

# **Der Einsatz von Object-Draw und EOS im Informatikunterricht der Jahrgangsstufen 6 und 7 am Gymnasium**

**(Martin Pabst, Descartes-Gymnasium Neuburg)**

Falls Sie Fragen zu den Programmen oder Anregungen für ihren unterrichtlichen Einsatz haben, können Sie mir gerne mailen: [martin.pabst@gmx.net](mailto:martin.pabst@gmx.net)

Die neuesten Programmversionen können Sie auf meiner Homepage herunterladen: <http://pabst.heim.at>

Passende Inhalte des Lehrplans für Gymnasien in Bayern:

## **NT 6.2:**

„Bei der problemorientierten Arbeit am Rechner erleichtert ihnen die objektorientierte Sichtweise, die gemeinsamen Grundprinzipien und Strukturen von Dokumenten zu erkennen. Damit lernen die Schüler im Laufe des Schuljahrs ein tragfähiges, produktunabhängiges und zeitbeständiges Konzept für die Darstellung von Information mittels Standardsoftware kennen und entwickeln ein allgemeines Verständnis für den prinzipiellen Aufbau dieser Softwaresysteme.“

### **NT 6.2.2 Informationsdarstellung mit Graphikdokumenten – Graphiksoftware**

„Den Schülern wird bewusst, dass man mittels Graphiken einfach und effektiv Information darstellen kann. Spielerisch und intuitiv gelingt es ihnen, reale Situationen zielgerichtet in Graphiken abzubilden. Überlegungen zur Struktur von Graphiken führen zur objektorientierten Sichtweise. Die Schüler erkennen, dass jedes Objekt der Graphik bestimmte Eigenschaften hat und einer Klasse gleichartiger Objekte zugeordnet ist. Bei der praktischen Arbeit mit Graphikprogrammen wird ihnen auch die Notwendigkeit einer einfachen, einheitlichen Beschreibungssprache zur eindeutigen und effektiven Verständigung deutlich.

- Objekte als Informationseinheiten in Graphiken
- Objekte einer Vektorgraphik: Attribut, Attributwert und Methode
- Beschreibung gleichartiger Objekte durch Klassen: Rechteck, Ellipse, Textfeld, Linie“

### **NT 6.2.3 Informationsdarstellung mit Textdokumenten – Textverarbeitungssoftware**

[...] Die Enthält-Beziehung zwischen Objekten; Entwerfen von Objekt- und Klassendiagrammen

### **NT 7.2.3 Beschreibung von Abläufen durch Algorithmen**

Die Schüler lernen eines der wichtigsten Grundprinzipien der automatischen Informationsverarbeitung kennen und erhalten einen ersten Einblick in seine Anwendung. Sie lernen, dass sich ganz allgemein mit Algorithmen Abläufe präzise und verständlich beschreiben lassen, und üben an konkreten Sachverhalten, insbesondere naturwissenschaftlichen Experimenten, Vorgänge aus einfachen Bausteinen aufzubauen. Dabei arbeiten sie mit einem Programmiersystem, mit dem sie die Algorithmen intuitiv umsetzen können und bei dem die Einzelschritte des Ablaufs altersgemäß visualisiert werden.

[...]

- Bausteine von Algorithmen: Anweisung, Sequenz, Bedingte Anweisung, Wiederholung
- Programmieren eines einfachen Informatiksystems unter Verwendung dieser Bausteine

## Unterrichtssequenz für die Jahrgangsstufe 6:

Die folgende Unterrichtssequenz deckt NT 6.2.2 ab. Im Lehrplan sind 8 Stunden vorgeschlagen. Da auch der Inhalt „Enthält-Beziehung zwischen Objekten“ schon angeschnitten wird, ist die Dauer von 9 Stunden sicher zu verantworten.

Die Zeitangaben sind nur grobe Schätzungen. Zur Sicherheit sollte man noch zusätzliche Aufgaben parat haben.

### 1. Stunde: Erstellung einfacher Grafiken mit Object-Draw

20 Min.:

Zunächst lernen die Schüler spielerisch Object-Draw kennen. Vorschlag: Am Beamer zeigt man ihnen das Bild „Haus mit LKW“ und fordert sie auf, es möglichst getreu nachzuzeichnen. Das Analysatorfenster bleibt während dieser ersten Phase unbeachtet. Der Lehrer geht durch die Reihen und hilft. Alle 5 Minuten wird die Arbeit kurz unterbrochen, um im Unterrichtsgespräch aufgetretene Probleme zu sammeln und deren Lösung durch Schüler am Beamer erklären zu lassen.

Insbesondere sollte angesprochen werden:

- Objekte können mit Hilfe der Entf-Taste gelöscht werden
- Wie können Objekte verschoben, vergrößert oder gedreht werden?
- Beim Aufziehen, Verschieben, Vergrößern und Drehen von Objekten können durch gleichzeitiges Festhalten der Strg- oder Umschalttaste nützliche Nebenbedingungen gesetzt werden.

15 Min.:

Nach Abschluss der ersten Phase werden die Schülerrechner deaktiviert. Mit Hilfe des Grafikdokuments „Haus mit LKW“ wird durch Betrachten der Objektkarten im unteren Teil des Analysatorfensters im Unterrichtsgespräch erarbeitet:

- Das Graphikdokument ist aus einzelnen Objekten zusammengesetzt
- Jedes Objekt hat einen eindeutigen Namen (Bezeichner)
- Die Merkmale des Objekts heißen Attribute. Den Wert eines Attributs nennt man Attributwert.

5 Min.:

Nachdem dies in einem kurzen Hefteintrag gesichert wurde, öffnen alle Schüler das Grafikdokument „Haus mit LKW“ und bearbeiten folgende Aufgaben (Aufgabenstellung z.B. als Präsentation am Beamer, Lösungen schriftlich ins Heft):

1. Öffne das Grafikdokument „Haus mit LKW“
2. Nenne drei Objektbezeichner!
3. Welche Attribute besitzt das Objekt Ladefläche?
4. Welchen Attributwert besitzt das Attribut Füllfarbe des Objekts Hausfront?
5. Nenne zwei weitere Attribute des Objekts Hausfront zusammen mit ihren Attributwerten. Notiere sie so ins Heft:

Objektbezeichner.Attributname = Attributwert  
z.B.: Hausfront.Länge = 2.70 cm

Die Lösungen der Aufgaben werden kurz miteinander besprochen.

## 2. Stunde: Dokumente und Dateien

5 Min.:

Wiederholung der wichtigsten Inhalte der letzten Stunde

10 Min.:

Die Zeichenfunktionen von Object-Draw werden an einem einfachen Beispiel erneut eingeübt. Am Ende dieser Phase macht die Lehrkraft darauf aufmerksam, dass es doch schön wäre, wenn man beim Zeichnen nicht jedes Mal vorne beginnen müsste, sondern an den Zeichnungen der letzten Stunde weiterarbeiten könnte.

5 Min.:

Im Klassenverband wird erarbeitet:

- Die Informationen im Rechner gehen beim Ausschalten verloren, es sei denn, man speichert sie in Dateien ab.

Ein entsprechender Merksatz wird ins Heft notiert.

10 Min.:

Das Speichern und Öffnen von Dateien wird gezeigt und praktisch geübt. Zur Vertrauensbildung kann zwischen Speichern und Öffnen auch mal der Rechner heruntergefahren werden.

10 Min.:

Die Lehrkraft zeigt den Schülern, wie die Informationen des Grafikdokuments in der Datei dargestellt werden (Wiederholung zu 6.2.1!). Dazu öffnet er vorne am Rechner ein einfaches Zeichendokument (z.B. „Sehr einfaches Auto“) mit Object-Draw. Parallel dazu öffnet er das selbe Dokument mit dem Editor. Gemeinsamkeiten werden gesucht. Es kann auch besprochen werden, warum die Darstellung als Datei nicht so aussehen kann wie auf dem Bildschirm.

**Bemerkung:** Damit Sie die Fragen der Schüler nicht in Verlegenheit bringen, gibt es im Anhang dieses Dokuments eine ausführliche Beschreibung der xml-Struktur, in der Object-Draw die Grafikdokumente speichert.

## 3. Stunde: Objekte verändern mit Methoden

5 Min.:

Wiederholung der wichtigsten Inhalte der letzten zwei Stunden

10 Min.:

Die Lehrkraft erklärt den Schülern, dass Objekte die Fähigkeit haben, bestimmte Dinge zu tun: sie besitzen **Methoden**. Jede Veränderung eines grafischen Objekts geschieht mit Hilfe eines geeigneten Methodenaufrufs. In manchen Zeichenprogrammen (CAD-Programme) ist es üblich, die Methodenaufrufe per Tastatur einzutippen. Um es dem Benutzer einfacher zu machen, lassen sich die meisten Programme per Maus bedienen. Aus den „Mausaktionen“ der Benutzer generieren sie automatisch im Hintergrund die

passenden Methodenaufrufe für die bearbeiteten Objekte. In Object-Draw kann man diese Aufrufe im Analysatorfenster „live“ ansehen. Die Lehrkraft zeigt dies an einem einfachen Beispiel (Verschieben der Sonne im Dokument „Haus mit LKW“ und stellt als Aufgabe an die Schüler, durch Ausprobieren möglichst viele Methodenaufrufe zu finden und im Heft zu notieren. Sie sollen überdies herausfinden, was die seltsamen Angaben in Klammern bedeuten.

10 Min.:

Die Ergebnisse der Aufgabenstellung werden gesammelt. Dabei wird herausgearbeitet:

- Die Schreibweise für einen Methodenaufruf ist  
Objektname.Methodenname(Parameter)
- Die Parameter geben genauer an, wie die Methodenausgeführt werden soll (z.B. Verschiebung in x- und y-Richtung bei der Methode verschieben)
- Evtl.: Nicht alle Objekte haben die selben Methoden.

In einem kurzen Hefteintrag wird das Wesentliche festgehalten.

5 Min.:

Im Unterrichtsgespräch wird erarbeitet, dass man die möglichen Methodenaufrufe für ein Objekt aus dessen Klassenkarte ersehen kann. Die Lehrkraft zeigt, wie in Object-Draw Methodenaufrufe auch per Hand eingegeben werden können. Die Schüler dürfen dies 5 Minuten lang an einer selbst erstellten Zeichnung ausprobieren. Aufgetretene Probleme werden im Klassenverband geklärt.

10 Min.:

Die Lehrkraft stellt als Aufgabe die „Übung zu Methodenaufrufen 1“ (Bem.: das Passwort ist „passwort“!). Die Schüler bearbeiten die Aufgabe, die Lösung wird anschließend gemeinsam besprochen.

#### **4. Stunde: Klassen**

5 Min.:

Wiederholung der Inhalte der letzten Stunde anhand einer praktischen Aufgabe (z.B. Übung zu einfachen Methodenaufrufen 2)

10 Min.:

Die Schüler sollen mit Hilfe von Object-Draw die Fragestellung  
„Hat jedes Objekt die gleichen Methoden/Attribute?“  
untersuchen.

Es wird schnell klar, dass alle Rechtecke die selben Methoden/Attribute haben, während Ellipsen z.B. zusätzlich das Attribut „RadiusX“ besitzen. Dies motiviert die Definition einer Klasse, die ins Heft notiert wird. Die Schüler sollen überdies alle möglichen Klassen in Object-Draw herausfinden und notieren.

10 Min.:

Danach wird erarbeitet: Klassen können mit Hilfe von Klassendiagrammen dargestellt werden, Objekte mit Hilfe von Objektdiagrammen. Zur Festigung des Gelernten kann man z.B. ein Klassendiagramm der Klasse „SCHÜLER“ zeichnen und passend dazu ein Objektdiagramm eines bestimmten Schülers.

10 Min.: Es bietet sich an, eine weitere schriftliche Aufgabe zum Begriff „Klasse“ zu bearbeiten, z.B. „Ikarus Natur und Technik“ (Oldenbourg), Seite 27/4 oder „Informatik 1“ (Klett), Seite 14/7.

## 5. Stunde: Die Enthält-Beziehung zwischen Objekten (Gruppieren); Arbeiten mit der Zwischenablage

5 Min.: Wiederholung der Inhalte der vorausgegangenen Stunde

5 Min.: Vorstellen der Aufgabe „Auto und Straße“. Problematik: Jedes Objekt, aus dem das Auto besteht, muss einzeln verschoben werden. Gibt es eine Möglichkeit, alle diese Objekte zu einem Objekt „Auto“ zusammenzufassen?

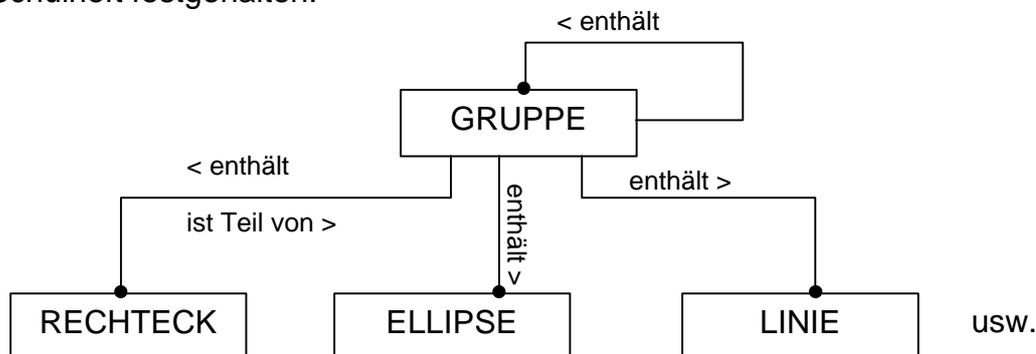
**Lösung:** Markieren aller zum Auto gehörigen Objekte und Gruppieren.

Die Lehrkraft zeigt, wie man mit Hilfe der Umschalttaste (oder durch Aufziehen eines Rahmens) mehrere Objekte gleichzeitig markieren kann und diese dann zu einem Objekt gruppiert (rechte Maustaste!).

5 Min.: Die Schüler lösen Aufgabe 1 an ihrem Rechner selbstständig.

(Für schnelle Schüler: Wie kann man die Mittelstreifen der Fahrbahn **exakt** gleichmäßig verteilen?)

10 Min.: Am Objektbaum wird sichtbar, dass ein Objekt der Klasse Gruppe andere Objekte enthalten kann. Das entsprechende Klassendiagramm und eine kurze Erklärung werden im Schulheft festgehalten:



5 Min.: Im Klassengespräch wird erarbeitet, wie man Objekte in die Zwischenablage kopiert und Duplikate in die Zeichnung einfügt

5 Min.: Die Schüler lösen Aufgabe 2 an ihrem Rechner selbstständig

5 Min.: Freie Aufgabe: Zeichne noch andere Autos und vervielfältige sie!

## 6. Stunde: Objektorientiertes Programmieren mit EOS (Einführung; die Klasse KREIS)

5 Min.: Wiederholung der Inhalte der vorausgegangenen Stunde

5 Min.:

Bisher haben wir mit Hilfe von Methodenaufrufen Objekte verändert und damit z.B. ein Grafikdokument bearbeitet. Ganz ähnlich funktionieren auch Computerprogramme: Sie bestehen - vereinfacht ausgedrückt - aus sehr vielen, hintereinandergeschriebenen Methodenaufrufen, die nacheinander abgearbeitet werden. EOS (Einfache Objektorientierte Sprache) ist ein Werkzeug, mit Hilfe dessen auch Schüler in der 6. Jahrgangsstufe schon kleine Programme schreiben können. Als Motivation sollten man den Schülern ein kleines EOS-Programm zeigen, z.B. „coole Buchstaben“ oder „Zug“.

10 Min.:

Natürlich muss man einfacher beginnen, z.B. mit Kreisen

Die Lehrkraft zeigt den Schülern anhand eines einfachen Beispiels den Umgang mit EOS, insbesondere

- Deklaration eines Objekts
- Umgang mit der Code-Vervollständigung
- Lesen von Fehlermeldungen
- Bearbeitungsmodus und Ausführungsmodus von EOS
- Starten und Anhalten des Programms

Als Beispiel eignet sich z.B.:

```
k1:KREIS
k2:KREIS

k2.füllfarbeSetzen(gelb)
k2.mittelpunktSetzen(30,20)
k2.verschieben(-20,-10)
k2.radiusSetzen(40)

k1.verschieben(-30,-10)
```

20 Min.:

Aufgabe: Schreibe ein EOS-Programm, das das nebenstehende gähnende Gesicht zeichnet!

(Für die Schnellen: zeichne ein lachendes Gesicht! Setze dem Kopf einen Hut auf!)

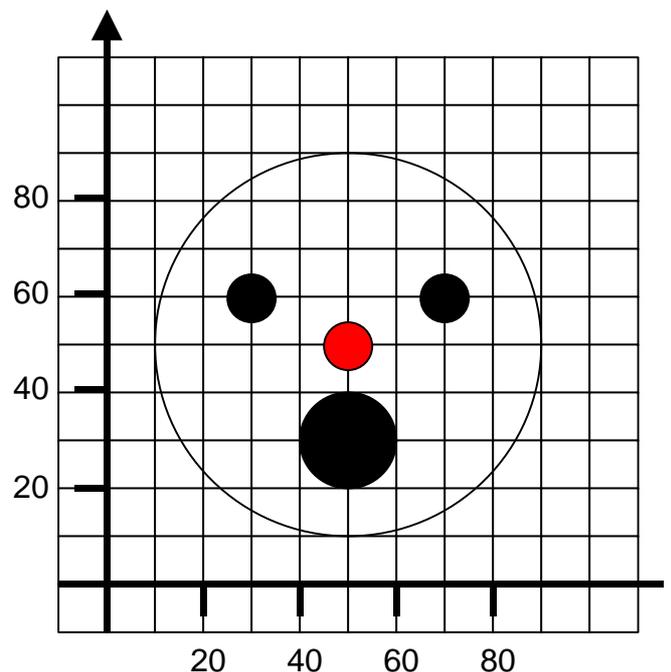
**Lösung:**

```
Gesicht, Auge1, Auge2, Nase, Mund: KREIS

Gesicht.mittelpunktSetzen(50,50)
Gesicht.füllartSetzen(durchsichtig)
Gesicht.radiusSetzen(40)

Nase.mittelpunktSetzen(50,50)
Nase.füllfarbeSetzen(rot)
nase.radiusSetzen(5)

Auge1.mittelpunktSetzen(30,60)
Auge1.radiusSetzen(5)
Auge1.füllfarbeSetzen(schwarz)
```



```
Auge2.mittelpunktSetzen(70,60)
Auge2.radiusSetzen(5)
Auge2.füllfarbeSetzen(schwarz)
```

```
Mund.mittelpunktSetzen(50,30)
Mund.radiusSetzen(10)
Mund.füllfarbeSetzen(schwarz)
```

## 7. Stunde: Objektorientiertes Programmieren mit EOS (Die Klasse TURTLE)

5 Min.:

Wiederholung der Inhalte der vorausgegangenen Stunde

10 Min.:

Vorstellung der Klasse TURTLE anhand eines einfachen Beispiels (Zeichnen eines gleichseitigen Dreiecks)

```
t:TURTLE
```

```
t.vor(50)
t.links(120)
t.vor(50)
t.links(120)
t.vor(50)
```

10 Min.:

Aufgabe: Zeichne mit Hilfe der Klasse Turtle ein 30 Punkte breites und 50 Punkte hohes Rechteck!

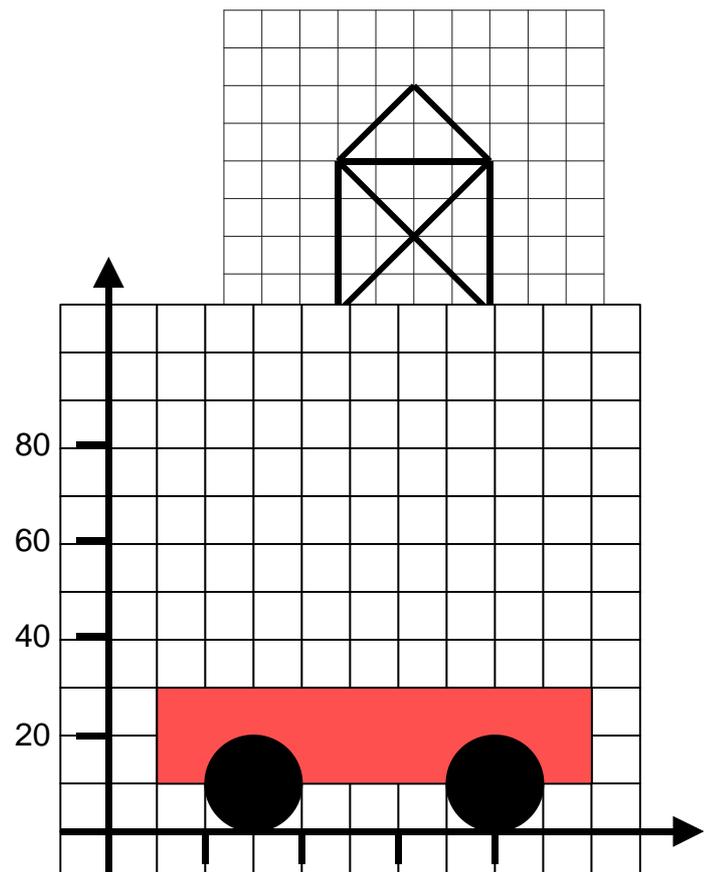
(Für die Schnellen: Zeichne daneben ein Fünfeck!)

10 Min.: Zeichne „Das Haus des Nikolaus“!  
(Tipp: die Diagonalen sind 56 Punkte lang!)

```
t:TURTLE
```

```
t.linienstärkeSetzen(3)
```

```
t.links(90)
t.vor(40)
t.rechts(90)
t.vor(40)
t.rechts(135)
t.vor(57)
t.links(135)
t.vor(40)
t.links(135)
t.vor(57)
t.rechts(90)
t.vor(28)
t.rechts(90)
t.vor(28)
t.rechts(45)
t.vor(40)
```



## 8. Stunde: Objektorientiertes Programmieren mit EOS (Die Klasse RECHTECK; evtl. Wiederholung mit fester Anzahl)

5 Min.: Wiederholung: Die Begriffe Attribut, Wert, Methode, Parameter, Klasse

10 Min.: Ziel der Stunde ist es, ein Auto zu zeichnen und fahren zu lassen. Dazu benötigen wir außer Objekten der Klasse KREIS mindestens noch solche der Klasse RECHTECK. An einem einfachen Beispiel erläutert die Lehrkraft die Verwendung dieser Klasse, insbesondere die Methode „Eckensetzen“.

10 Min.: Aufgabe: Erstelle ein EOS-Programm, das nebenstehendes Auto zeichnet!

(Für die Schnellen: Ergänzt das Auto, so dass es schicker aussieht!)

5 Min.:

Aufgabe: Ergänzt euer EOS-Programm so, dass das Auto nach rechts fährt!

Die Schüler werden Lösungen der folgenden Art finden:

```
re.verschieben(10,0)
rad1.verschieben(10,0)
rad2.verschieben(10,0)
re.verschieben(10,0)
rad1.verschieben(10,0)
rad2.verschieben(10,0)
re.verschieben(10,0)
rad1.verschieben(10,0)
rad2.verschieben(10,0)
...
```

10 Min.:

Im Klassenverband werden die Lösungsansätze besprochen. Ergebnis:

- Den Eindruck eines fahrenden Autos kann man erwecken, indem man alle beteiligten Objekte sehr oft um ein ganz kleines Stück nach rechts schieben lässt.
- Es wäre praktisch, wenn man dem Computer sagen könnte: „Wiederhole das 100-mal“

Zu diesem Zweck gibt es außer den Methodenaufrufen noch andere Anweisungen, sog. Strukturanweisungen. Sie bewirken, dass das Programm nicht Schritt für Schritt abgearbeitet wird (Sequenz), sondern z.B. dass ein bestimmter Block von Anweisungen 100-mal wiederholt wird:

```
r:RECHTECK
rad1,rad2:KREIS

rad1.mittelpunktSetzen(30,10)
rad1.radiusSetzen(10)
rad1.füllfarbeSetzen(schwarz)

rad2.mittelpunktSetzen(80,10)
rad2.radiusSetzen(10)
rad2.füllfarbeSetzen(schwarz)

r.eckenSetzen(10,30,100,10)
r.füllfarbeSetzen(rot)

wiederhole 100 mal
```

```
rad1.verschieben(1,0)
rad2.verschieben(1,0)
r.verschieben(1,0)
*wiederhole
```

Wenn noch Zeit bleibt kann man als Aufgabe stellen, das Auto solle wieder zum Ausgangspunkt zurückfahren.

**Bemerkung:** Die Strukturanweisungen sind im Lehrplan erst in Jahrgangsstufe 7 vorgesehen. Außer der Wiederholung mit fester Anzahl (die auch in der 6. JgSt schon gut von den Schülern verstanden wird und sehr motivierend ist) sollte man **auf keinen Fall** weitere Strukturanweisungen in die 6. Jahrgangsstufe vorziehen.

## 9. Stunde: Fortgeschrittenes Programmieren mit EOS (die Klassen GRUPPE)

Projekt „Coole Buchstaben“

Als Motivation kann zu Beginn der Stunde das Programm „Coole Buchstaben“ gezeigt werden.

Die Schüler entwerfen einen aus Rechtecken, Kreisen, usw. zusammengesetzten Buchstaben auf Karopapier und fertigen dann ein EOS-Programm an, das diesen Buchstaben möglichst bunt auf den Bildschirm zeichnet.

**Aufgabe 1:** Der Buchstabe soll sich um 360° drehen.

Problem dabei: Wiederholtes Drehen der einzelnen Objekte um je 1° bringt nicht den gewünschten Effekt, da sich jedes Objekt um seinen eigenen „Mittelpunkt“ dreht (siehe erster Teil des Programms „Buchstabe E“)

Lösung: Die Objekte müssen gruppiert werden. Da die Gruppe ein eigenes Objekt ist, muss sie in EOS extra instanziiert werden:

```
e: GRUPPE
Mit
e.schlucke(r1)
e.schlucke(r2)
usw.
```

werden jetzt die einzelnen Objekte der Gruppe hinzugefügt. Die Wiederholung

```
wiederhole 360 mal
    e.drehen(1)
```

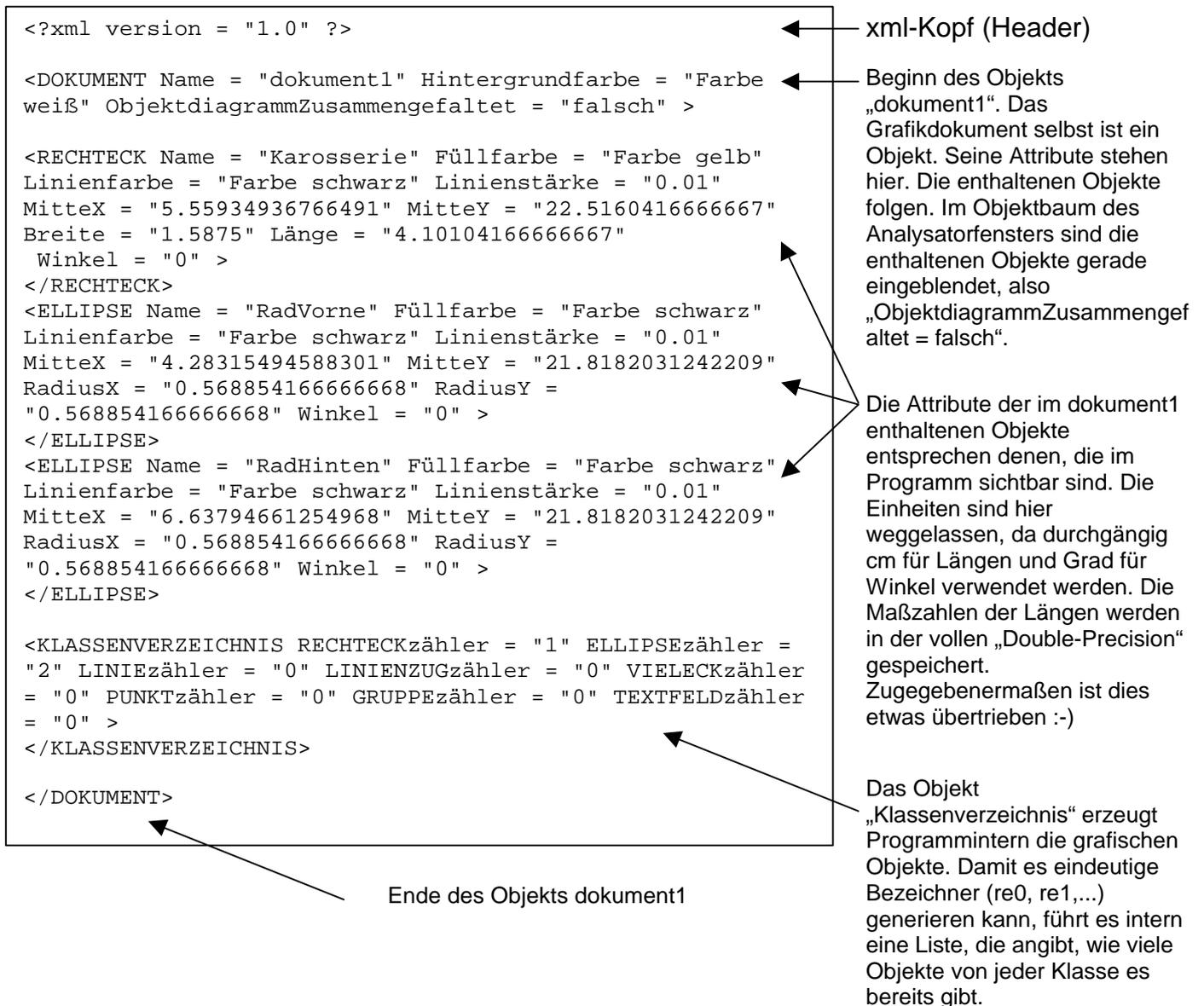
```
*wiederhole
```

leistet jetzt das Gewünschte.

**Aufgabe 2:** Bewege den Buchstaben auf verschiedene Arten (möglichst „cool“); füge noch weitere Buchstaben hinzu!

(Man kann an dieser Stelle interessierten Schülern die Methode strecken erklären ☺)

## Anhang 1: Beschreibung der xml-Struktur der .odr-Dateien am Beispiel der Datei „Sehr einfaches Auto“



**Anhang 2: „EOS-Koordinatensysteme“ zur Verwendung in Arbeitsblättern**

