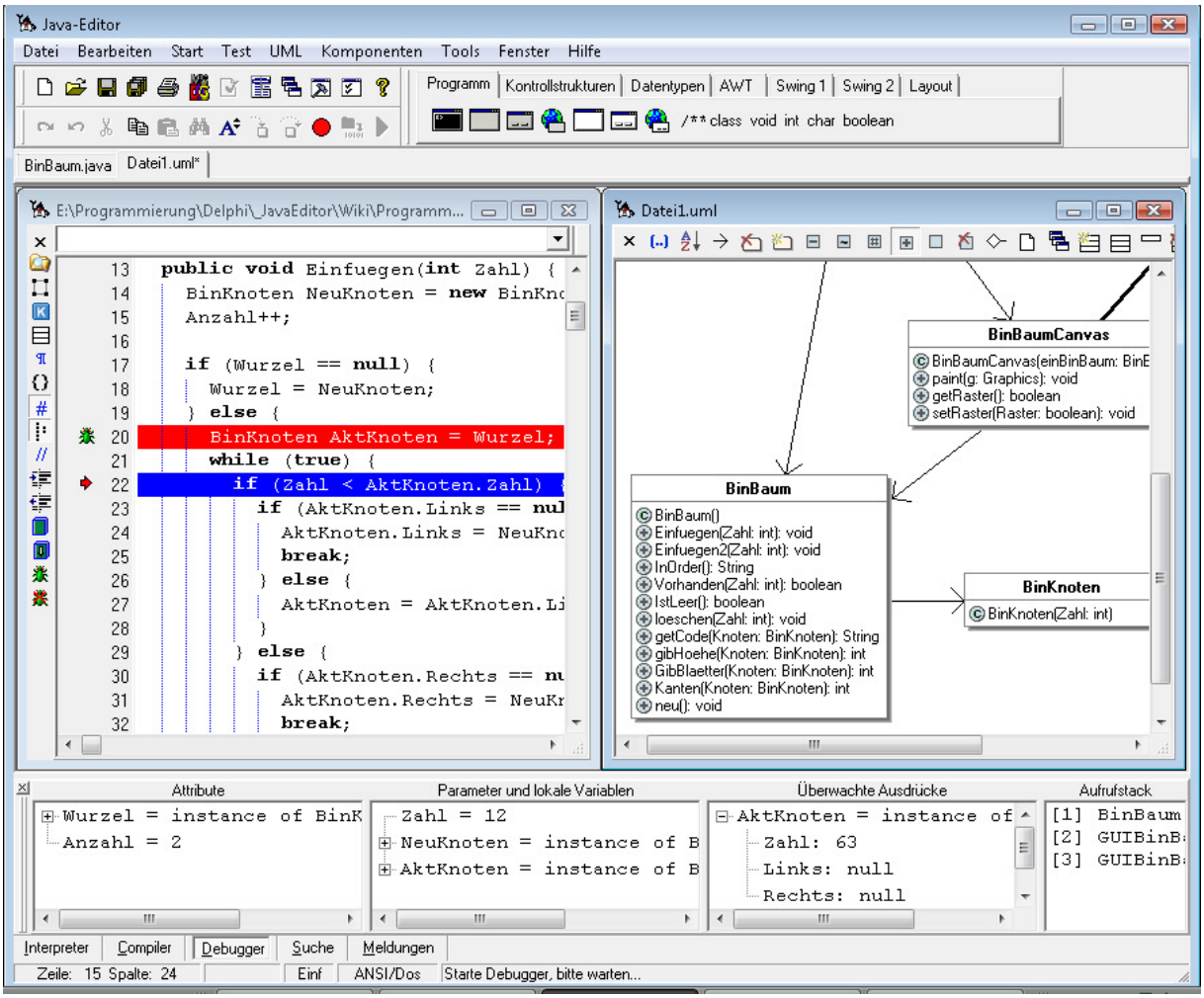
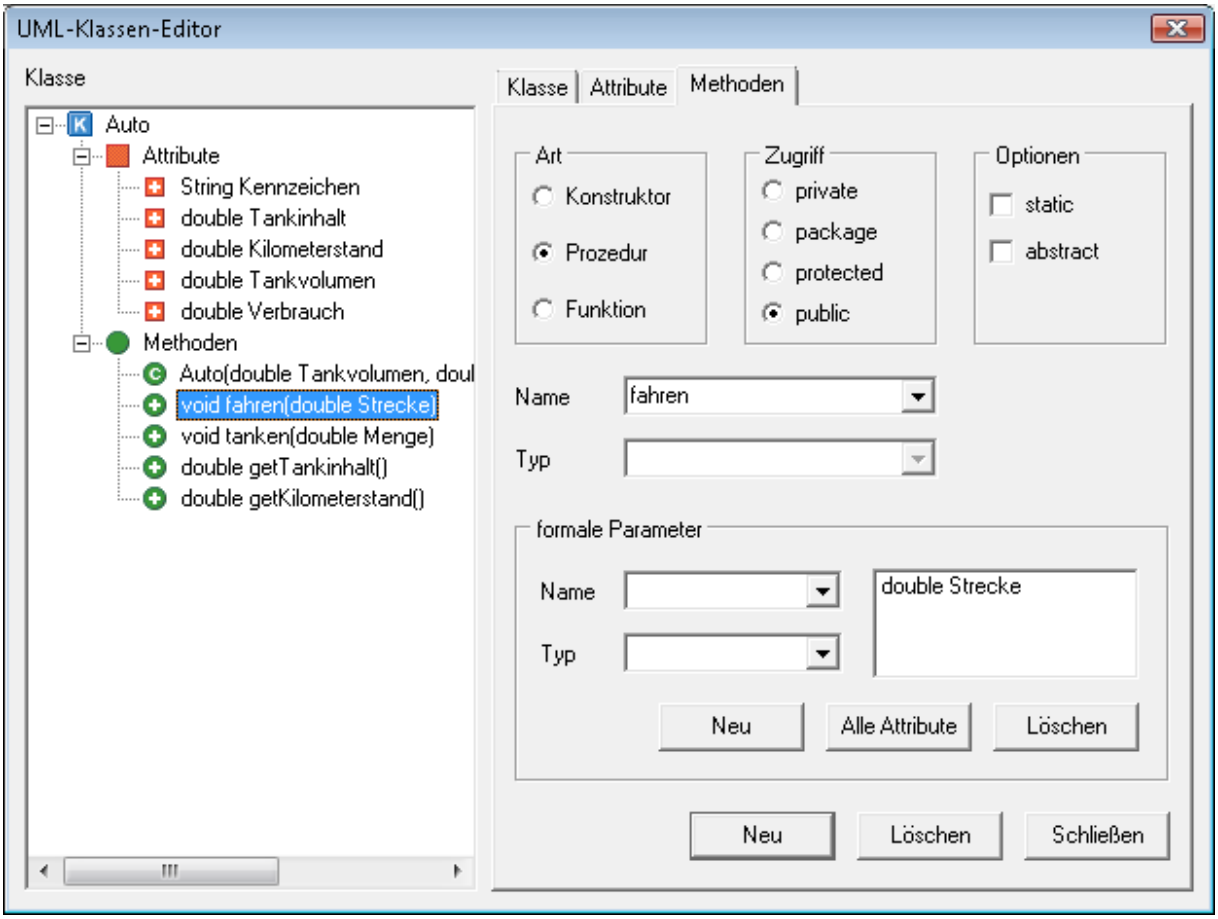
Integrierte Entwicklungsumgebungen

Integrierte Java-Entwicklungsumgebungen setzen hohe Anforderungen an die Ausstattung der Computer. Diese Anforderungen sind in der Schule nicht immer umzusetzen. Sollen die Schüler auch zu Hause mit diesen Entwicklungssystemen arbeiten, so kommen eigentlich nur frei verfügbare Systeme in Frage. Da bleiben kaum noch Systeme übrig, die in der Schule tatsächlich einsetzbar sind. Dies war für mich Anlass, den nachfolgend beschriebenen Java-Editor zu entwickeln.
Im Bild sehen Sie das Programmfenster des Java-Editors mit Menü- und Symbolleiste, sowie einigen Registern mit Hilfen zur Programmentwicklung. Die Ausstattung ist an den Notwendigkeiten der Schule ausgerichtet. Daher gibt es zum Beispiel das Register Programm mit den Programm-Grundstrukturen für Konsolen-Anwendungen, Applikationen und Applets und das Register Kontrollstrukturen mit den Java-Kontrollstrukturen.



Klassen-Modellierer

Mit dem Klassen-Modellierer können Sie neue Klassen erzeugen und bestehende Klassen bearbeiten. Er liest Java-Dateien und stellt sie als Klasse mit Attributen und Methoden dar. Änderungen im Klassen-Modellierer werden direkt in die zugehörige Java-Datei übernommen.
Der Klassen-Modellierer ist ein einfach benutzbares Werkzeug zur Modellierung von Klassen.



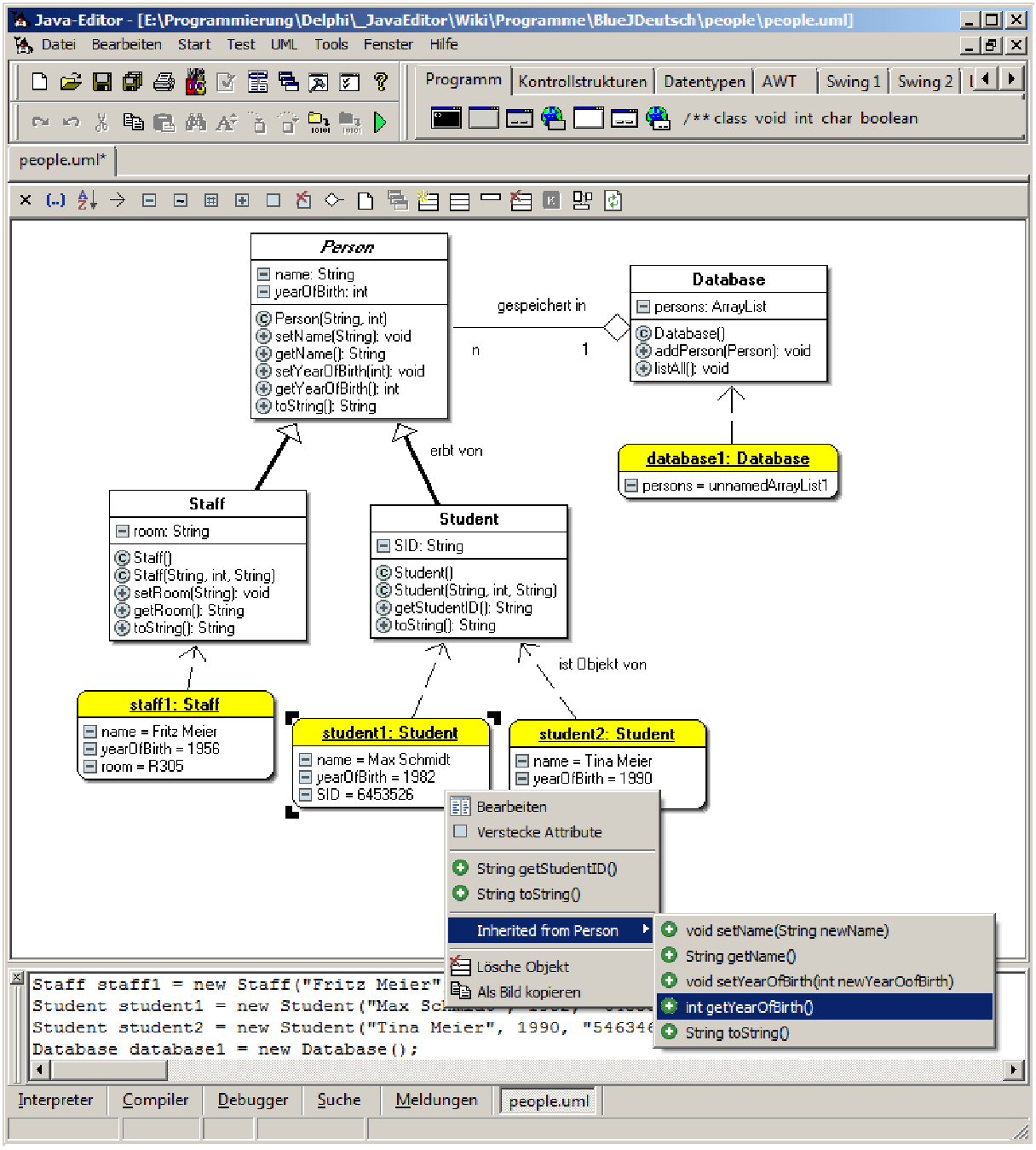
Die Reihenfolge der Attribute und Methoden im Strukturbaum kann mittels Drag&Drop verändert werden.
Den Klassen Modellierer rufen Sie über das Symbol ☰ im Editorfenster auf. Sie können auch im UML-Fenster die gewünschte Klasse doppelklicken, um den Klassen-Modellierer aufzurufen.

UML-Fenster

Nach dem Modellieren einer Klasse sollte man das Modell zu testen. Im UML-Fenster können Sie durch Aufrufen des Kontextmenüs einer Klasse mit der rechten Maustaste einen Konstruktor zum interaktiven Erzeugen eines Objekts aufrufen. Es wird dann ein Objekt dieser Klasse erzeugt.

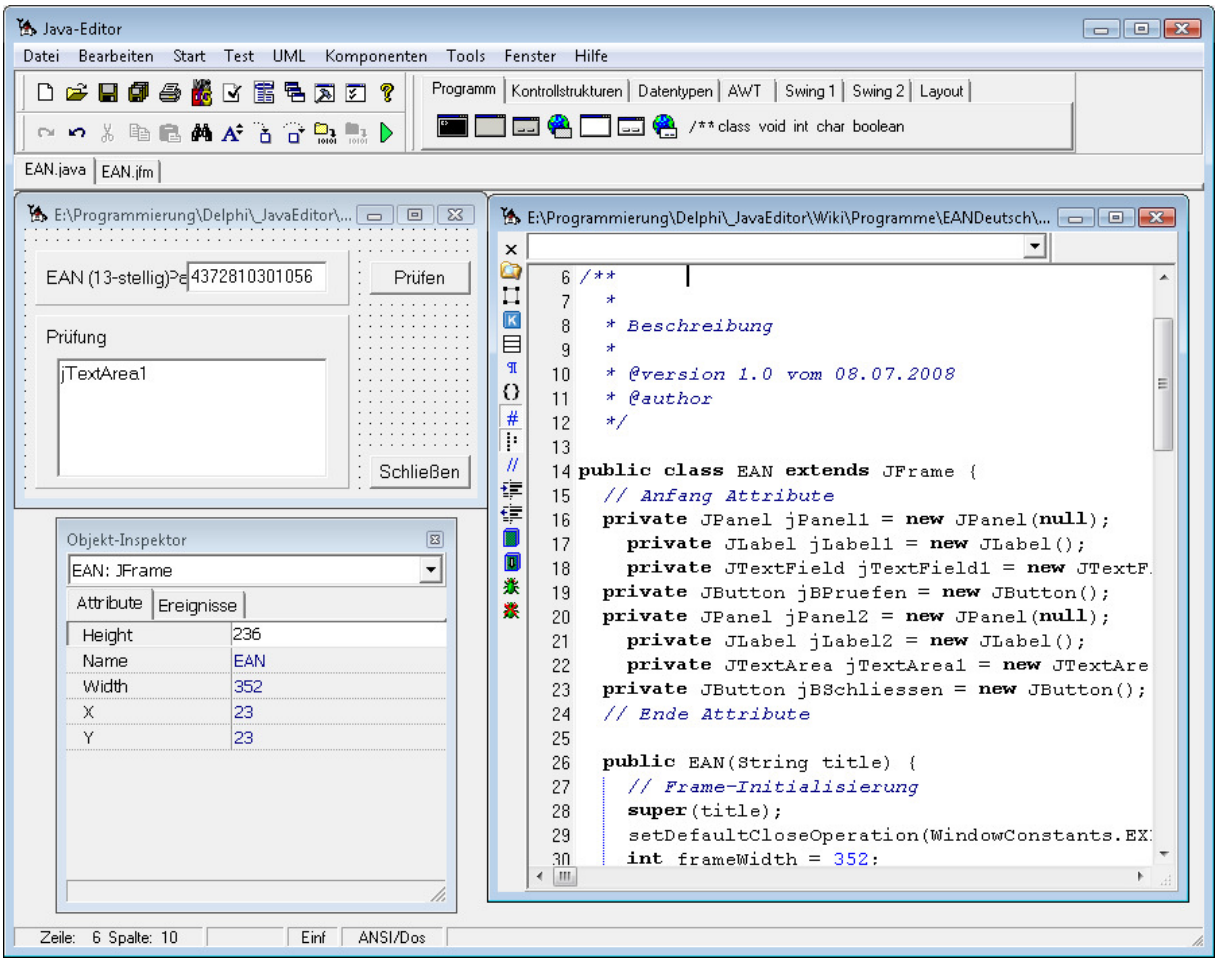
Rufen Sie anschließend das Kontextmenü des Objekts auf, so können Sie die Attribute des Objekts editieren und Methoden des Objekts aufrufen.

Die interaktive Objekterzeugung orientiert sich an der entsprechenden BlueJ-Funktionalität. Es wird keine *main*-Methode benötigt, modellierte Klassen können also interaktiv getestet werden.



GUI-Designer

Nach dem Testen eines Modells im UML-Fenster kann man es in einem Konsolen- oder GUI-Programm benutzen.
Es ist ziemlich kompliziert eine graphische Benutzungsoberfläche mit Layout-Managern zu gestalten. Schülerinnen und Schüler haben damit erhebliche Schwierigkeiten. Viel einfacher ist es mittels Drag&Drop GUI-Komponenten an die gewünschte Position auf dem Formular zu ziehen. Der im Java-Editor verfügbare GUI-Designer benutzt daher keine Layout-Manager sondern absolute Positionierung, womit man ziemlich schnell sein Layout gestalten kann, und das ist genau das, was in der Schule gebraucht wird.



Zum Anlegen eines GUI-Formulars klicken Sie auf der Registerkarte *Programm* auf eines der sechs Symbole: Frame, Dialog, Applet, JFrame, JDialog oder JApplet.
Zum Platzieren einer GUI-Komponente klickt man zunächst auf das entsprechende Symbol der Registerkarten AWT, Swing1 bzw. Swing2 und dann auf das GUI-Formular. Anschließend kann man die Komponente auf dem GUI-Formular

positionieren und mit dem Objekt-Inspektor konfigurieren. Im Quelltext werden die jeweiligen Einstellungen synchron vorgenommen. Dazu werden die Abschnitte

```
// Anfang Attribute
// Ende Attribute
```

und

```
// Anfang Komponenten
// Ende Komponenten
```

benutzt. Diese Abschnittskennzeichnungen und der darin befindliche, automatisch erzeugte Quelltext darf nur auf eigene Gefahr geändert werden.
Über das Symbol  eines Quelltext-Fensters können Sie das zugehörige GUI-Formular öffnen und zusammen mit dem Quelltext-Fenster anordnen.
Das GUI-Formular verfügt über ein Kontextmenü zur Bearbeitung der Komponenten. In anderen Entwicklungssystemen wird die Struktur der grafischen Benutzungsoberfläche als Baum dargestellt. Der Java-Editor verzichtet auf ein weiteres Fenster und zeigt die Struktur im Quelltext durch entsprechendes Einrücken der zugehörigen Variablen an.

Menü-System

Datei-Menü

Das Datei-Menü stellt die üblichen Operationen wie *Neu*, *Öffnen*, *Speichern* und *Drucken* zur Verfügung. Alle geöffneten Fenster können als Java-Editor-Projekt (.jep-Datei) gespeichert werden. *Alle speichern in* speichert alle Fenster in einem anderen Ordner. Eine Java-Datei können Sie im HTML und RTF-Format exportieren, um die farbige Syntaxauszeichnung außerhalb des Editors verwenden zu können.

Editor-Menü

Der Editor unterstützt die Arbeit mit der Windows-Zwischenablage, mehrstufiges Rückgängig machen von Editoroperationen und das Suchen und Ersetzen von Texten. Markierter Text kann in unterschiedlichen Formaten kopiert werden. Suchen und Ersetzen ist in Dateien und mit regulären Ausdrücken möglich. Sie können markierten Texte einrücken (Strg+Umsch+I), ausrücken (Strg+Umsch+U) sowie ein- und auskommentieren (Strg+K). Zur einfachen Verwendung der Konsolenausgabe

fügt Strg+U *System.out.println()* in den Quellcode ein, Strg+Y löscht eine Zeile. Sie können bis zu zehn Lesezeichen mit der Maus oder der Tastatur (Strg+Umsch+#) setzen und zu einem Lesezeichen springen (Strg+#). Jeder Befehl des Editor-Menüs hat ein Taststenkürzel und ein Symbol auf der Symbolleiste. Der Editor basiert auf der SynEdit Komponente. In der Konfiguration kann man sich eine Liste aller Tastenkürzel anzeigen lassen.

Start-Menü

Über das Start-Menü bzw. die zugehörigen Symbole kann das im aktiven Editor-Fenster stehende Programm compiliert werden. Wenn Sie den *jikes*-Compiler installieren haben Sie einen zweiten Compiler zur Verfügung. Fehlermeldungen werden im Compiler-Fenster angezeigt. Per Doppelklick auf eine Fehlermeldung wird der Cursor an die Fehlerstelle im Quelltext positioniert. Falls die Fehlermeldung unverständlich ist, ist es eine gute Idee mit den anderen Compiler zu verwenden. Manchmal sind dessen Fehlermeldungen besser verständlich.
Wenn Sie ein Programm *starten*, das zuvor geändert wurde, wird es automatisch gespeichert, compiliert und falls fehlerfrei gestartet.
Applets werden im Appletviewer angezeigt. Die zum Aufruf nötigen HTML-Dokumente werden automatisch oder manuell per Befehl generiert. Startet Sie das zum Applet gehörige HTML-Dokument, so wird das Applet im Browser angezeigt.
Zusätzlich lassen sich über das Start-Menü der Debugger, Disassembler und JavaDoc-Dokumentierer aufrufen. Mit dem Jar-Untermenü können Sie jar-Dateien ausführbare jar-Dateien erzeugen. Der Pack-Befehl unterstützt den einfachen Austausch von Dateien zwischen Schülern und Lehrern. In der Konfiguration wird festgelegt, welche Dateien gepackt werden. Eine gepackte jar-Datei kann leicht auf eine Lernplattform wie z. B. Moodle hochgeladen oder per E-Mail an den Lehrer geschickt werden.

Test-Menü

Mit den Befehlen des Test-Menüs führen Sie ein Programm unter der Kontrolle des Debuggers aus. Wenn ein Haltepunkt in einer Java-Datei gesetzt ist und Sie ihr Programm starten, wird der Debugger benutzt. Nach dem Erreichen eines Haltepunktes wird die Ausführung unterbrochen. Sie können dann ihr Programm

schrittweise ausführen. Während des Debuggens zeigt das Meldungsfenster detaillierte Informationen über Attribute, Parameter, lokale Variable, überwachte Ausdrücke und den Aufrufstack.
Um ein GUI-Programm zu debuggen setzen Sie einen Haltepunkt an die gewünschte Stelle. Starten Sie ihr GUI-Programm und warten Sie, bis das GUI-Formular gezeigt wird. Klicken Sie dann den Schalter an, der zum Aufruf der Stelle mit dem Haltepunkt führt.

UML-Menü

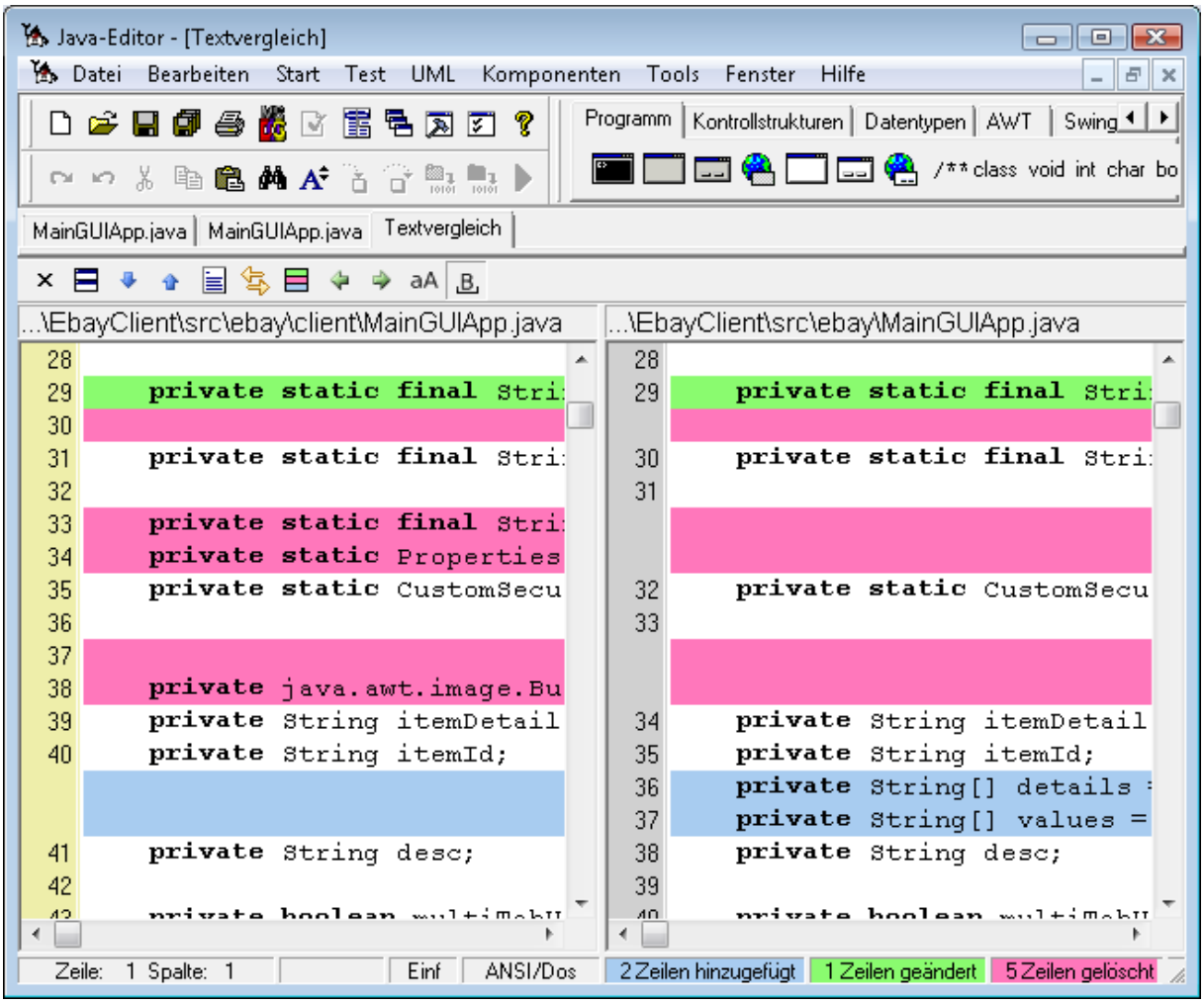
Mit dem UML-Menü können Sie neue UML-Fenster öffnen, neue Klassen anlegen und Klassen öffnen und bearbeiten. Zum Speichern und Laden von Klassendiagrammen benutzen Sie die normalen Öffnen und Speichern-Symbole und wählen dabei den Dateityp '.uml' aus. Sie können ein Klassendiagramm als Bild speichern oder in die Zwischenablage zur direkten Weiterverarbeitung kopieren. Mit dem Befehl 'Ordner öffnen' können Sie aus allen Java-Dateien eines Ordners sowie seiner Unterordner ein Klassendiagramm erstellen. Wenn Sie schon ihre Java-Dateien geöffnet haben benutzen Sie einfach *Diagramm aus Dateien* um ein Klassendiagramm zu erzeugen.
Das UML-Fenster bietet eine reichhaltige Symbolleiste zur Bearbeitung des Klassendiagramms an. Klassenbeziehungen werden automatisch erkannt, sofern entsprechende Attribute vorhanden sind. Anderenfalls können Sie Klassen manuell miteinander verbinden. Markieren Sie dazu die Startklasse und ziehen Sie dann mit der Maus eine Verbindung zur Zielklasse.

Komponenten-Menü

Standardmäßig wird das Komponenten-Menü nicht angezeigt. Es steht für sehbehinderte oder blinde Benutzer zur Verfügung, die keine Maus benutzen können. Das Komponenten-Menü ermöglicht die Erstellung von GUI-Oberflächen ohne den GUI-Designer.

Tools-Menü

Das Tools-Menü bietet das *Checkstyle*-tool an, das die Einhaltung eines Codierstandards für einen Java-Quelltext prüft. Das *Jalopy*-tool formatiert einen Java-Quelltext gemäß einem vorgegebenen Codierstandard. Das Tool zum Vergleichen von Dateien zeigt neue, geänderte und gelöschte Zeilen. Änderungen können in die jeweils andere Datei übernommen werden. Wenn Sie Versionsverwaltung Subversion installieren, erscheint ein eigenes Subversion-Untermenü.



Fenster-Menü

Neben den Grundfunktionen zum Anordnen, Öffnen und Schließen von Fenstern können Sie Symbolleisten anzeigen oder verstecken, Konsolen-, Explorer- und Browser-Fenster öffnen. Zusätzlich können Sie für das aktive Fenster die Schriftart wählen und den Konfigurationsdialog aufrufen.

Hilfe-Menü

Das Hilfe-Menü gibt Zugriff auf die Java-Dokumentation, Demos und das Tutorial. Sie können das Javabuch in HTML-Form integrieren, haben einen Verweis auf die Java-Editor Homepage und können den Java-Editor updaten.

Betriebssysteme

Windows

Der Java-Editor wird unter Windows entwickelt.

Linux

Der Java-Editor funktioniert unter Linux mit der *Wine*-Erweiterung. Hinweise zur Installation finden Sie auf der WineHQ Seite.

Mac

Der Java-Editor läuft auf dem Mac sowohl mit CrossOver Office (Wine) als auch in einer virtualisierten Windows XP-Instanz problemlos.
Bei einem 64-Bit Mac funktioniert er unter der Virtualisierungssoftware VMWare Fusion einwandfrei. Die Alternativen Boot Camp, VirtualBox, CrossOver und Parallels scheinen dies nicht zu ermöglichen.

Barrierefreiheit

Der Java-Editor hat spezielle Unterstützung für Sehbehinderte und Blinde. Im Optionen-Menü kann die Schriftgröße für das Komponenten-Menü eingestellt werden. Die Schriftgröße für den Editor und weitere Fenster stellt man über das Symbol Schriftgröße oder das Schriftarten-Menü ein.
Wenn nicht mit der Maus gearbeitet werden kann, so schaltet man die Darstellung der Symboleiste im Fenster-Menü ab und aktiviert in der Konfguration bei den Optionen die Option *Komponentenmenü anzeigen*. Mit dem Komponenten-Menü kann man GUI-Formulare ohne Mausbenutzung erstellen.

Einführung in den Java-Editor mit Applets

Stefan Bartels hat freundlicherweise eine Einführung in den Java-Editor mit Applets (http://www.javaeditor.org/images/f/fe/Einfuehrung_Java-Editor.pdf) bereitgestellt