



**Chefredakteur (V.i.S.d.P.):**  
StD Dr. Ludger Humbert  
**Redaktion:** StD Christian F. Görlich  
& Fachseminare Informatik Hamm und Arnsberg  
**Layout:** Ludger Humbert (Vorlage von Torsten Bronger)  
**Copyright:** Für namentlich gekennzeichnete Artikel übernimmt die Autorin die Verantwortung.



**SOME RIGHTS RESERVED**

Der Inhalt unterliegt der [Creative Commons License](http://creativecommons.org/licenses/by-nc-sa/2.0/de-Lizenz)  
by-nc-sa/2.0/de-Lizenz ...  
If Fase ... auch im Netz ([humbert.in.hagen.de/iffase](http://humbert.in.hagen.de/iffase))  
ISSN 1861-0498

# If Fase

IF FASE: INFORMATIKFACHSEMINARE HAMM, ARNSBERG

Bildung  
Fachseminare  
Veranstaltungen  
Informatiksysteme  
Ausbildungsschulen

If Fase

Nr. 10

HERAUSGEGEBEN VON DER IF FASE-REDAKTION

1. Juni 2006

## Termine



Freitag, 23. Juni 2006

### Schuljahr 2005/2006

letzter Unterrichtstag NW  
(Nordrhein-Westfalen)

Dienstag, 27. Juni 2006

### BdU Vorbereitung 2006/2007

ab 10:00  
Hamm – Studienseminar

Auftakttermin – weitere Termine:

**Donnerstag, 29. Juni** **Dienstag, 1. August**

**Mittwoch, 2. August 2006**

semsek2.ham.nw.schule.de

Mittwoch, 9. August 2006

### Schuljahr 2006/2007

Unterrichtsbeginn NW

## KurzNotiert



### Erklärung der Fachleitungen Informatik NW

Die aktuelle Situation des Schulfachs Informatik wurde von den Fachleitungen Informatik zum Anlaß für eine gemeinsame Erklärung. Diese Erklärung wurde von einem Kollegen beim **Bildungspolitischen Netzwerk** in Düsseldorf dem Staatssekretär **Günter Winands**, der dort das neue Schulgesetz vorstellte, zur Kenntnis gegeben. Die Erklärung der Fachleitungen ist verfügbar unter: [haspe.homeip.net:8080/blog/Texte/PDF/2006-03-10\\_Informatik\\_SekI.pdf](http://haspe.homeip.net:8080/blog/Texte/PDF/2006-03-10_Informatik_SekI.pdf) Es ist skandalös, dass es im Ministerium für das Zukunftsfach Informatik keine/n qualifizierte/n Ansprechpartner/in gibt. Soviel zum Informatikjahr aus Sicht der Schule in Nordrhein-Westfalen (NW).

### Lernen mit Medien – empirische Befunde und didaktische Konsequenzen

Medien – auch ein Thema für die Didaktik der Informatik. Prof. Dr. Bardo Herzig von der Universität Paderborn informierte in der Aula des Hammonense für die Referendarinnen und Referendare über Forschungsergebnisse. Er teilte aus der Sicht einer pragmatisch orientierten Forschung mit, welche Notwendigkeiten sich für die weitere Arbeit ergeben: [www.ham.nw.schule.de/pub/bscw.cgi/d317258/Hamm\\_29\\_05\\_06\\_Herzig.pdf](http://www.ham.nw.schule.de/pub/bscw.cgi/d317258/Hamm_29_05_06_Herzig.pdf)

### MWS – 2006

Am Freitag, 5. Mai 2006 fand in Münster der Münsteraner Workshop zur Schul-informatik (MWS) [ddi.uni-muenster.de/ab/se/mws](http://ddi.uni-muenster.de/ab/se/mws) statt. Neben dem Beitrag *Informatische Bildung – Fehlvorstellungen und Standards* ([ddi.uni-muenster.de/0x8d593037\\_0x0006190d](http://ddi.uni-muenster.de/0x8d593037_0x0006190d)) (Ludger Humbert) wurde auch der Beitrag *Kryptographie im Informatikunterricht der Sekundarstufe I* ([pasternak.in.hagen.de/iff/informat.htm](http://pasternak.in.hagen.de/iff/informat.htm)) **Arno Pasternak** in einer öffentlich zugänglichen Fassung vorgelegt. Dabei ist anzumerken, dass der Reader *nicht* in digitaler Form veröffentlicht werden soll – Ergebnis einer Rückfrage bei Prof. Dr. Marco Thomas – Didaktik der Informatik an der Universität Münster.



## L<sup>A</sup>T<sub>E</sub>X – Teil 10: Abbildungen

In einer Reihe von Artikeln in der If Fase werden nützliche Elemente von L<sup>A</sup>T<sub>E</sub>X vorgestellt, die erprobt sind und bei der Arbeit der Informatiklehrerin eingesetzt werden.

(von Dr. Ludger Humbert) In den bisher vorgelegten zehn Teilen der Artikelserie – Ausgaben 0 ... 9: [humbert.in.hagen.de/iffase/Archiv](http://humbert.in.hagen.de/iffase/Archiv) – finden Sie Hinweise und Anmerkungen zu den Themen: Installation, grundlegende Arbeitsweisen, Quellen zu Dokumentationen, Arbeit mit KOMA-Script, PSTricks, Erstellung von Arbeitsblättern, Struktogrammen, Automaten- grafen, Elemente von UML, Barcodes, Formularerstellung, Zitieren. Beginnend mit der letzten Ausgabe (Zitieren – normgerecht) wird das Thema von Fragen getrieben, die von den Referendarinnen gestellt werden. Damit soll der konkrete Ausbildungsnutzen nachhaltig verbessert werden.

### Abbildung – ein Bild sagt mehr ...

In dieser Ausgabe werden einige Hinweise zu Abbildungen für den Textsatz mit L<sup>A</sup>T<sub>E</sub>X gegeben. Abbildungen liegen entweder in einer verlustfrei skalierbaren vektororientierten oder in einer verlustbehafteten pixelorientierten Form vor. Zunächst ist anzumerken, dass vektororientierten Formaten immer der Vorzug gegenüber pixelorientierten Formaten zu geben ist, wenn es um Qualität im Ergebnis geht. Das bedeutet für die Erstellung von Grafiken: Versuchen Sie möglichst vektororientierte Grafiken mit Ihren Hilfsmitteln zu erstellen. Dabei muss berücksichtigt werden, dass in Vektorgrafikdateien Pixelgrafiken eingebaut sein können ... Diese Eigenschaft kann zu sehr schlechten Ergebnissen im Satz führen, obwohl Vektorgrafiken erstellt wurden. Auf jeden Fall ist dafür Sorge zu tragen, dass verwendete Zeichensätze (ggf. als Teilmenge – Subset) Bestandteil der Grafik sind. Damit wird die Grafik eigenständig und portierbar.

Mit folgenden Kommando wird die PDF-Datei `in.pdf` nachträglich um die fehlenden Zeichensätze ergänzt – Ergebnis in `out.pdf`

```
ps2pdf14 -dPDFSETTINGS=/prepress in.pdf out.pdf
```

### Einfügen von Grafiken – in L<sup>A</sup>T<sub>E</sub>X

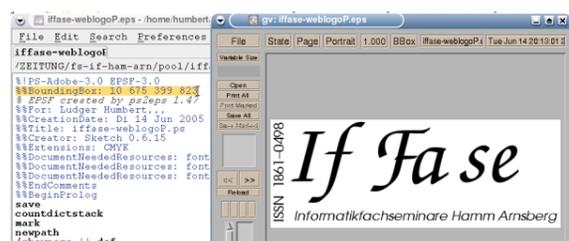
Da L<sup>A</sup>T<sub>E</sub>X die Art der eingefügten Grafik automatisch feststellt, ist unbedingt zu empfehlen, die Datei ohne Dateiendung (wie `.pdf`, `.eps` oder `.png`) anzugeben.

```
\begin{figure}
\begin{center}
\includegraphics[ABB/didr_emb]
\end{center}
\begin{center}nach \cite[S.~132]{Meyer1988a}\end{center}
\caption{
\label{cap:Didaktisches-Dreieck}
Didaktisches Dreieck\index{didaktisches!Dreieck}
}
\end{figure}
```

Nach der Vorabveröffentlichung dieses Artikels erhielt ich folgende Rückmeldung: »Vielleicht wäre es noch interessant, auf die Argumente von `\begin{figure}` einzugehen. Ich habe ganz gute Erfahrungen mit folgender Version gemacht, um (gerahmte) Bilder auf ganzer Seitenbreite möglichst groß einzufügen. Ich hoffe, das Benutzen von `!` ist nicht zu dirty.«

```
\begin{figure}[ht!]
\begin{center}
\fbbox{\includegraphics[height=20cm]{Beispiel}}
\end{center}
\end{figure}
```

### Begrenzungsbox – Bounding Box



### Begrenzungsbox bearbeiten

Zur Einbindung von Grafiken in gesetzte Dokumente ist es sinnvoll, Grafiken zurechtzuschneiden, so dass eine Skalierung nur auf den anzuzeigenden Bereich bezogen wird. Postscript-Dateien enthalten üblicherweise keine Begrenzungsbox (engl. Bounding-Box) für den sichtbaren Bereich. Dazu steht das Format `eps` (encapsulated Postscript) zur Verfügung. Mit einem

Werkzeug, wie `ps2epsi` oder `ps2eps` erledigt ein Automat die Umwandlung. Die aus reinen ASCII-Werten bestehende Begrenzungsbox wird ermittelt und in das Dokument eingefügt. Die Werte lassen sich mit einem üblichen Editor ändern. Das Ergebnis kann mit `gv` (im Beobachtungsmodus – `watch`) direkt angezeigt werden.

Mit `epstopdf` kann die `eps`-Datei in ein PDF-Dokument umgewandelt werden, das die Begrenzungsbox erhält.

### Werkzeuge

Zur Erstellung kommen vielfältige Werkzeuge in Betracht: der Klassiker `xfig`, `inkscape`, `skencil` (vormals `sketch`), `sodipodi`, `dia` ... um nur einige zu nennen. In der Artikelserie zu L<sup>A</sup>T<sub>E</sub>X wurden darüber hinaus vielfältige Möglichkeiten vorgestellt, um grafische Darstellungen zu generieren.

Zunehmend kommt dem Format SVG (Scaleable Vector Graphics), das von dem W3-Konsortium »genormt« wurde, eine wichtige Bedeutung zu, da es in Webseiten eingebunden werden kann und von den Browsern *Firefox* und *Konqueror* angezeigt wird. Der Vorteil des Formats besteht in der Skalierbarkeit und der mit dem offenen (XML-)Format verbundenen Änderungsmöglichkeit.

Sollen Bilder in ein Dokument eingefügt werden, ist auf die folgenden Randbedingungen zu achten: wird das Ergebnis gedruckt (hohe Auflösung – z. B. 600 dpi) oder kommt es für eine Präsentation (z. B. in einem PDF-Dokument) am Bildschirm zum Einsatz. Das dazu übliche Pixelgrafikformat ist PNG. Dateien in diesem Format werden problemlos in ein mit `pdfLATEX` erstelltes PDF-Dokument eingefügt.

Zur Konvertierung von Grafiken liegt eine kaum zu überschauende Anzahl von Werkzeugen vor. Daher erheben die folgenden Hinweise keinen Anspruch auf Vollständigkeit, sondern illustrieren vielmehr, wie mit kommandozeilenorientierten kleinen, spezialisierten (und selbstverständlich im Sinne der GPL freien) Werkzeugen eine Produktionskette mit maximalen Eingriffsmöglichkeiten aufgebaut werden kann.

Ein Werkzeug, um aus einem PDF-Dokument SVG-Grafiken zu erstellen steht mit `pstoedit` zur Verfügung: `pstoedit -page 1 -dt -psarg "-r9600x9600" -f sk $1.pdf $1.sk` erstellt eine `skencil`-Datei, die dann mit `skconvert $1.sk $1.svg` nach SVG umgewandelt wird.

Um innerhalb eines L<sup>A</sup>T<sub>E</sub>X-Dokuments, das mit `pdfLATEX` gesetzt werden soll, Postscript Kommandos (beispielsweise aus PSTricks) einzubetten, kann das Werkzeug `ps4pdf` genutzt werden.

Weitere nützliche Werkzeuge: `ps2pdf`, `ps2epsi`, `ps2pdf14`. Einige Werkzeuge sind Bestandteil üblicher L<sup>A</sup>T<sub>E</sub>X-Installationen. Es empfiehlt sich, bevor Sie damit beginnen, die Hilfsmittel zu installieren, zunächst mit `which foobar` zu prüfen, ob das benötigte Werkzeug bereits installiert ist.

Die Werkzeuge `ps2eps` und `epstopdf` sind unter [ftp://dante.ctan.org/tex-archive/support/ps2eps/ps2eps.zip](http://ftp://dante.ctan.org/tex-archive/support/ps2eps/ps2eps.zip) bzw. [ftp://dante.ctan.org/tex-archive/support/epstopdf.zip](http://ftp://dante.ctan.org/tex-archive/support/epstopdf.zip) verfügbar.

### Wo Hin\_de\_mit?



Wo hin soll das Bild? [ftp://dante.ctan.org/tex-archive/macros/latex/contrib/bgteubner/doc/src/bilder/bild\\_zu\\_lang2.pdf](http://ftp://dante.ctan.org/tex-archive/macros/latex/contrib/bgteubner/doc/src/bilder/bild_zu_lang2.pdf)

Denjenigen, die vormalig mit WYSIWYG-Systemen, wie OpenOffice.org gearbeitet haben, sind die »Qualen« gegenwärtig, die mit der physischen Platzierung von zusammenhängenden Elementen, wie Tabellen oder Abbildungen verbunden sind. Dies wird dank des Textsatzes und eines Satzspiegels in L<sup>A</sup>T<sub>E</sub>X vermieden. Die Elemente werden an Stellen in das Gesamtdokument eingefügt, die vom Satz her für sinnvoll erachtet werden. Dies hat allerdings zur Folge, dass im Text explizit auf die Abbildung oder die Tabelle verwiesen werden muss. Diese Verweise wiederum setzen voraus, dass alle Abbildungen eine Kennzeichnung (z. B. `\label{cap:AB-ponto}`) erhalten.

Durch Wie in `Bild-\ref{cap:AB-ponto}` ... wird der Verweis beim Satz entsprechend der korrekten Nummerierung aufgelöst.

## Wettbewerb

### Das EI:SPIEL

(von Dr. Ludger Humbert) Am 1. Juni geht es los! Das mysteriöse **EI:SPIEL** öffnet seine Räume. Entdeckt mit Freundinnen und Freunden die Welt der Informatik und lasst euch dafür belohnen. Zu finden ist das **EI:SPIEL** über [www.einstieg-informatik.de](http://www.einstieg-informatik.de)



*Einstieg Informatik – Logo Quelle:* [www.einstieg-informatik.de](http://www.einstieg-informatik.de)  
31. Mai 2006

Wir suchen Menschen ab 18, die ganz persönlich Einstieg Informatik unterstützen und **EI:SCOUT** werden wollen. Mehr dazu unter [www.einstieg-informatik.de/images/stories/pdf/ei\\_scout.pdf](http://www.einstieg-informatik.de/images/stories/pdf/ei_scout.pdf)

\*\*\*

### Volltexte Cyberpunk

Abgesehen davon, dass wir alle(?) gebundene Bücher mit hervorragendem Textsatz lieben, ist es ab und zu nützlich, in Textdokumente hineinschauen zu können, die uns die Quelle so zur Verfügung stellen, dass wir dort nach einzelnen Phrasen suchen können, mal einen Begriff im Kontext nachschauen möchten, ... Für einige Bücher des Genres Cyberpunk gibt es eine Quelle, in der Werke von William Gibson, Neal Stephenson, Bruce Sterling, Bruce Bethke (hat das Genre begründet) und Philip K Dick abgelegt sind. [www.voidspace.org.uk/library/cyberpunk.shtml#gibson](http://www.voidspace.org.uk/library/cyberpunk.shtml#gibson)

\*\*\*

### Neue Rubrik: Glossar

In dieser Ausgabe finden Sie erstmalig die Rubrik **Glossar**. Dort werden in lockerer Folge Fachbegriffe des Informatikunterrichts skizziert und gegenüber der Alltagssprache abgegrenzt. So soll nach und nach ein nützlicher Fundus entstehen.



bessere Lehrerbildung -- dank Informatik

## PISA und Informatik

(von Astrid Grabowsky) Der folgende Artikel basiert auf dem Beitrag »Die schulische Computernutzung in den Ländern und ihre Wirkungen« von Martin Senkbeil. Er befindet sich in der PISA-Länderstudie, die 2003 durchgeführt und 2005 veröffentlicht wurde (ISBN: 3-8309-1560-8).

PISA und Informatik – ist denn Informatik überhaupt Inhalt der PISA-Studie gewesen?

Die Antwort ist ganz klar: **Nein**.

Es wurden nicht die informatischen Fähigkeiten der Schülerinnen und Schüler getestet, sondern lediglich die Computerkompetenz. In dem PISA-Reigen wird dies als IT-Kompetenz bezeichnet. Und auch diese wurde nicht objektiv bewertet, sondern es wurde die »Selbstwirksamkeit« getestet, d. h. die »Überzeugung, spezifische Handlungen in diesem Bereich erfolgreich ausführen zu können« (Senkbeil). Mit anderen Worten: die Schülerinnen und Schüler wurden gefragt, wie sie selbst ihre Computerkenntnisse einschätzen.

Die Befragung wurde in drei »typische Anwendungsbereichen« durchgeführt:

- Routineanwendungen – Beispiel: »Eine Datei auf dem Computer von einem Ort an einen anderen verschieben«
- Anforderungen im Umgang mit dem Internet – Beispiel: »Ein Dokument oder eine Datei an eine E-Mail anhängen«
- Anspruchsvolle Anwendungen – Beispiel: »Mit einem Tabellenkalkulationsprogramm ein Diagramm erstellen«

Man sieht also, auf welchem Niveau sich die Fragen bewegten ... dies hat ja nun wirklich äußerst wenig mit Informatik zu tun.

Betrachten wir trotzdem, was für Erkenntnisse die Studie für die Selbsteinschätzung bezüglich der Computerkenntnisse und ihrer Verbindung mit schulischem Computereinsatz erbracht hat:

Interessant ist hier insbesondere, dass sich durch den Computereinsatz an den Schulen in Deutschland die Schere zwischen Jugendlichen, die schon von zu Hause gute Computererfahrung mitbringen, und solchen, die außerhalb der Schule keine oder nur sehr eingeschränkte Möglichkeiten der Computernutzung haben, nur noch weiter öffnet.

Immerhin ist positiv zu bemerken, dass in NRW dieser Schereneffekt nicht sehr groß ist und die Jugendlichen – im Gegensatz zu denen in anderen Bundesländern – deutlich vom Computereinsatz an den Schulen profitieren.

Erwartungsgemäß kann man an der Studie auch ablesen, dass die soziale Herkunft der Jugendlichen die Möglichkeiten der außerschulischen Computernutzung stark beeinflusst: wer also einen geringen ökonomischen, sozialen und kulturellen Status (ESCS) hat, wird eher schlechten Zugriff auf Computer haben, und wer einen hohen ESCS hat, wird eher komfortable Möglichkeiten der Computernutzung haben.

### Mein Fazit:

Keine großartigen neuen oder unerwarteten Erkenntnisse. Und vor allem: Schade, dass hier eine Möglichkeit vergeben wurde, die Bedeutung der Informatik zu untersuchen, und unser Fachgebiet wieder mal auf die Computernutzung reduziert wurde.

Weitere Infos zu PISA 2003 bei learn-line ([www.learn-line.nrw.de/angebote/slstudien/s1\\_pisa-e\\_2003.html](http://www.learn-line.nrw.de/angebote/slstudien/s1_pisa-e_2003.html))

## L<sup>A</sup>T<sub>E</sub>X – Arbeit mit Texten auch für blinde und sehbehinderte Menschen?

(von Ralf Greb) Die Textbeschreibungssprache L<sup>A</sup>T<sub>E</sub>X bietet im Informatikunterricht der Sekundarstufe I einen interessanten Zugang zum Unterrichtsinhalt »Textverarbeitung«, denn sie ermöglicht es, bei diesem Thema informatisch (d. h. strukturorientiert) vorzugehen und damit den SchülerInnen eine strukturorientierte Sicht auf Textdokumente nahezubringen. Aus diesem Grund kann L<sup>A</sup>T<sub>E</sub>X auch blinden und sehbehinderten Menschen die Möglichkeit bieten, Texte strukturiert zu erstellen und, umgekehrt, zugänglich gemacht zu bekommen.

Bei Textbeschreibungssprachen oder Auszeichnungssprachen, die jedem Textsatzsystem zugrunde liegen, handelt es sich um formale Sprachen zur logischen Beschreibung des Aufbaus von Textdokumenten. Dabei wird jeder Bestandteil eines Dokuments nach seiner Funktion (etwa als neues Kapitel samt Überschrift, als Abbildung, Tabelle, Aufzählung, Gleichung, Zitat, Inhaltsverzeichnis, etc.) ausgezeichnet (markiert) und nach Bedarf mit Attributen wie Stil, Größe und Ausrichtung versehen. Die strukturierenden Elemente werden dabei in den Text des zu erzeugenden Dokuments integriert und beschreiben seinen inhaltlichen Aufbau. Deswegen ist die Struktur, die dem Textdokument zugrunde liegt, jederzeit in dessen Quelltext verfügbar.

Im Gegensatz dazu ist es das Prinzip von WYSIWYG-Systemen (»What you see is what you get«), solange einen Text zu bearbeiten, bis er auf der Bildschirmanzeige ein ansprechendes Layout hat.

Ein auf diesen Überlegungen basierender Aspekt von Textbeschreibungssprachen besteht darin, elektronische Texte für sehbehinderte und blinde Menschen zugänglich zu machen und es Sehbehinderten zu ermöglichen, Textdokumente selbst elektronisch zu erstellen. Im Unterschied zu Dokumenten, die unter Verwendung von WYSIWYG-Systemen erzeugt werden, bieten Textbeschreibungssprachen den Vorteil, dass sie Textdokumente anhand ihrer logischen Struktur erzeugen. Blinde können deshalb lineare Textdokumente erstellen, in-

dem sie beim Eingeben des Textes diesen durch L<sup>A</sup>T<sub>E</sub>X-Befehle so strukturieren, wie sie sich seinen Aufbau vorstellen. Dadurch entsteht zusammen mit dem Textdokument eine Beschreibung seines Aufbaus – praktisch eine Anleitung zum strukturierten Vorlesen des entstandenen Textdokuments. Diese ermöglicht es, den Text Blinden und Sehgeschädigten logisch strukturiert und damit verständlich mit Hilfe geeigneter Informatiksysteme zugänglich zu machen.

Die konkrete Umsetzung dieses Ansatzes steht noch am Anfang. Ich verweise hier auf Beiträge und Berichte verschiedener Institutionen und Personen zu diesem Thema. Diese Hinweise sind im weltweiten Netz unter den angegebenen URLs verfügbar.

Michael Schäffler berichtet über seine Erfahrungen mit L<sup>A</sup>T<sub>E</sub>X an der Schloss-Schule Ilvesheim ([www.schloss-schule-ilvesheim.de](http://www.schloss-schule-ilvesheim.de)) und stellt fest: »Auf Grund dieser Erfahrung bin ich der Ansicht, dass es sich für Blinde, die L<sup>A</sup>T<sub>E</sub>X als mathematischen Code benutzen, durchaus lohnt, sich auch mit L<sup>A</sup>T<sub>E</sub>X als Satzsystem zu befassen.« Er schreibt weiter: »Die Befehle zur Gestaltung des Dokumentes stehen direkt im Text. Wer ihre Bedeutung kennt, weiß, wie sie sich auswirken und dass sie von L<sup>A</sup>T<sub>E</sub>X nach allen Regeln der Buchdruckerkunst umgesetzt werden. So gesehen ist L<sup>A</sup>T<sub>E</sub>X vielleicht die ideale Textverarbeitung für Blinde, unter dem Motto: »What you say is what you get«. Man muss L<sup>A</sup>T<sub>E</sub>X eben nur sagen, was man will. Dagegen war der Slogan »What you see is what you get« für Blinde schon immer eine zweifelhafte Verheißung« ([www.schloss-schule-ilvesheim.de/Software/demo/demo.html](http://www.schloss-schule-ilvesheim.de/Software/demo/demo.html)).

Der DVBS (Deutscher Verein der Blinden und Sehbehinderten in Studium und Beruf e. V.) beschäftigt sich auf seiner Homepage und in seiner dort verfügbaren Online-Zeitschrift **horus** u. a. mit der Rezeption elektronischer Dokumente durch Blinde und Sehbehinderte, mit L<sup>A</sup>T<sub>E</sub>X als Mathematiksschrift für Blinde und mit dem Einsatz von L<sup>A</sup>T<sub>E</sub>X schon

in der Grundschule für die Eingabe von Texten durch Blinde [vgl. [www.dvbs-online.de](http://www.dvbs-online.de)].

Einen Bericht über den Einsatz von L<sup>A</sup>T<sub>E</sub>X als Mathematiksschrift für Blinde in Schleswig-Holstein liefert Ulrike Baasch [siehe: [sonderpaedagogik.bildung.hessen.de/blind\\_sehbehindert/lamas/ErfahrungenSchleswig/](http://sonderpaedagogik.bildung.hessen.de/blind_sehbehindert/lamas/ErfahrungenSchleswig/)].

In einem Positionspapier der Kultusministerkonferenz vom Juni 2001 zur Verbesserung der Literaturversorgung für blinde und sehbehinderte Studierende heißt es unter dem Stichpunkt »Nutzung elektronischer Dokumente«: »In der Publikationswelt häufig verwendete Formate wie RTF, Postscript, PDF, QuarkXpress oder Pagemaster sind nicht unmittelbar und vollständig für Blinde oder Sehbehinderte nutzbar. Textanteile, die in der Regel mittels Filterprogrammen problemlos aus den Datenformaten extrahiert werden können, verlieren während der Konvertierung ihre Struktur und sequentielle Ordnung. Das macht ein zeitaufwendiges Nacharbeiten der Texte durch Sehende erforderlich. Auch Bild- und Grafikanteile bedürfen einer manuellen Umsetzung bzw. Beschreibung durch Sehende. Entscheidende Vorteile bieten strukturierte Dokumentenformate wie L<sup>A</sup>T<sub>E</sub>X, HTML oder XML. Bei Einsatz der in diesen Formaten vorhandenen Strukturierungsmittel können komfortabel blinden- und sehbehindertengerechte Darstellungen erzeugt werden. Derzeit sind erste Softwarewerkzeuge wie Browser und Konvertierungsprogramme in Entwicklung (z. B. DAISY Projekt). Die Entwicklungen stehen hier erst am Anfang. Jedoch liegt auf diesem Gebiet, verbunden mit einer intensiven Kooperation mit Autoren und Verlagen, ein enormes Potenzial für eine entscheidende Verbesserung des Informationszuganges Blinder und Sehbehinderter« [Verbesserung der Literaturversorgung für blinde und sehbehinderte Studierende; Positionspapier der Kultusministerkonferenz vom 28.06.2001] vgl. [www.kmk.org/doc/publ/posblind.pdf](http://www.kmk.org/doc/publ/posblind.pdf)

## Datenschutz und Schule

### Konsequenzen der Datenschutzgesetzgebung für Schulen und Lehrkräfte

(von Ralf Greb / Markus Hufnagel) Für uns Informatiker ist es wichtig, die Kompetenzen unseres Faches auch bei gesellschaftlichen Fragen und Problemstellungen zur Verfügung zu stellen. Ein wichtiges Feld, auf dem dies geschehen kann – ja muss – ist der Datenschutz. Dabei ist für uns als angehende Lehrer die Frage des Datenschutzes in der Schule und im Umfeld von Schule von immenser Bedeutung. In diesem Zusammenhang haben die Autoren dieses Berichtes im Rahmen einer Lehrerkonferenz am Städtischen Gymnasium Sundern einen Vortrag mit dem Thema »Datenschutz und Schule« gehalten, dessen Zusammenfassung diesen Bericht bildet. Insbesondere wird auf die Konsequenzen der Datenschutzgesetzgebung für die Lehrerinnen und Lehrer eingegangen.

Die Datenschutzgesetzgebung soll sicherstellen, dass personenbezogene Daten nur zweckgebunden verarbeitet werden. Dabei muss gewährleistet sein, dass das informationelle Selbstbestimmungsrecht des Einzelnen gewahrt bleibt, nämlich das Recht auf Auskunft und Berichtigung fehlerhafter Daten, das Recht auf Sperrung und Löschung von Daten und das Recht auf Schutz der Daten vor unbefugtem Zugriff. Diese Vorschriften werden geregelt durch das Datenschutzgesetz des Landes Nordrhein-Westfalen aus dem Jahr 2000 (zuletzt geändert 2003) und gelten insbesondere für Schulen, Lehrerinnen und Lehrer und ihren Umgang mit den Daten der Schüler-

innen und Schüler. Für die Einhaltung der Vorschriften ist die Schulleiterin respektive der Schulleiter verantwortlich. Die Schulleitung muss die Einhaltung der Vorschriften durch technische und organisatorische Maßnahmen sicherzustellen.

Datenschutz in der Schule selbst bedeutet insbesondere die Trennung des Schulverwaltungsrechners von anderen Netzen (besonders vom Internet), den Schutz für Rechner mit Zugang zu Schülerdaten durch Passwortzugang (Accounting) und das Abschließen der Räume, in denen Akten oder Rechner stehen.

Datenschutz für Lehrerinnen und Lehrer bedeutet aber noch mehr. Lehrerinnen und Lehrer, die personenbezogene Daten von Schülerinnen und Schülern zuhause auf ihren privaten Rechnern speichern und verarbeiten, benötigen eine schriftliche Genehmigung der Schulleitung. Diese erhalten sie, wenn die Verarbeitung der betreffenden Schülerdaten für die Erfüllung der schulischen Aufgaben erforderlich ist und wenn ein angemessener technischer Schutz der Daten zugesagt wird. Der diesbezügliche Antrag ist bei der Schulleitung einzureichen und beinhaltet Fragen nach Art, Herkunft und Weitergabe der Daten, nach der Dauer der Speicherung, nach benutzter Hardware und Software, nach Vorhandensein eines Internetzugangs und nach entsprechenden Schutzmaßnahmen.

Einzuhaltende Schutzmaßnahmen sind dabei unter anderem: Passwortschutz am Rechner, ein abschließbares Arbeitszimmer, bei Rechnern mit Internetzugang eine Firewall, ein regelmäßig aktualisierter Virenschutz und die regelmäßige Sicherung der Daten.

Natürlich müssen diese dann auch gesichert aufbewahrt werden. Darüber hinaus ist besonders im Umgang mit Wechseldatenträgern große Sorgfalt geboten. Die datenschutzrechtlich relevanten Daten auf ihnen sollten verschlüsselt sein, für den Fall, dass sie verlorengehen. Weiterhin müssen nicht mehr benötigte SchülerInnen-Daten zeitnah gelöscht werden.

Verkauft einen Lehrer seinen Computer oder auch Wechseldatenträger (hier vor allem USB-Sticks), so muss sichergestellt sein, dass sämtliche Schülerdaten tatsächlich von den Geräten entfernt worden sind. Ein einfaches Löschen reicht hierzu im allgemeinen nicht aus, da gelöschte Daten wiederhergestellt werden können. Ein qualitativ bessere Möglichkeit besteht darin, die Festplatte zu formatieren.

### Quellen:

Datenschutzgesetz Nordrhein-Westfalen – zu erhalten unter: [www.lfd.nrw.de/fachbereich/fach\\_3\\_1\\_1.html](http://www.lfd.nrw.de/fachbereich/fach_3_1_1.html)

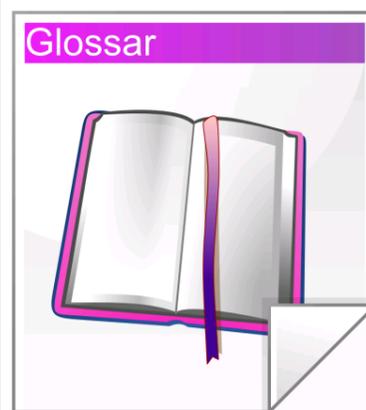
Verordnung über die Verarbeitung zugelassener Daten von Schülerinnen, Schülern und Erziehungsberechtigten. zu erhalten unter: [www.lfd.nrw.de/fachbereich/fach\\_3\\_2\\_37.html](http://www.lfd.nrw.de/fachbereich/fach_3_2_37.html)

Aktuelle Hinweise zum Datenschutz: Virtuelles Datenschutzbüro ([www.datenschutz.de](http://www.datenschutz.de))



Steht Datenschutz nur in Regalen? [www.lfd.nrw.de/pix/regal.gif](http://www.lfd.nrw.de/pix/regal.gif)

## Glossar: Verzeichnis



(von Dr. Ludger Humbert) Mit dieser Ausgabe der If Fase wird die Rubrik Glossar eingeführt. In lockerer Folge sollen Begriffe vorgestellt werden, denen im Informatikunterricht eine große Bedeutung zukommt, die aber nicht einheitlich verwendet werden. Die Ursachen sind vielfältig:

- Falsche Übersetzung aus dem angelsächsischen Sprachraum
- Vermischung der Umgangssprache mit der Fachsprache
- Fachsprache versus Benennung in Handbüchern, Benutzungsoberflächen, ...
- Es haben Begriffe Eingang in die Informatik gefunden, die aus Anwendungsbereichen stammen

Gerade bei der zunehmenden Verbreitung von Abkürzungen, die mehr recht als schlecht in die deutsche Sprache übertragen werden, stellen sich u. a. folgende Fragen:

1. Welchen Begriff soll ich im Unterricht verwenden?
2. Darf ich oder muss ich Begriffe festlegen?
3. Wie kann ich dafür Sorge tragen, dass die Schülerinnen einen fachsprachlich korrekten Begriff verwenden und nicht die Begriffe, die z. B. in einer Benutzungsoberfläche verwendet werden?

Der konstruktiven Diskussion um die **Fachsprache im Informatikunterricht** kommt eine große Bedeutung zu. »Falsche« Begriffe sorgen für Irritation, für ungewollte/unbeabsichtigte Lernwiderstände, für Mißverständnisse und für eine fehlerbehaftete Argumentation im Fachkontext. Diese Rubrik will eine Lücke füllen, die bisher in der Fachdidaktik nur selten thematisiert wird.

\* \* \*

### Verzeichnis

Zunehmend tritt der englische Begriff »**directory**« (deutsch: **Verzeichnis** – aber auch – **Katalog**) gerade im Zusammenhang mit grafischen Benutzungsoberflächen in den Hintergrund.

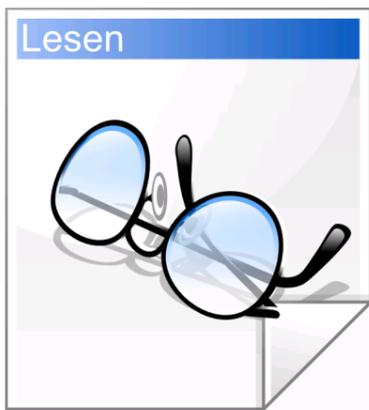
Häufig wird unreflektiert die Schreibmethode verwendet, in der der Begriff »**folder**« (deutsch: **Ordner**) verwendet wird.

Betrachten wir die Situation als Informatiker, so richten wir den Blick zunächst auf die dahinterstehende fachliche Struktur: es handelt sich um eine Konstruktion zur Hierarchisierung von Dateien. Diese Konstruktion ist nicht die einzig mögliche, wird aber zur Zeit fast ausschließlich verwendet (Vergleiche dazu die Ansätze von Ted Nelson und seine frühen Versuche, nicht hierarchisch strukturierte Organisationsformen zu propagieren).

In Informatikbüchern finden wir inzwischen den Begriff **Ordner**. In den Begriffen Verzeichnisbaum, Verzeichnisstruktur, ... wird anschließend unreflektiert das Präfix Verzeichnis durch Ordner ersetzt. Im aktuellen Informatikduden findet sich dieser Begriff weder als eigenes Stichwort noch im Register. Das sollte allen zu denken geben, die diesen Begriff **noch** im Unterricht verwenden.



Lesen



(von Dr. Ludger Humbert)

**The Access Principle: The Case for Open Access to Research and Scholarship**

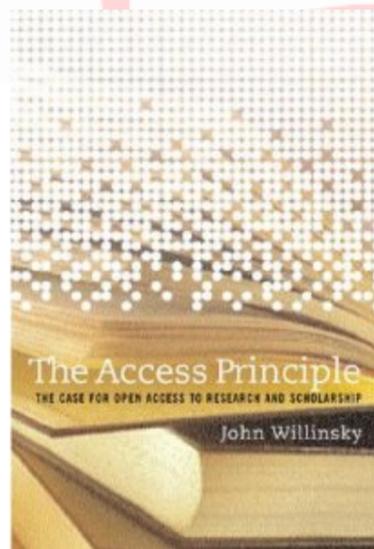
Die Situation ist bekannt und m. E. auch bezeichnend: in den westlichen Staaten lernen wir in der universitären Ausbildung andere Positionen, Hintergründe, weitere Quellen kennen, in dem wir in die Universitätsbibliothek gehen und dort die verfügbaren aktuellen (aber auch weiter zurückliegenden) Veröffentlichungen studieren.

Sobald wir im Studienseminar oder in der Schule einen solchen Service erwarten, werden wir mit einer völlig anderen Situation konfrontiert:

Nach und nach wurde – unter reinen Kostengesichtspunkten – dort sogar der Bezug aktueller Zeitschriften reduziert oder völlig eingestellt. In den ersten Jahren meiner Tätigkeit als Lehrer fuhr ich regelmäßig am Samstag in die Bibliothek der FernUniversität, um mal wieder aktuelle Veröffentlichungen studieren zu können.

In vielen Ländern ist heute schon der Zugang zu Wissen in Form von Zeitschriften oder teuren Konferenzbänden nicht mehr möglich (auch nicht in den Universitäten). Hier stellt sich in aller Schärfe die Frage nach dem Zugang zu Wissen, das – häufig genug mit staatlicher Unterstützung – geschaffen wurde und wird; während die Vermarktung privatisiert ist. Eine aktuelle Variante von Verlagen besteht darin, dass eine Autorin Geld dafür bezahlen muss, wenn sie ihre Veröffentlichung auch unter einer offenen Lizenz zugänglich machen will. Diese Pervertierung des Prinzips der Zugänglichkeit des Wissens wurde unter dem Buchtitel »Freie Kultur« von Lawrence Lessig angeprangert – wir berichteten in der Ausgabe 7 der *If Fase* vom 1. März 2006.

Unter **The Access Principle: The Case for Open Access to Research and Scholarship** findet sich ein weiterer Titel, der sich diesem Problem widmet. Autor ist *John Willinsky*.



The Access Principle

- ISBN: 0262232421
- Quanta Costa: 34.97 Euro

Das Buch ist auch im Volltext im Netz verfügbar [https://mitpress.mit.edu/books/willinsky/TheAccessPrinciple\\_TheMITPress\\_0262232421](https://mitpress.mit.edu/books/willinsky/TheAccessPrinciple_TheMITPress_0262232421).

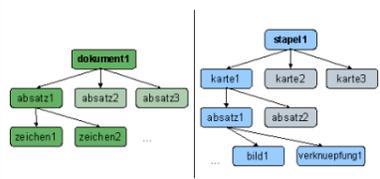
**WontoML – Mobilkommunikation und Informatikunterricht**

(von *Martin Reinertz*) Mobilkommunikation als thematische Basis des Informatikunterrichts erscheint bereits auf den ersten Blick als vielversprechend und verlockend. Zum einen stellen Mobiltelefone (»Handys«) prinzipiell nichts anderes als Informatiksysteme dar (die sich zunehmend auch in ihrer Leistungsfähigkeit an »herkömmliche« Systeme annähern), zum anderen sichert ihre große Verbreitung und Beliebtheit unter Schülerinnen und Schülern eine hohe Motivationslage, wie Befragungen in weiterführenden Dortmunder Schulen belegen. Für eine tiefergehende Betrachtung stellt sich daher die Frage, ob und in welchem Ausmaß sich bislang als wertvoll betrachtete Ausrichtungen des Informatikunterrichts am Themenkomplex Mobilkommunikation vermitteln lassen.

Weit etabliert hat sich als richtungsweisender Fokus großer Anteile des Informatikunterrichts u.a. das objektorientierte Paradigma. Vor diesem Hintergrund wird mit WontoML eine Schnittstelle (Klassenbibliothek) für die Programmiersprache Python vorgestellt, mittels derer objektorientierte Modellierung anwendungsbezogen am Themenfeld Mobilkommunikation vermittelt (und komplementärerweise erlernt) werden kann.

**Aha ... und was genau kann ich mit WontoML machen?**

Zusammengefasst handelt es sich bei WontoML um eine Python-Klassenbibliothek zur objektorientierten Generierung von Dokumenten gemäß WML 1.2 ([en.wikipedia.org/wiki/Wireless\\_Markup\\_Language](http://en.wikipedia.org/wiki/Wireless_Markup_Language)), die auf nahezu jedem heutzutage erhältlichen Mobiltelefon mit WAP-Browser angezeigt werden können. Aufgrund eben dieser maximalen Verbreitung kompatibler Browser wurde auf eine Realisierung von WontoML auf Basis einer moderneren Markup-Sprache zur Mobilitätsunterstützung, etwa XHTML MP ([de.wikipedia.org/wiki/XHTML\\_Mobile\\_Profile](http://de.wikipedia.org/wiki/XHTML_Mobile_Profile)), verzichtet. Die von WontoML zur Verfügung gestellten Klassen entsprechen zum Großteil etablierten Metaphern für die Bestandteile von WML-Seiten (Stapel, Karte, Absatz, etc.). Die Klassenhierarchie selbst lehnt sich an die für die Sekundarstufe I in Bayern vorgeschlagene Struktur zur objektorientierten Modellierung von Textdokumenten an. Einige grundlegende Klassenbezeichner (bzw. davon instanziierte Objekte mit entsprechenden Namen) von WontoML sowie besagte Anlehnung an das bayrische Konzept sind in Abb. 1 überblickhaft dargestellt. Ein Quellcode-Beispiel, das die wesentlichen Elemente der Generierung einer »Hallo Welt«-WML-Seite mittels WontoML berschreibt, steht unter [www.ham.nw.schule.de/bscw/bscw.cgi/d315457/hallowelt.py](http://www.ham.nw.schule.de/bscw/bscw.cgi/d315457/hallowelt.py) zur Verfügung. Die vollständige Klassenhierarchie von WontoML ist jedoch um einiges komplexer; die komplette und ausführlich kommentierte API-Referenz kann unter [www.wontoml.de/vu](http://www.wontoml.de/vu) eingesehen werden.



**Abb. 1:** Vergleich – Aufbau des »Objektbaums« eines Textdokuments im Pflichtfach Informatik in Bayern (links) und des Objektbaums eines WML-Dokuments in WontoML (rechts). Klassenbezeichner entsprechen jeweils den Namen der dargestellten Objekte ohne Zahlenangabe. Der erste Buchstabe von Klassenbezeichnern wird in WontoML außerdem grundsätzlich GROSS geschrieben. D.h. *stapel1* ist Instanz der Klasse *Stapel*, *karte2* Instanz der Klasse *Karte*, etc.

Der besondere Clou an WontoML im Vergleich zur eher statischen Modellierung von Dokumenten im bayrischen Konzept für die Sekundarstufe I ist allerdings, dass WontoML-basierte Python-Programme als CGI-Skripte eingesetzt werden können und WontoML in diesem Rahmen Konstrukte zur Interaktion mit dem (eine entsprechende Website besuchenden) Benutzer zur Verfügung stellt. So ist es z. B. möglich, Texteingabefelder zu instantiieren und zu WML-Seiten hinzuzufügen oder bei Betätigung einer Softkey ([de.wikipedia.org/wiki/Softkey](http://de.wikipedia.org/wiki/Softkey)) WML-internen Variablen Werte zuzuweisen, die wiederum mittels POST anderen Python-Skripten zugänglich gemacht werden und von diesen ausgewertet werden können. Es ist die Verwendung eben solcher Klassen zur Interaktion mit dem Benutzer, die Unterrichtsprojekte auf Basis von WontoML schnell so komplex werden lassen, dass sie für die Sekundarstufe II als angemessen und mitunter auch anspruchsvoll angesehen werden können. Dies beantwortet dann auch die von *Ralph Carrie* in einem ersten Erfahrungsbericht zu WontoML (bzgl. eines Einsatzes im Rahmen des Informatiktags 2006 in Paderborn) gestellte Frage, »ob das Ponto-Konzept [gemeint ist das Konzept der objektorientierten Modellierung von Textdokumenten nach Hubwieser, Anm. des Autors] nicht die gleichen Konzepte vermittelt und wozu dann WontoML?« ([humbert.in.hagen.de/iffase/Artikel/2006-04-18\\_carrie-WontoML.html](http://humbert.in.hagen.de/iffase/Artikel/2006-04-18_carrie-WontoML.html)).

Fakt ist in diesem Zusammenhang sicherlich, dass die objektorientierte Modellierung von Textdokumenten im bayerischen Konzept (und die praktisch orientierte Umsetzung Ponto ([ddi.cs.uni-dortmund.de/projekte/ponto.html](http://ddi.cs.uni-dortmund.de/projekte/ponto.html))) sowie die mit WontoML vorgeschlagene objektorientierte Modellierung und Generierung von WML-Dokumenten auf nahezu denselben didaktischen Annahmen hinsichtlich der Vermittlung von Konzepten (wie etwa objektorientierte Modellierung als sinnvoller Bestandteil des Unterrichts) fußen. Fakt ist aber auch, dass die konzeptionelle Komplexität von WontoML diejenige von Ponto u.a. durch die Möglichkeit zur Interaktion mit dem Benutzer (und der damit verbundenen dynamischen Generierung von WML-Seiten) um ein Vielfaches übersteigt. Es muss beachtet werden: Während Ponto vom Aufbau her allenfalls für eine Einführung in die Objektorientierung dacht ist, bietet WontoML auch vielfältige Möglichkeiten zur Vertiefung des Konzepts (z. B. Polymorphie bei Parameterübergabe, abstrakte Klassen, etc.).

**Soweit, so gut – aber wer soll die entstehenden Handy-Rechnungen bezahlen?**

Einer der größten Einwände, die bislang gegen WontoML geäußert wurden, ist die (berechtigte) Frage, ob ein schulischer Einsatz nicht zwangsläufig die Handy-Rechnungen der Schülerinnen und Schüler in die Höhe treiben würde – schließlich müssen die generierten Seiten mit einem WAP-Browser betrachtet werden; nur so können die Lernenden die Korrektheit ihrer Konstruktionen überprüfen. Glücklicherweise existiert eine Alternative zur tatsächlichen Verwendung von Mobiltelefonen – und zwar in Form von zahlreichen (für nahezu jede Plattform erhältlichen) WAP-Browseremulatoren (s. Abb. 2). Derartige Emulatoren ahmen WAP-Funktionen (und häufig auch Aussehen) eines Mobiltelefons nahezu täuschend echt nach. Steht für das in der eigenen Schule verwendete Betriebssystem kein Emulator zu Verfügung, können die erstellten WontoML-basierten Python-CGI-Skripte oder ggfs. die statisch mittels WontoML erstellten WML-Seiten (auch eine statische Generierung ist möglich) über einen öffentlich zugänglichen Webserver mittels eines Online-WAP-

Browsers betrachtet werden (z. B. Wapsilon ([www.wapsilon.com](http://www.wapsilon.com)) oder Waptiger ([www.waptiger.com/waptiger](http://www.waptiger.com/waptiger))).



**Abb. 2:** Darstellung einer mittels WontoML generierten WML-Seite im WAP-Browseremulator des Herstellers Openwave. Konstituierende Objekte für diese Seite sind Instanzen der Klassen *Stapel*, *Karte*, *Absatz*, *Zeichen* und *Bild* (zum Verständnis dieser Metaphern kann eine beliebige Beschreibung der WML-MarkUp-Sprache herangezogen werden).

In diesem Zusammenhang äußert allerdings *Carrie* wiederum, dass seiner Ansicht nach »der Einsatz des Mobiltelefons [...] innerhalb des WontoML-Konzeptes [...] doch etwas an den Rand [rückt]. Man erzeugt Seiten für das Mobiltelefon und benutzt (aus Kostengründen) einen Handy-Emulator.« Dieser Kritikpunkt mag zunächst verständlich sein, jedoch sollte bedacht werden, dass Unterricht notwendigerweise grundsätzlich einen gewissen Anteil *Simulation* beinhaltet. Ob nun fremdsprachlicher Unterricht, in dem die Begegnung mit der anderssprachlichen Kultur größtenteils ebenfalls nur *simuliert* wird oder Informatikunterricht, in dem Modellierung zwar mit Anwendungsbezug, jedoch i. d. R. in *reduzierten* Kontexten behandelt wird – unterrichtlicher Rahmen bedeutet auch ein gewisses Maß an Einschränkung. Vor diesem Hintergrund stellt es dann aber vermutlich eher zusätzliche Motivation für die Lernenden dar, dass WontoML eben gerade die Möglichkeit bietet, die unterrichtlich entwickelten Produkte später auch mittels eines realen Mobiltelefons zu testen (obgleich die ständige Verwendung des Mobiltelefons im Unterricht aufgrund der damit verbundenen Kostenentwicklung sicherlich nicht möglich bzw. sinnvoll ist). So stellt *Carrie* dann auch weiter fest: »Argumente, die sicherlich für den Einsatz von WontoML sprechen sind, dass die Schülerinnen und Schüler keine Mobiltelefone haben müssen und auch keine Kosten entstehen (müssen). Zusätzlich ist es für Schülerinnen und Schüler sicherlich motivierend – angesichts der zunehmenden Bedeutung von Mobiltelefonen.«

**Hinweise**

Der vorliegende Artikel stellt einen kurzen Abriss auf WontoML und einen sehr eingeschränkten Blick auf die Funktionsfülle und der unterrichtlichen Einsatzmöglichkeiten dar. Hinweise zu WontoML, inkl. des kompletten (ausführlich kommentierten) Quellcodes, der API-Referenz, eines vollständigen Klassendiagramms, einer Installationsanleitung, Beispielen für Aufgaben und Unterrichtsprojekte mit WontoML etc. finden sich unter: [www.wontoml.de/vu](http://www.wontoml.de/vu) Dort kann auch (nach Freischaltung durch den Autor) die vollständige Arbeit heruntergeladen werden.



Logo: WontoML

Datenbank ohne SQL



**Datenbanken im Informatikunterricht**

(von *Dr. Ludger Humbert*) Je nach Bundesland finden sich in der BRD unterschiedliche Vorgaben in der gymnasialen Oberstufe bezüglich der Inhalte. Allerdings wurde spätestens mit der Verabschiedung der aktuell gültigen EPA (Einheitliche Prüfungsanforderungen für das Abitur) durch die Kultusministerkonferenz (KMK) im Jahr 2003 deutlich, dass zumindest in der Vorbereitungsgemeinschaft eine große Mehrheit die Notwendigkeit der Datenbankmodellierung im Informatikunterricht unterstützt (vgl. [www.kmk.org/doc/beschl/EPA-Informatik.pdf](http://www.kmk.org/doc/beschl/EPA-Informatik.pdf)). Immerhin wurden vier der 20 Beispielaufgaben so formuliert, dass sie mit Hilfe einer Datenbankmodellierung (ER und SQL werden explizit benannt) bearbeitet werden müssen.

Nun stellt sich – wie für viele Gegenstände des Informatikunterrichts – die Frage nach der Umsetzung und den dazu unabdingbaren Hilfsmitteln, den Werkzeugen und den Techniken, die bei der Modellierung nützlich sind. Aus der Sicht der Didaktik können – ausgehend vom wissensbasierten Paradigma – solche Modellierungen mit Mitteln spezialisierter Inferenzmaschinen durch eine Reihe von Fakten und Regeln vorgenommen werden (häufig mit der Programmiersprache Prolog umgesetzt). Die zweite Variante besteht in der Regel in der Modellierung mit Hilfe von Entity-Relationship-Diagrammen, die dann mit einer Datenbank implementiert werden. Als dritte Variante bietet sich an, die (auch für andere Aufgabenbereiche notwendige) objektorientierte Modellierung (OOM) zu verwenden.

```
from buzhug import Base
from datetime import date
# erstelle Tabelle 'teams'
teams = Base('teams')
teams.create(("name",str),("stadt",str))
```

Wie dem Codeschnipsel entnommen werden kann, wird hier das Objekt *teams* aus der Klasse *Base* erzeugt und verfügt über die Methode *create*. Daneben existieren die Methoden: *open* (um eine bestehende Tabelle zu öffnen), *insert* (um einen Datensatz einzufügen) *teams.insert(name="Arsenal", stadt="London")* *update* (um einen Datensatz zu ändern). Wie kann ein Datensatz gefunden werden?

```
datensatz=[r for r in teams if r.city == "London"] [0]
```

liefert den Datensatz, der dann z.B. mit *teams.update(datensatz, name="Aston Villa")* geändert werden kann. Weitere Hinweise (inkl. Tutorial) finden sich über [buzhug.sourceforge.net](http://buzhug.sourceforge.net). Ich bin der Überzeugung, dass eine OOM für Aufgaben aus diesem Kontext eine erheblich bessere Integration in den Informatikunterricht ermöglicht, da weiterhin mit einer bereits bekannten Problembeschreibungstechnik gearbeitet werden kann.

Soll mit erstellten Datenbanktabellen gearbeitet werden, die SQL-kompatibel sind, so lohnt ein Blick in SQLite – eine kleine Bibliothek, mit der transparent auf SQL-Dateien zugegriffen werden kann: [pysqlite.org](http://pysqlite.org)