

Teil I

Packages und Klassen

In den folgenden Abschnitten sollen die Packages und die darin enthaltenen Klassen und Methoden näher erläutert werden.

1 Package de.BibTexMaker

Das Package *de.BibTexMaker* beinhaltet das gesamte Programm BIBTEXMAKER. Die darin enthaltenen Packages und Klassen werden im folgenden aufgezählt und deren Funktion erläutert.

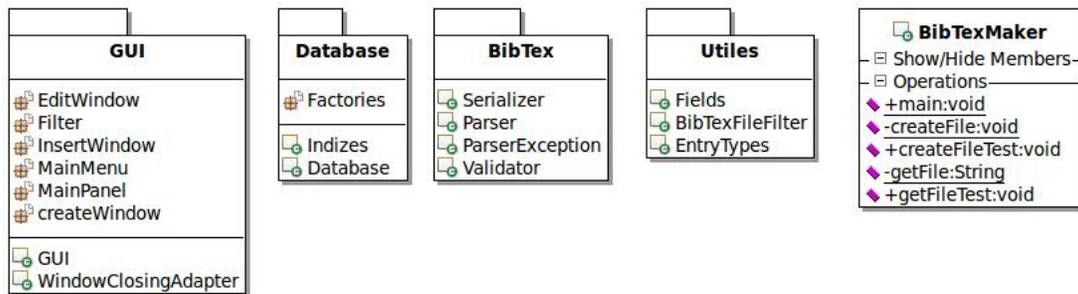


Abbildung 1: Klassendiagramm Package *de.BibTexMaker*

1.1 Klasse BibTexMaker

Die Klasse *BibTexMaker* beinhaltet die Methoden *main*, *createFile* und *getFile*. Die Klasse bildet den Einstieg in das Programm BIBTEXMAKER

Die Methode *main* ist der Einstieg in das Programm BIBTEXMAKER. Sie prüft ob Argumente übergeben wurden und reagiert entsprechend. Wurde ein Dateiname mit Pfad für eine BibTexMaker-Datenbank-Datei übergeben, so veranlasst die Methode das Laden dieser. Ansonsten wird eine leere Datenbank (`./library.xml`) angelegt. Dabei greift die Methode *main* auf die Methode *getFile* zu und übergibt dieser die Argumentenliste.

Die Methode *getFile* liest die dem Programm übergebenen Parameter aus und wertet den Parameter *-f* aus und gibt den Dateinamen mit Pfadangabe zurück. Der Parameter *-f* ist dafür gedacht eine bereits vorhandenen Datenbankdatei direkt beim Programmstart zu laden. Wurde kein Parameter *-f* übergeben, so wird der Methode *main* ein leerer String zurückgegeben.

Die Methode *main* prüft nun ob die Datei existiert. Existiert die Datei noch nicht wird diese erzeugt. Dazu wird die Methode *createFile* aufgerufen.

Die Methode *createFile* erstellt eine neue Datenbankdatei mit dem entsprechenden Format. Anschließend übergibt die Methode *main* den Pfad und den Dateinamen der Datenbank, so dass diese die Daten laden kann.

Zum Schluss startet die Methode *main* die GUI.

1.2 Package de.BibTexMaker.GUI

Das Package *de.BibTexMaker.GUI* beinhaltet alle Packages und Klassen welche zum Graphical User Interface (GUI) gehören.

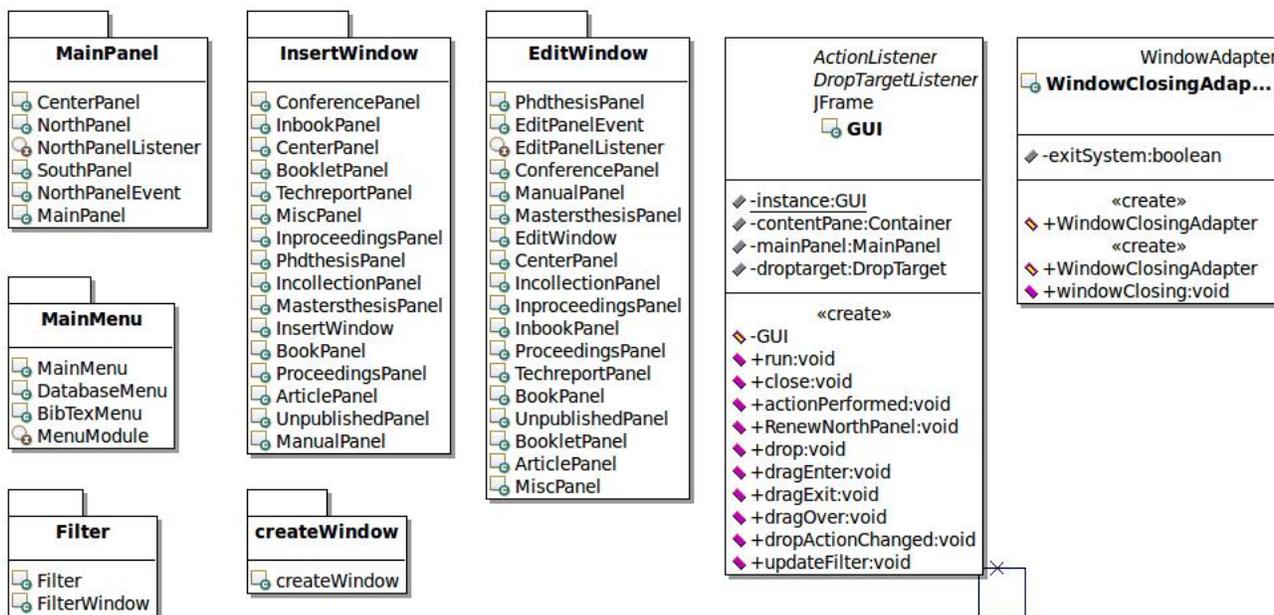


Abbildung 2: Klassendiagramm Package *de.BibTexMaker.GUI*

1.2.1 Klasse GUI

Die Klasse *GUI* ist, wie die Abkürzung schon vermuten lässt, das Graphic Unit Interface des Programms bzw. beinhaltet alle notwendigen Methoden, um dieses darzustellen. Die Klasse ist nach dem Singleton-Pattern-Verfahren programmiert. Das heißt es kann im Programm nur eine Instanz der Klasse *GUI* erzeugt werden. Dazu wurde der Konstruktor *private* gekennzeichnet und eine Methode *getInstance* implementiert. Mithilfe der

Methode *getInstance* kann auf die Instanz der Klasse zugegriffen werden. Ist die Instanz noch nicht angelegt, wird Sie durch die Methode *getInstance* erzeugt.

Mithilfe der Methode *run* wird die GUI zur Anzeige gebracht. Die Methode stellt dabei die Größe des Bildschirms fest und berechnet entsprechend die Größe des Hauptfensters und setzt dieses in die Mitte des Bildschirms.

Die Methode *close* kann dazu genutzt werden einen *Window_Closing-Event* auszulösen, um damit das Fenster oder das Programm zu beenden.

Mit der Methode *RenewNorthPanel* wird veranlasst, dass das *NorthPanel* neu aufgebaut wird und somit aktualisiert wird. Das *NorthPanel* beinhaltet die Buttons, die Comboboxen zur Auswahl des Publikationstypen usw.

Die Methode *drop* reagiert auf einen *Drop-Event* und veranlasst das Prüfen des übergebenen Elementes. Liegt hier eine BibTex-Datei vor, so wird diese an den Parser weitergeleitet, damit die enthaltenden Daten in die Datenbank aufgenommen werden.

Die Methode *updateFilter* löst das aktualisieren des Filters aus.

Alle anderen Methoden haben keine Funktion, sondern sind vorhanden, da dies die Implementation der Interfaces *ActionListener* und *DropTargetListener* erfordert.

1.2.2 Klasse *WindowClosingAdapter*

Die Klasse *WindowClosingAdapter* dient zur Entscheidung, ob nur das Hauptfenster geschlossen werden soll oder das gesamte Programm beendet werden soll, wenn ein *Window_Closing-Event* ausgelöst wird.

Dazu kann die Klasse über zwei Konstruktoren instantiiert werden. Wird ein Objekt der Klasse *WindowClosingAdapter* ohne Übergabe des Parameters *exitSystem* übergeben, so wird bei Auslösen des *Window_Closing-Events* lediglich das Hauptfenster geschlossen. Wird der Parameter übergeben, dann wird bei *true* auch das Programm beendet und nicht nur das Hauptfenster geschlossen.

Wird nun ein *Window_Closing-Event* ausgelöst, dann wird die Methode *windowClosing* ausgeführt. Dabei schließt Sie das Filterfenster falls geöffnet und das Hauptfenster. Anschließend wird geprüft ob das Attribut *exitSystem* auf *true* gesetzt wurde. Ist dies der Fall veranlasst die Methode das Beenden des Programmes.

1.2.3 Package MainPanel

Das Package *MainPanel* beinhaltet alle Klassen zum Aufbau des Hauptfensters.

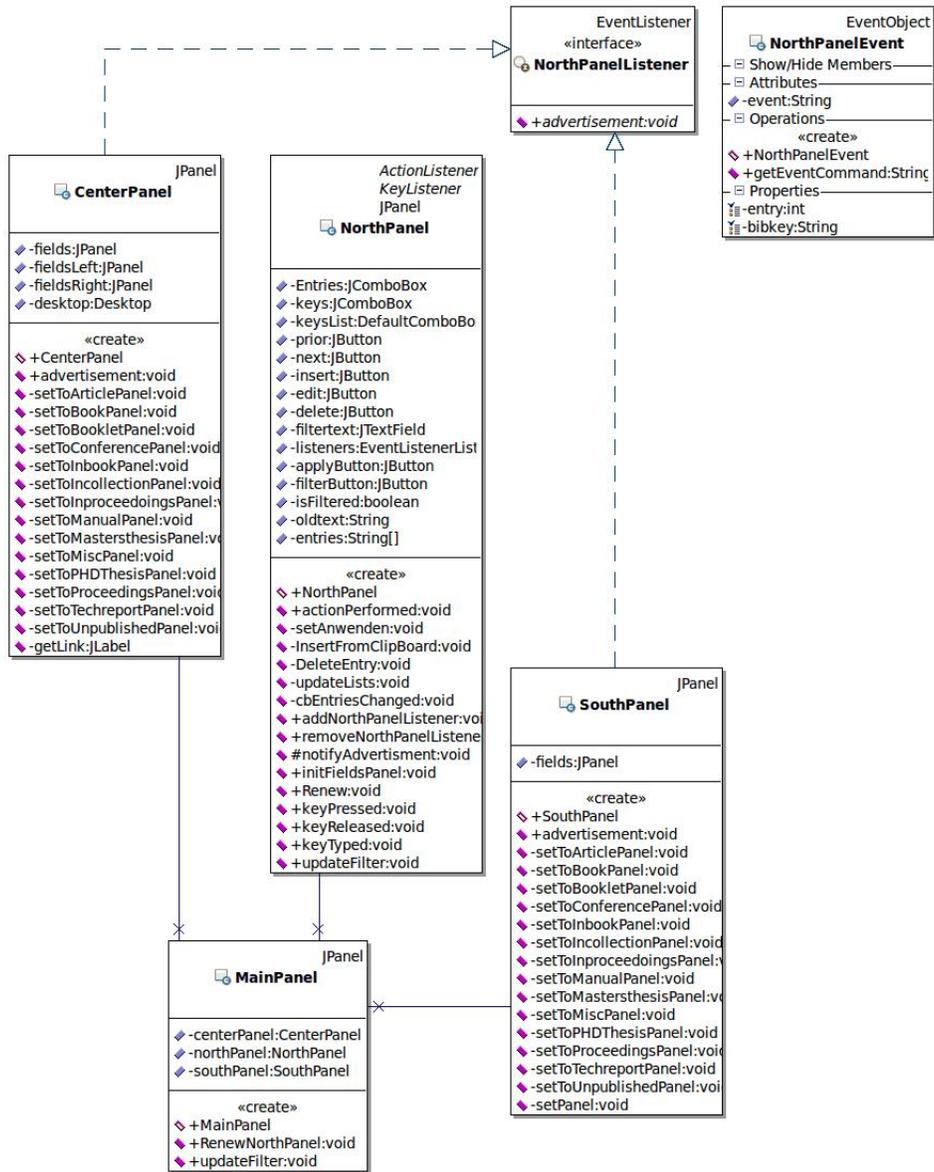


Abbildung 3: Klassendiagramm Package *de.BibTexMaker.GUI.MainPanel*

1.2.3.1 Klasse MainPanel Die Klasse `MainPanel` beinhaltet alle Methoden zum Aufbau des Hauptfensters. Dazu wird im Konstruktor das Hauptfenster initialisiert. Dazu wird das `Layout` und die `Border` festgelegt. Als `Layout` wird das `BorderLayout` verwendet. An die Position `North` kommt das `NorthPanel`, Center das `CenterPanel` und an die Position `South` das `SouthPanel`. Es werden als Attribut jeweils ein Objekt des `North`-, `Center`- und `SouthPanel` angelegt.

Des Weiteren meldet der Konstruktor das `CenterPanel` und das `SouthPanel` am `NorthPanelListener` an, damit diese auf `NorthPanelEvents` reagieren können.

Zum Schluss wird die Methode `initFieldsPanel` des `NorthPanel`s ausgeführt, um die `Comboboxen` für den `Publikationstyp` und der `Auswahl` der `Publikation` zu füllen.

Zur Klasse `NorthPanel` gehören auch noch die Methoden `RenewNorthPanel` und `updateFilter`, welche das Aktualisieren des `Filterfensters` und des `NorthPanel`s veranlassen.

Das `MainPanel` ist eine abgeleitete Klasse von `JPanel`.

1.2.3.2 Klasse NorthPanel

1.2.3.3 Interface NorthPanelListener

1.2.3.4 Klasse NorthPanelEvent

1.2.3.5 Klasse CenterPanel

1.2.3.6 Klasse SouthPanel

1.2.4 Package MainMenu

1.2.4.1 Klasse MainMenu

1.2.4.2 Klasse DatabaseMenu

1.2.4.3 Klasse BibTexMenu

1.2.4.4 Interface MenuModule

1.2.5 Package Filter

1.2.5.1 Klasse Filter

1.2.5.2 Klasse FilterWindow

1.2.6 Package InsertWindow

1.2.6.1 Klasse InsertWindow

1.2.6.2 Klasse CenterPanel

1.2.6.3 Klasse ArticlePanel

1.2.6.4 Klasse BookPanel

1.2.6.5 Klasse BookletPanel

1.2.6.6 Klasse ConferencePanel

1.2.6.7 Klasse InbookPanel

1.2.6.8 Klasse TechreportPanel

1.2.6.9 Klasse MiscPanel

1.2.6.10 Klasse InproceedingsPanel

1.2.6.11 Klasse PhdthesisPanel

1.2.6.12 Klasse IncollectionPanel

1.2.6.13 Klasse MastersthesisPanel

1.2.6.14 Klasse ProceedingsPanel

1.2.6.15 Klasse UnpublishedPanel

1.2.6.16 Klasse ManualPanel

1.2.7 Package EditWindow

1.2.7.1 Klasse EditWindow

1.2.7.2 Klasse CenterPanel

1.2.7.3 Klasse EditPanelListener

1.2.7.4 Klasse EditPanelEvent

1.2.7.5 Klasse PhdthesisPanel

1.2.7.6 Klasse ConferencePanel

1.2.7.7 Klasse ManualPanel

1.2.7.8 Klasse MastersthesisPanel

1.2.7.9 Klasse IncollectionPanel

1.2.7.10 Klasse InproceedingsPanel

1.2.7.11 Klasse InbookPanel

1.2.7.12 Klasse ProceedingsPanel

1.2.7.13 Klasse TechreportPanel

1.2.7.14 Klasse BookPanel

1.2.7.15 Klasse UnpublishedPanel

1.2.7.16 Klasse BookletPanel

1.2.7.17 Klasse ArticlePanel

1.2.7.18 Klasse MiscPanel

1.2.8 Package createWindow

1.2.8.1 Klasse createWindow

1.3 Package de.BibTexMaker.Database

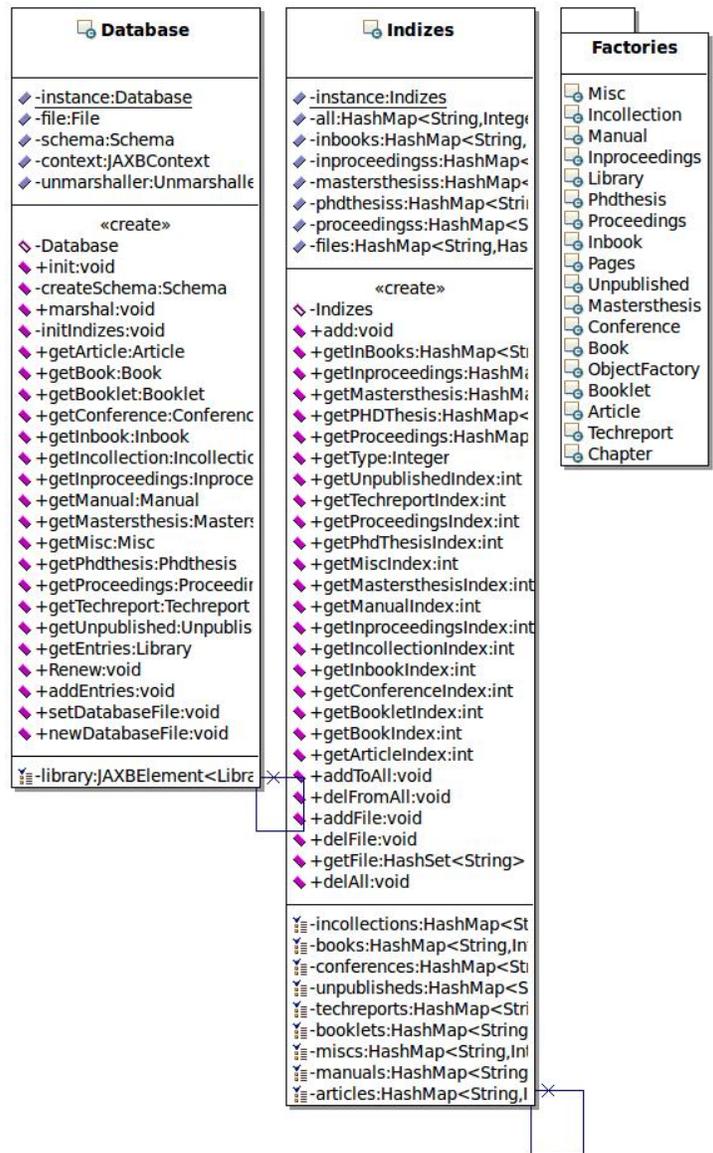


Abbildung 4: Klassendiagramm Package *de.BibTexMaker.Database*

1.3.1 Klasse Database

1.3.1.1 Klasse Indizes

1.3.1.2 Packages Factories

1.4 Package de.BibTexMaker.BibTex

1.4.1 Klasse Parser

1.4.2 Klasse ParserException

1.4.3 Klasse Serializer

1.4.4 Klasse Validator

1.5 Package de.BibTexMaker.Utiles

1.5.1 Klasse Fields

1.5.2 Klasse BibTexFileFilter

1.5.3 Klasse EntryTypes