

Designbeschreibung der Applikation: "GeoViewer"

1. Allgemeines

1.1. Kurzcharakterisierung

GeoViewer ist eine menügesteuerte graphische Java-Applikation, mit der sich geometrische Konfigurationen visualisieren lassen, die in Form von GEO-Records vorliegen. Die algebraischen Rechnungen zur Bestimmung der Koordinaten der geometrischen Objekte erfolgt mit dem GeoProver-Paket und dem Computeralgebrasystem Maple.

1.2. Systemvoraussetzungen

GeoViewer ist als eigenständige Applikation konzipiert, die über eine grafische Nutzerschnittstelle mit gängigen Fenster-Techniken bedient wird.

GeoViewer benötigt als Voraussetzungen zum Starten die Java JRE 1.4., Maple (ab Version 5), das GeoProver-Paket für Maple sowie GEO-Records zur Eingabe. Die genaue Lage der einzelnen Ressourcen kann in einer Datei GeoViewer.rc spezifiziert werden.

1.3. Beschreibung der Produktumgebung

GEO-Records enthalten Beschreibungen geometrischer Beweisschemata, die in einem speziellen XML-artigen Format abgelegt sind, das auf der GeoCode-Spezifikation aufsetzt. Beides ist näher in der Dokumentation des SymbolicData-Projekts (<http://www.symbolicdata.org>) beschrieben.

Ein GEO-Record enthält neben der Beschreibung der geometrischen Konfiguration eine Liste von unabhängigen Parametern, die für die konkrete Visualisierung mit geeigneten Zahlenwerten zu belegen sind. Die Parameterwerte haben Einfluss auf die Lage einzelner geometrischer Objekte (freier Punkte und Gleiter) und damit auch auf die Lage abgeleiteter Objekte. Verschiedene Parameterwerte ergeben also verschiedene Bilder derselben geometrischen Konfiguration. In diesem Sinne kann die Visualisierung „dynamisiert“ werden.

Zur Interpretation dieser Beweisschemata und Behandlung der algebraischen Fragen wird eine Implementierung des GeoCode-Standards im GeoProver-Paket für das Computeralgebrasystem Maple herangezogen.

1.4. Abgrenzung

Ein GEO-Record kann auch abhängige Parameter enthalten, deren Werte sich als Lösungen eines Gleichungssystems ergeben, dessen Koeffizienten mit den unabhängigen Parametern variieren. Da die Behandlung von (nichtlinearen) Gleichungssystemen mit variierenden reellen Koeffizienten auch für Maple schwierig ist, werden derartige GEO-Records (vom Gleichungstyp) nicht visualisiert.

GEORecords ohne abhängige Parameter bezeichnet man als Records vom konstruktiven Typ.

Eine „Dynamisierung“ der Visualisierungen durch direkte Mausmanipulation ist nicht implementiert, da hierfür weitergehendes Event-Handling erforderlich ist. Parameterwerte einer Visualisierung können nur über ein Dialogfeld geändert werden.

Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

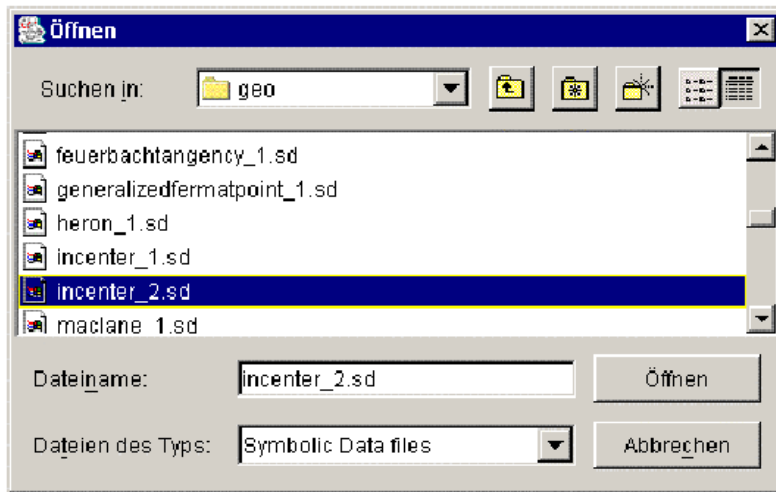
2. Produktübersicht

Die Applikation wird durch den Kommandozeilen-Aufruf

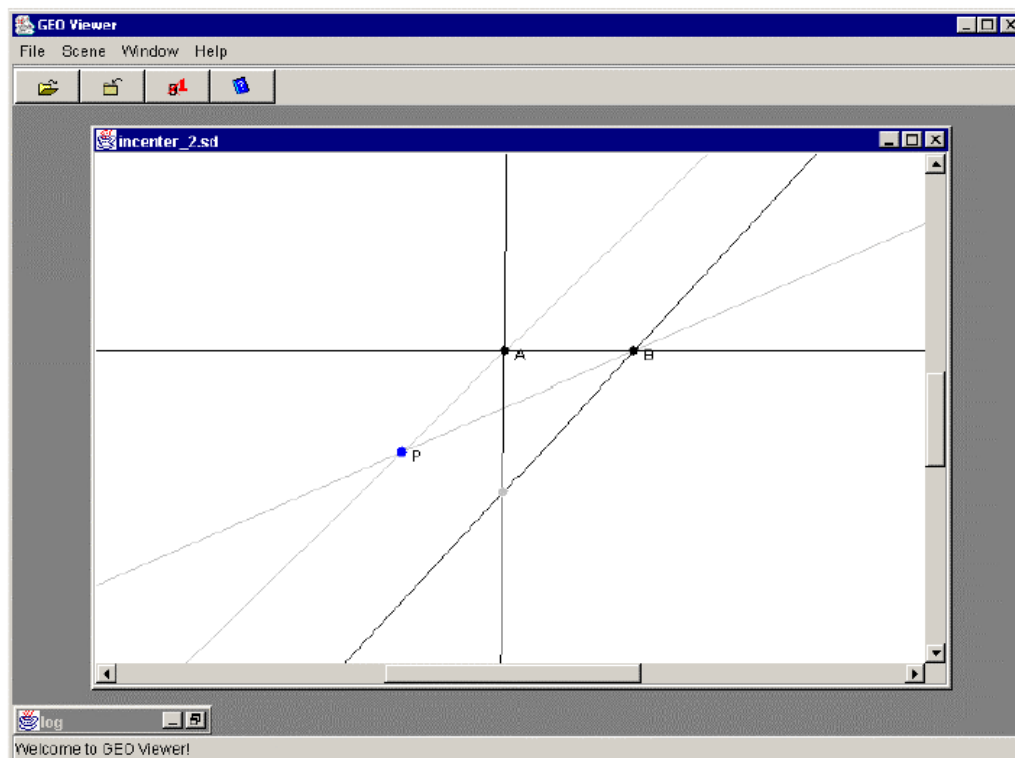
```
java geometry.GeoViewer
```

im Wurzelverzeichnis der Pakethierarchie gestartet. Während der *Initialisierung* wird die Verbindung zu Maple hergestellt und im Erfolgsfall das Hauptfenster aufgebaut. Das *Hauptfenster* enthält am oberen Rand einen Menü- und Toolbalken, am unteren Rand einen Statusbalken und in der Mitte eine Desktop-Fläche, in der im weiteren Verlauf die verschiedenen Zeichenflächen sowie ein spezielles Fenster für Fehlermeldungen geöffnet werden können.

Über den Menüeintrag *File* oder das linke Icon wird das *Laden eines GEO-Records* gestartet.

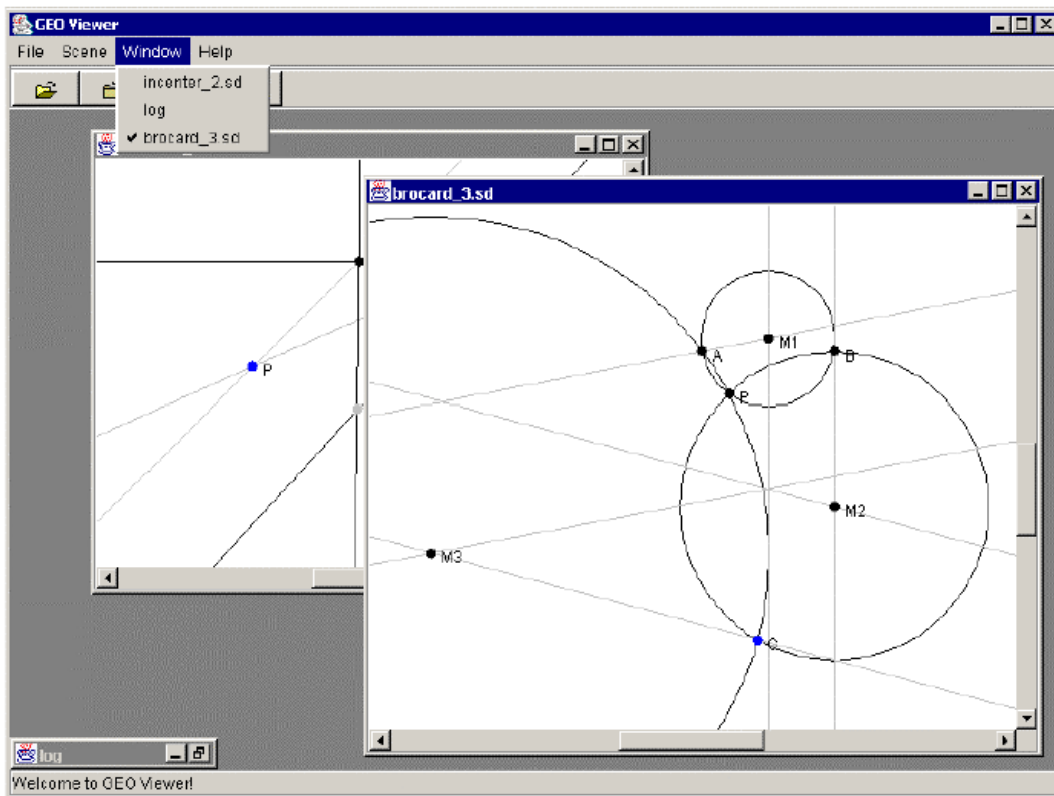


Handelt es sich um ein konstruktives Schema, so wird die zugehörige Visualisierung generiert und in einer neuen Zeichenfläche im Desktopbereich angezeigt und aktiviert. Die Parameterwerte werden dabei zunächst mit Zufallszahlen initialisiert.

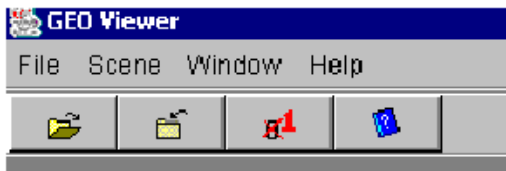


Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

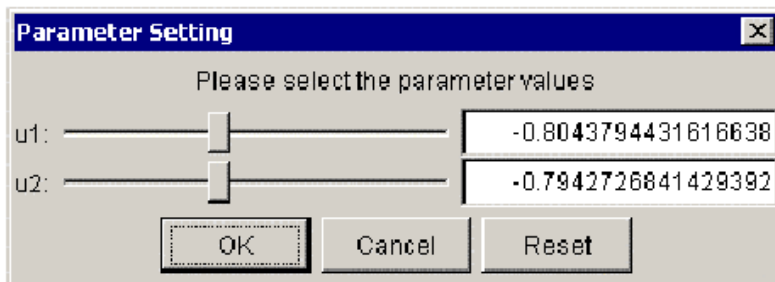
Zwischen verschiedenen Fenstern der Desktopfläche kann mit der Maus hin- und hergeschaltet werden. Alternativ kann das aktuelle Fenster über ein RadioButton-Menü eingestellt werden.



Wird als Fenster eine Zeichenfläche ausgewählt (aktive Zeichenfläche), so werden für diese die Menüpunkte und Icons „Parameterwerte ändern“ und „Zeichenfläche schließen“ aktiviert.



Die Parameterwerte können für die aktive Zeichenfläche über einen Menüeintrag geändert werden, der ein Dialogfenster mit Schieberegler öffnet. Danach werden die Koordinaten der einzelnen Objekte von Maple neu berechnet und die Visualisierung in der aktiven Zeichenfläche neu dargestellt.



Über einen weiteren Menüpunkt, Icon oder einen speziellen Knopf am Fensterrand wird die aktive Zeichenfläche geschlossen.
Die Applikation wird über den Exit-Eintrag im File -Menü beendet.

Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

3. Grundsätzliche Designentscheidungen

GeoViewer:

ist die Hauptklasse des Programms. In ihr werden die Grundeinstellungen für das Programm aus der Datei GeoViewer.rc geladen.

Parser

GeoParser:

Leitet die Klasse Parser aus dem Paket Java_Cup ab. Die Funktion prüft ob die vom GeoScanner eingelesene Datei syntaktisch korrekt ist. Diese Funktion wird mit Hilfe eines Stacks realisiert. Dabei werden nach einander die Starttags auf den Stack verschoben und beim lesen des zugehörigen Endtags wieder entfernt.

GeoScanner:

Liest eine Datei ein .
Bei einer leeren Datei wird ein Exception ausgelöst, sonst kann mit der Funktion next_token ein Zeichen aus der Datei gelesen werden.
Die Sonderzeichen werden in der Klasse Geosymbols definiert.

GeoSymbols:

Definition von Sonderzeichen z.B. END_POINTS, BEGIN_PARAMETERS

Translator:

Stellt Funktionen bereit, die vom Parser genutzt werden, um aus der SD-Datei die Objekte der Scene zu rekonstruieren.

TranslatorException:

Gibt eine detaillierte Fehlermeldung aus, falls es in der Klasse Translator zu einem Ausnahmezustand kommt.

CAS

CASInterface:

Ist eine Schnittstelle zur CAS für die numerische Auswertung von symbolischen Ausdrücken. Der Ausdruck wird zur CAS übermittelt. CAS wertet diesen Ausdruck aus und gibt einen Wert zurück, dieser wird dann als Wert des Ausdrucks gesetzt. Fehler bei der Auswertung werden durch CASExceptions und IOExceptions signalisiert.

MapleInterface:

Diese Klasse implementiert das CASInterface und realisiert die Kommunikation mit der Maple-Konsole.

MupadInterface:

Diese Klasse implementiert auch das CASInterface und soll eine Kommunikation mit Mupad erlauben. Die Funktionen dieser Klasse sind aber noch nicht vollständig implementiert.

CASException:

ist eine IOException und wird ausgelöst wenn Ausnahmen in der CAS auftreten.

CASDivisionByZeroException:

ist eine CASException die bei Division durch 0 ausgelöst wird.

Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

Controller

Expression:

Von dieser abstrakten Klasse werden alle weiteren Ausdrücke abgeleitet.
Ein Objekt dieser Klasse besteht aus einer Bezeichnung, einem Wert und eventuell einem Namen.
Die Bezeichnung ist der Ausdruck der in der CAS ausgewertet wird.

Function:

Die Klasse wird von Expression abgeleitet. Es wird der Name der Funktion oder der Operator gespeichert. Von dieser Klasse werden die Geo-Funktionen und Algebraische Operationen abgeleitet.

Constant:

Die Klasse wird von Expression abgeleitet. Objekte dieser Klasse repräsentieren konstante Ausdrücke.

Parameter:

Die Klasse wird von Expression abgeleitet und modelliert Parameter. Ein Parameter muss immer den Namen behalten unter dem er deklariert wurde.

Variable:

Die Klasse wird von Expression abgeleitet und modelliert Variablen.

Operation:

Diese Klasse wird von Function abgeleitet und modelliert algebraische Operationen.

GeoElement:

Ist die abstrakte Oberklasse aller geometrischen Objekte und implementiert die Schnittstelle Comparable. Sie besteht aus Koordinaten und einer Priorität. Die Vergleichsfunktion ordnet die Objekte nach aufsteigender Priorität.

GeoPoint, GeoDistance, GeoAngle, GeoLine, GeoCircle:

Diese Klassen sind Unterklassen von GeoElement. Sie repräsentieren Punkte, Abstände, Winkel, Linien und Kreise. Jedes Objekt dieser Klassen hat eine feste Priorität.

Punkte haben Priorität 6

Abstände Priorität 5

Winkel Priorität 4

Linien Priorität 3

Kreise Priorität 2

SceneController:

Die Klasse stellt Funktionen zur Kontrolle der Szene zur Verfügung. Objekte der Klasse bestehen aus einem CASInterface, einem Namen und Feldern für Expressions, GeoElemente und Parameter.

Es wird eine Funktion zum laden von Dateien im SymbolicData-Format zur Verfügung gestellt. Diese Funktion greift auf das Paket Parser zu.

Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

View

ApplikationFrame:

Diese Klasse ist aus der Swingklasse JFrame abgeleitet. Sie erzeugt das Startfenster und liefert die zugehörige Funktionalität.

OpenAction, ExitAction, ParameterAction, CloseAction, AboutAction:

Diese Klassen sind von AbstractAction abgeleitet und implementieren die Funktionen der zugehörigen Buttons/Menüpunkte. OpenAction erstellt dabei ein FileSelectionField. ParameterAction erstellt dabei eine ParameterBox.

DesktopListener, DesktopEvent, DesktopAdapter:

Beim DesktopEvent handelt es sich um ein ObjektEvent. Es werden Events ausgelöst wenn der aktuelle Frame deaktiviert, ein neuer Frame hinzugefügt oder der aktuelle Frame geschlossen wird. Diese Events werden von Klassen aufgefangen die den DesktopListener implementieren, der DesktopAdapter (aus Applikationframe und aus Windowmenü) implementiert die Schnittstelle.

FileSelectionField:

Erstellt ein Fenster zur Auswahl einer Datei. Der User kann den Dateinamen direkt in ein Textfeld eingeben oder er klickt auf eine Schaltfläche und es öffnet sich ein JFileChooser in dem man eine Datei auswählen kann. Die Klassen FileFieldAdapter und FileButtonAdapter implementieren die zugehörige Funktionalität.

ExtensionFileFilter:

Es werden nur Dateien mit einer bestimmten Endung im Fenster angezeigt.

GeoToolBar:

Ist eine JToolBar. Fügt man einen Eintrag hinzu wird, das in der Action festgelegte Bild verwendet und der Eintrag als Button eingefügt. Die SHORT_DESCRIPTION der Action wird als Tool-Tip-Text verwendet.

Statusbar:

Wird aus der Swingklasse JPanel abgeleitet. Es handelt sich um eine Statuszeile mit einem Standardtext, in der ein übergebener Text angezeigt wird.

LogFrame LogDeamon:

Ist von JInternalFrame abgeleitet. In dem Frame werden alle Fehlermeldungen ausgegeben. Das Umleiten der Meldungen wird im Hintergrund mit dem Thread LogDeamon realisiert.

GeoDesktopFrame:

Ist ein JDesktopFrame mit zusätzlichen Funktionen. Es enthält eine Liste mit allen offenen Frames und einen GeoDesktopManager.

GeoDesktopManager:

Aktivieren, deaktivieren schließen oder minimieren von Frames.

AboutBox:

Stellt einen About-Dialog mit zugehöriger Funktionalität zur Verfügung. Die Klasse wurde von JDialog abgeleitet. Die Klasse wird von AboutAction benötigt.

Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

Parameterbox:

Hierbei handelt es sich um eine Dialogbox zum verändern der Parameter eines Objektes. Die Parameter können hier mit JSlidern verändert werden.

SceneFrame, ScenePane:

Bei SceneFrame handelt es sich um eine Ableitung der Klasse JInternalFrame. In diesen Frame befindet sich ein ScenePane in den die geometrischen Objekte mit Hilfe der Klasse ElementGraphics gezeichnet werden. Bei Fehlern wird eine ElementGraphicsException geworfen.

ElementGraphics, ElementGraphicsException:

Für jedes GeoElement muss zum zeichnen ein ElementGraphics erstellt werden. Dieses zeichnet bei Aufruf der Funktion draw das entsprechende Objekt. Dabei wird für jedes Geoelement die entsprechende drawmethode aufgerufen die in dieser Klasse definiert ist. Die Koordinaten werden dabei mit AffineTransform umgerechnet. Kann das Objekt nicht gezeichnet werden wird eine ElementGraphicsException ausgelöst.

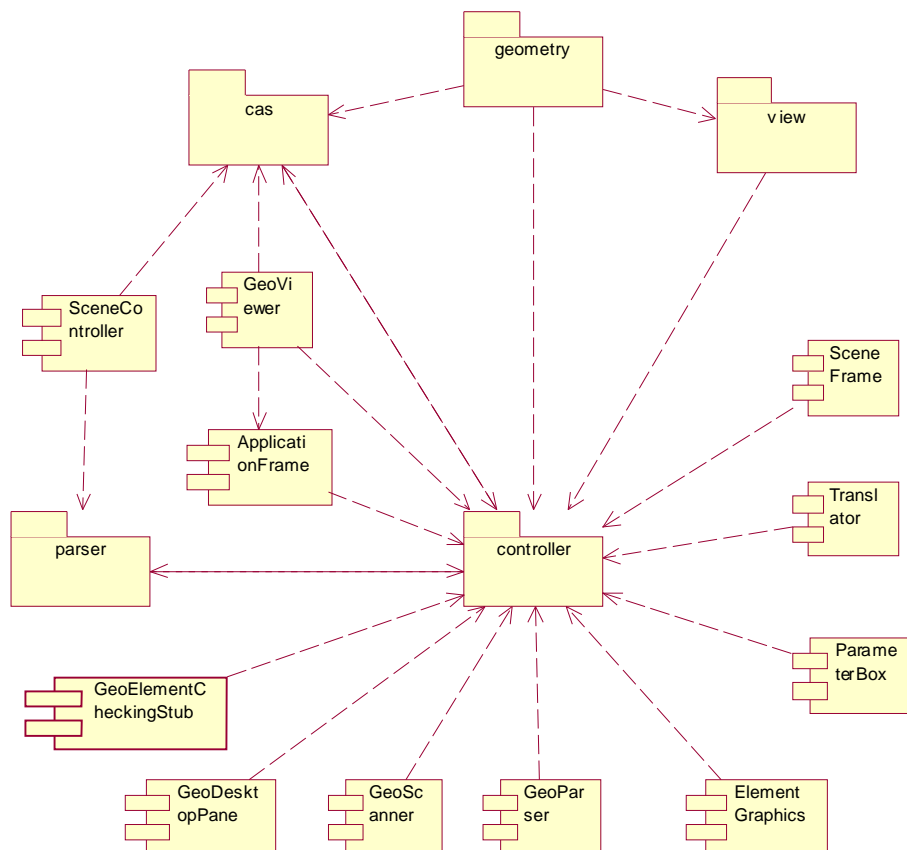
GeoElementCheckingStub:

Testet ob die Elemente korrekt gezeichnet wurden.

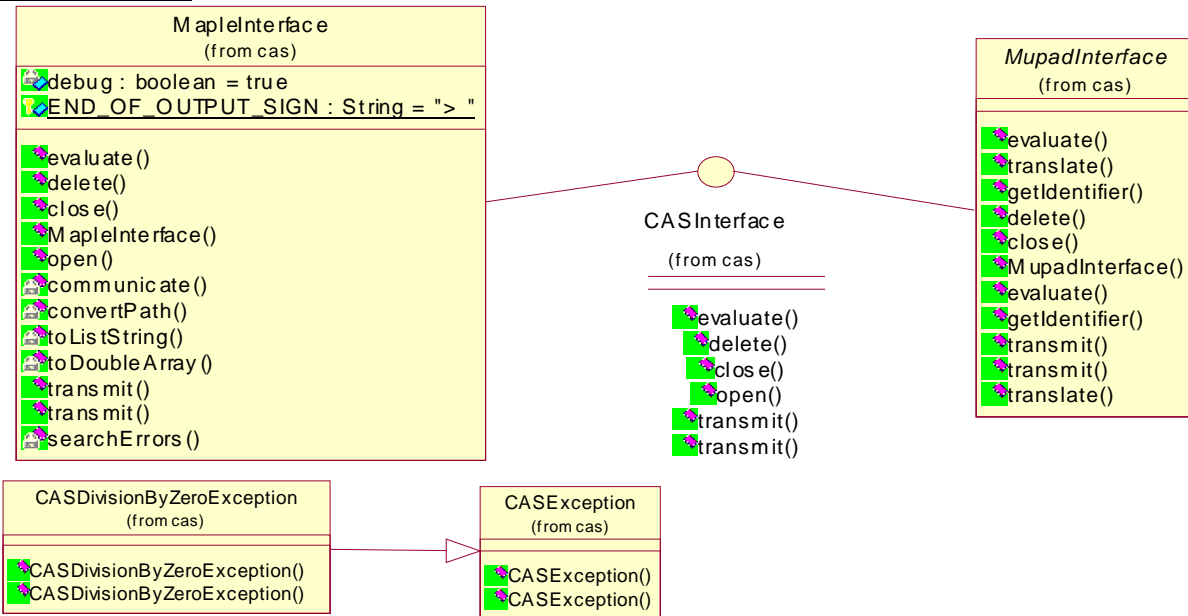
WindowMenu:

Implementiert ein Menü das dem Nutzer erlaubt einen Frame aus dem damit verbundenen Desktop zu wählen. Die Funktionalität stellt der DesktopAdapter und der FrameButton zur Verfügung.

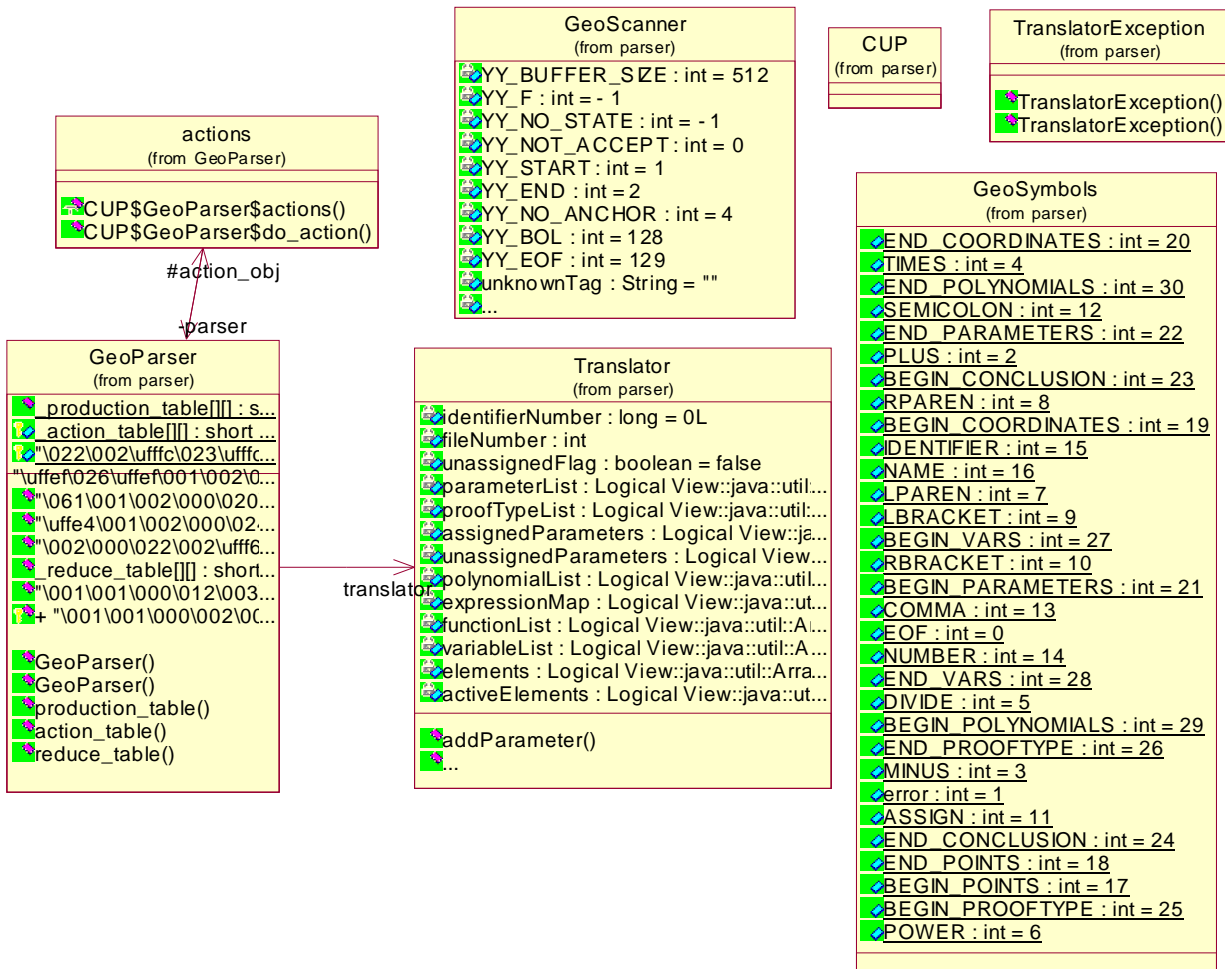
4. Paket und Klassenstruktur



Das Paket Cas:

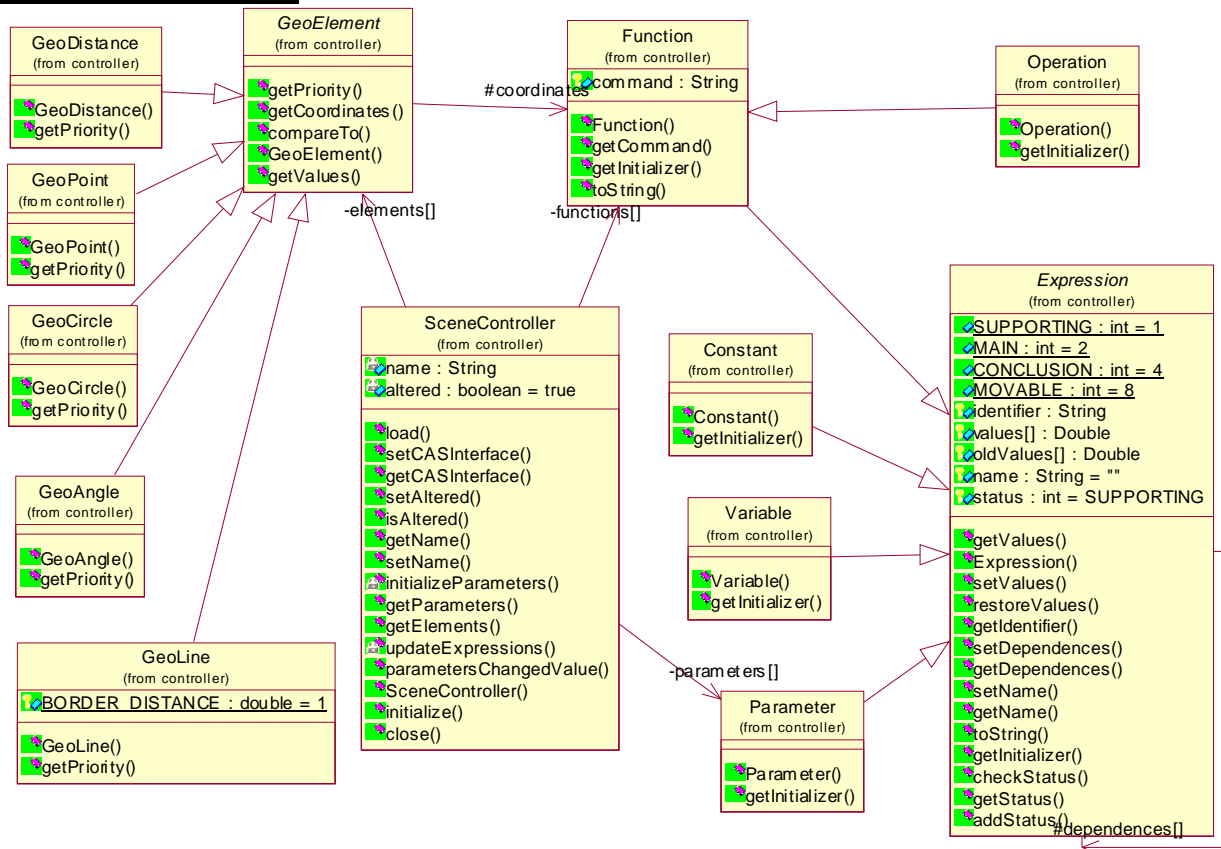


Das Paket Parser:



Erstelldatum: 25.05.2003
 Gruppe: geo03
 Verantwortliche für Design: Sabine Lehmann

Das Paket Controller:



Erstelldatum: 25.05.2003
Gruppe: geo03
Verantwortliche für Design: Sabine Lehmann

Das Paket View:

