlegt und ein Prüfsummenalgorithmus ausgesucht werden. Als Prüfsummenalgorithmus bieten sich mehrere Verfahren an. Denkbar sind hier die Verwendung von Paritätsbits, Techniken, die auf modulo Operationen basieren, oder kryptographische Einwegfunktionen (Hashmethoden).

Die ganze Information muss dann in einem geeigneten Format kodiert und gesendet werden. Wurde ein Fehler entdeckt (falsche Prüfsumme, fehlendes Paket), muss das entsprechende Paket neu angefragt werden. Der Vorgang ist in der folgenden Abbildung visualisiert. Ein zu sendender Block wird um eine Nummerierung und Prüfsumme ergänzt und anschließend mit den Funktionen der darunter liegenden Schicht Bit für Bit gesendet.

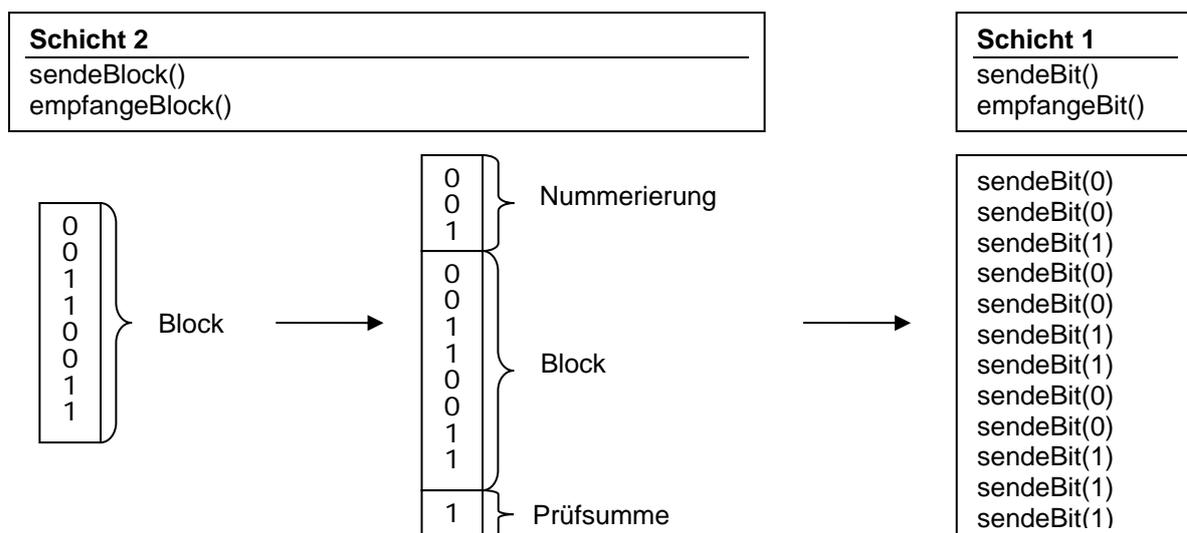


Abbildung 5: Sendevorgang auf Schicht 2 mit Hilfe von Schicht 1.

- **Vermittlungsschicht:** Hier muss ein geeignetes Zustellungsverfahren für einzelne Pakete implementiert werden. Zu den Paketen sollten dabei zunächst eindeutige Absender- und Empfängeradressen hinzugefügt werden. Je nach Art der zu

implementierenden Netzwerkkomponente muss unterschiedlich mit eingehenden und ausgehenden Paketen verfahren werden. Bei einem Endgerät, wie einer Netzwerkkarte, müssen eingehende Pakete auf Relevanz überprüft werden. Ist das Paket für einen anderen Empfänger adressiert, kann es entweder ignoriert werden oder es muss weitergesendet werden (wie in einem Ring). Möchte man einen Switch implementieren, müssen Tabellen geführt werden, an welchem Port welcher Client angeschlossen ist. Sind dem Switch diese Information noch nicht bekannt, arbeitet er wie ein Hub, sendet also alle eintreffenden Pakete an alle angeschlossenen Clients. Eine weitere Geräteklasse, der Router, kann ebenfalls implementiert werden. Er muss in der Lage sein, Absender- und Empfängerinformationen entsprechend der Netzaufteilung manipulieren zu können.

- **Transportschicht:** Für die Transportschicht kann die Datenübertragung zwischen den Benutzern implementiert werden. Ähnlich der Sicherungsschicht erfolgen eine Segmentierung, eine Prüfsummenberechnung, sowie eine Fehlerkontrolle, jedoch diesmal bezüglich der Ende-zu-Ende Kommunikation (also bezüglich der Kommunikation zwischen den obersten Schichten). Eine Stauvermeidung lässt sich beispielsweise über Empfangsquittungen realisieren, das heißt weitere Daten werden erst gesendet, wenn Empfangsbestätigungen der vorher gesendeten Daten vorliegen.
- **Sitzungsschicht:** Zu implementieren ist der Aufbau und der Abbau einer Sitzung zwischen zwei Kommunikationspartnern. Innerhalb einer Sitzung müssen Daten ausgetauscht werden können.
- **Darstellungsschicht:** Zu den Aufgaben der Darstellungsschicht gehört hauptsächlich die Codierung. Hier kann der Schüler Überlegungen treffen, wie er zum Beispiel ein Bild, einen Text oder Zahlen überträgt. Als zusätzliche Funktionen kann eine Kompression oder eine Verschlüsselung implementiert werden.
- **Anwendungsschicht:** Hier müssen Funktionen für die Anwendungen implementiert werden. Zum Ergebnis gehören Möglichkeiten zum Auf- und Abbau von Verbindungen sowie das Senden und Empfangen in vorgegeben Formaten.

### *Implementierung eines kompletten Netzwerk-Stapels*

Neben der Implementierung einzelner Aspekte bietet es sich auch an, einen kompletten Netzwerkstapel zu implementieren. Einzelne Gruppen können dabei einzelne Schichten behandeln. Im Vorfeld müssen genau die Schnittstellen zwischen den ein-

zelen Schichten und die Anforderungen an die Schichten definiert werden, damit zum Schluss alle Schichten erfolgreich zusammengeführt werden können. Wegen des hohen Zeitaufwands und der Komplexität sollte die Implementierung eines kompletten Netzwerkstapels dem Leistungskurs vorbehalten sein.

### *Entwicklung einer netzwerkfähigen Software*

Eine weitere Projektidee liegt in der Programmierung einer netzwerkfähigen Software als Vertiefung der Schicht 5 bis 7. Mit der bereits beschriebenen Java-Funktionalität sollte eine Umsetzung sehr leicht möglich sein. Mögliche Anwendungen sind beispielsweise Multinutzer-Spiele (Beispiel siehe Abbildung 6), Kommunikationsanwendungen (einfaches Chatprogramm) und Datenbankanwendungen (Datenabfragen, Expertensysteme).

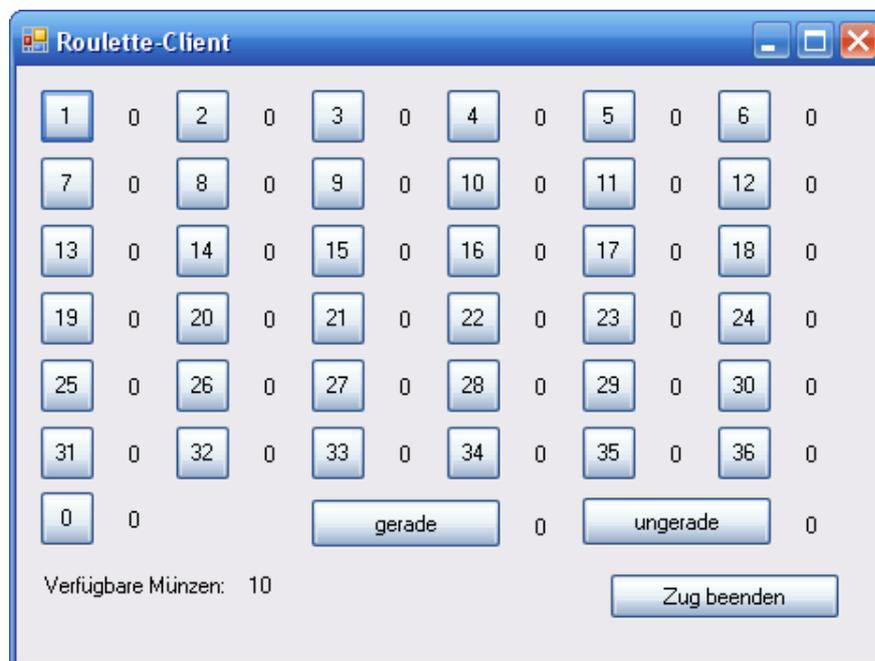


Abbildung 6: Programmoberfläche eines „Netzwerk-Roulette“.

Die Durchführung eines Programmierprojektes und der damit verbundenen intensiven Betrachtung des ISO/OSI-Referenzmodells bietet insgesamt also einen sehr guten Rahmen zur Stoffvermittlung.

## **4.6 Wissenskontrolle**

Eine Wissenskontrolle kann entweder über klassische Aufgabenstellungen, Online-Wissenstests oder Multiple Choice Tests erfolgen. Als Aufgabentypen sind dabei vor

allem Fragen zu den Modellschichten, bei der einzelne Protokolle den entsprechenden Schichten zugeordnet werden sollen, die Erklärung einzelner Schichten mit dem besonderen Augenmerk auf die unterschiedlichen Gerätetypen (Netzwerkkarte, Hub, Switch, Router, Firewall, etc.) denkbar sowie die Planung eines (sicheren) Netzwerkes in Hardware. Weitere Beispiele für Übungen und Kontrollfragen finden sich in [HoKe98, Kap.7].

#### **4.7 Bewertung des Konzeptes**

Das vorgestellte Konzept unterteilt sich in theoretische und praktische Phasen. Es ist anwendungsbezogen und durch das Projekt auch produktorientiert. Dabei werden die Schüler auf Probleme aus der Praxis stoßen, die bei einer rein theoretischen Betrachtung der Thematik unentdeckt bleiben würden. Somit werden die Problemlösungskompetenzen des Schülers gefordert und gestärkt. Eigene Kenntnisse des Schülers können besonders im Projekt eingebracht werden. Bei der Ausführung des Projektes wurde auf die Bildung von Teams als kooperative und kommunikative Arbeitsform besonderer Wert gelegt. Die Analyse und kritische Themenbetrachtung beinhaltet auch gesellschaftliche und soziale Aspekte. Der beschriebene Ansatz ermöglicht ein Spiralcurriculum und erfüllt die im Lehrplan vorgegebenen Leitlinien.

### **5 Zusammenfassung und Fazit**

Die Behandlung des ISO/OSI-Referenzmodells im Rahmen des Wahlpflichtunterrichts umfasst ein sehr breites und interessantes Themengebiet mit starkem Bezug zu heutigen Netzwerken und Entwicklungen in diesem Bereich. Insbesondere weil das ISO/OSI-Referenzmodell nicht ein eigenständiges Wahlthema darstellt, sondern in den Bereich des Wahlthemas Rechnernetze fällt und hier weitere Inhalte wie beispielsweise Sicherheitsaspekte betrachtet werden können. Der Rahmen des Wahlthemas sollte aber durch das ISO/OSI-Referenzmodell klar abgesteckt sein und somit die Grundlage für Lehr- und Lerninhalte bilden.

Trotzdem ist das ISO/OSI-Referenzmodell mit einem Makel behaftet. Seine geringe Verbreitung und die Dominanz des TCP/IP Modells in der Praxis sollte nicht unerwähnt bleiben. Die Betrachtung eines vereinfachten OSI-Schichtenmodells, in dem nicht alle Schichten gezielt betrachtet werden oder gar die Kombination von TCP/IP und ISO/OSI zu einem Hybridmodell, könnte diesem Punkt aber Rechnung tragen.

Im Vergleich zu anderen Wahlthemen ist eine Bewertung und Einordnung im Hinblick auf seine Bedeutung für den Schüler sehr schwierig. Außerdem ist es abhängig vom Interesse der jeweiligen Schüler, die im Sinne einer modernen Pädagogik in die Wahl des Vertiefungsthemas eingebunden werden sollten, welches Wahlthema im Unterricht letztendlich ausgewählt wird. Wir halten die Behandlung des Themenblocks Rechnernetzen und damit verbunden des ISO/OSI-Referenzmodells als Wahlthema nicht für zwingend erforderlich. Dann sollte auf das Thema aber mindestens im Rahmen eines Exkurses eingegangen werden. Wann und in welcher Form dieses erfolgen soll ist recht beliebig. Es bieten sich hier umfangreiche Möglichkeiten an, zum Beispiel Referate oder Facharbeiten durch die Schüler anfertigen oder die Inhalte gezielt bei der Behandlung von anderen Pflichtelementen (zum Beispiel Internet und TCP/IP in 11.1 oder Informatikmodelle in 12.2) einfließen zu lassen.

### III. Literaturverzeichnis

- [BaRü96] Baumann, Rüdiger, Didaktik in der Informatik, 2. Auflage 1996.
- [DoCo04] Douglas, Comer E., Computernetzwerke und Internets., Gebundene Ausgabe 2004.
- [EbFr96] Eberle, Franz, Didaktik der Informatik bzw. einer informations- und kommunikationstechnologischen Bildung auf der Sekundarstufe II, 1. Auflage 1996.
- [HeHa97] Heymann ,Hans W., Allgemeinbildung und Fachunterricht, 1. Auflage 1997.
- [HoKe98] Horn, Christian und Kerner, Immo O., Lehr- und Übungsbuch Informatik, Band 4: Technische Informatik und Systemgestaltung 1998.
- [HuPe01] Hubwieser, Peter, Didaktik der Informatik – Grundlagen, Konzepte, Beispiele, 1. korrigierter Nachdruck 2001.
- [HuLu05] Humbert, Ludger, Didaktik der Informatik mit praxiserprobtem Unterrichtsmaterial, 1. Auflage 2005.
- [IS7498-1] International Standard, ISO/IEC 7498-1, Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model, Corrected and reprinted 1996-06-15.
- [LehHes] Lehrplan des Landes Hessen, G9 – Informatik, derzeit gültige Fassung.
- [PeDa04] Peterson, Larry L. und Davie, Bruce S., Computernetze. Eine systemorientierte Einführung, Gebundene Ausgabe 2004.
- [ScSc04] Schubert, Sigrid und Schwill Andreas, Didaktik der Informatik, 1. Auflage 2004.
- [TaAn03] Tanenbaum, Andrew S., Computernetzwerke, 4. überarbeitete Auflage 2003.

## IV. Anlage

Quellcode für eine einfache Client-Server Anwendung in Java.

```
// Datei: Connection.java

import java.io.*;
import java.net.*;

/**
 * @author Jonas Steitz, 2007
 *
 * Beispielanwendung fuer eine Client-Server Anwnendung, bestehend aus:
 * - Connection: Verbindungsklasse von Client zu Server
 * - JavaClient: Clientanwendung zum Senden von Serveranfragen
 * - JavaServer: Serveranwendung zum Beantworten von Clientanfragen
 */
class Connection extends Thread {

    /**
     * Socket für Kommunikation.
     */
    private Socket socket;

    /**
     * Eingaben vom Client.
     */
    private BufferedReader in;

    /**
     * Ausgaben vom Server.
     */
    private PrintWriter out;

    /**
     * @param s - Socket für Kommunikation.
     * @throws IOException - Ein- und Ausgabefehler (keine
     * Fehlerbehandlung).
     */
    public Connection(Socket s) throws IOException {
        socket = s;
        in = new BufferedReader(new InputStreamReader(
            socket.getInputStream()));
        out = new PrintWriter(new BufferedWriter(
            new OutputStreamWriter(socket.getOutputStream())), true);
        start(); // Startet run()
    }

    /**
     * @see java.lang.Runnable#run()
     */
    public void run() {
        // Hier wird der Programmablauf auf der Server-Seite gesteuert
        while (true) {
            String dm = receiveFromClient();
            sendToClient("SERVER: Anfrage erhalten: " + dm);
            if (dm.equals("EUR")) {
                sendToClient("SERVER: 1,95583");
            }
        }
    }
}
```

```

        else if (dm.equals("LIR")) {
            sendToClient("SERVER: 1.936,27");
        }
        else if (dm.equals("EXIT") ||
            dm.equals("#Fehler beim Empfang")) {
            closeConnection();
            break;
        }
        else {
            sendToClient("SERVER: Waehrung nicht bekannt");
        }
    }
}

/**
 * Clientanfragen empfangen.
 * @return String - Empfangene Nachricht oder Fehlermeldung.
 */
private String receiveFromClient() {
    String buffer = null;
    try {
        buffer = in.readLine();
        if (buffer == null) {
            return "#Fehler beim Empfang";
        }
    }
    catch (IOException e) {
        return "#Fehler beim Empfang";
    }

    return buffer;
}

/**
 * Clientanfrage beantworten.
 * @param data - Zu sendende Daten.
 */
private void sendToClient(String data) {
    out.println(data);
}

/**
 * Client-Server Verbindung beenden.
 */
private void closeConnection() {
    try {
        socket.close();
    }
    catch (IOException e) {
        // Fehlerbehandlung...
    }
}
}
}

```

```

// Datei: JavaServer.java

import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;

/**
 * @author Jonas Steitz, 2007 (JavaServer)
 *
 * Ein JavaServer mit der Moeglichkeit, mehrere
 * Verbindungen zu verwalten.
 */
public class JavaServer {

    /**
     * Port für die Kommunikation.
     */
    public static final int PORT = 2477;

    /**
     * Haupteinstiegspunkt fuer die Anwendung.
     * @param args - Startargumente (keine notwendig).
     * @throws IOException - Alle Ein- und Ausgabefehler.
     */
    public static void main(String[] args) throws IOException {

        // Server-Socket erstellen
        ServerSocket ssocket = new ServerSocket(PORT);
        System.out.println("SERVER: Gestartet: " + ssocket);

        // Auf Client-Anfragen warten und Verbindungen aufbauen
        try {
            while (true) {
                Socket socket = ssocket.accept();
                try {
                    new Connection(socket);
                }
                catch (IOException e) {
                    socket.close();
                }
            }
        }
        finally {
            // Server Socket schliessen
            ssocket.close();
        }
    }
}

```

```

// Datei: JavaClient.java

import java.io.*;
import java.net.*;

/**
 * @author Jonas Steitz, 2007 (JavaClient)
 *
 */
public class JavaClient {

    /**
     * Port für die Kommunikation.
     */
    public static final int PORT = 2477;

    /**
     * IP-Adresse des Servers.
     */
    public static final String IPNUMBER = "127.0.0.1";

    /**
     * "Reader"-Objekt fuer Eingabedaten.
     */
    private static BufferedReader in;

    /**
     * "Writer"-Objekt fuer Ausgabedaten.
     */
    private static PrintWriter out;

    /**
     * Haupteinstiegspunkt fuer die Anwendung.
     * @param args - Startargumente (keine notwendig).
     * @throws IOException - Moegliche Ein- und Ausgabefehler.
     */
    public static void main(String[] args) throws IOException {

        // Verbindung aufbauen
        InetAddress addr = InetAddress.getByName(IPNUMBER);
        System.out.println("CLIENT: Adresse:" + addr);
        Socket socket = new Socket(addr, PORT);
        try {
            // Reader zum Empfangen vom Server
            in = new BufferedReader(new InputStreamReader(socket
                .getInputStream()));

            // Writer fuer das Senden zum Server
            out = new PrintWriter(new BufferedWriter(
                new OutputStreamWriter(socket.
                    getOutputStream())), true);

            // Programmablauf Client starten
            run();
        }
        finally {
            socket.close();
        }
    }

    /**
     * Programmablauf, clientseitig.
     */
}

```

```

public static void run() {
    // Reader zum Lesen von Benutzereingaben auf der Tastatur
    BufferedReader line = new BufferedReader(
        new InputStreamReader(System.in));

    try {
        while (true) {
            System.out.print("CLIENT: Geben Sie eine" +
                " Waehrung ein oder EXIT zum beenden: ");
            String s = line.readLine();
            out.println(s);

            if (s.equals("EXIT")) {
                System.out.println("CLIENT: Verbindung" +
                    " beendet!");
                System.exit(0);
            }

            String str = in.readLine();
            System.out.println(str);
            str = in.readLine();
            System.out.println(str);
        }
    } catch (Exception e) {
        // Unbekannter Programmfehler
        System.out.println(e.getMessage());
        System.exit(0);
    }
}
}

```