

{lithie-boxes}

Florian Sihler

22. April 2022

Zusammenfassung

Dieses Paket ist die eigenständige, (immens) verbesserte und polierte Variante des Box-Mechanismus des **Lilly-Frameworks**. Die Namen einiger Bezeichner innerhalb dieses Dokuments hängen von der konfigurierten Sprache ab. Da dieses Dokument in deutscher Sprache gefasst wurde, sind die hier angegebenen Standardbezeichner auch jeweils in deutscher Sprache gesetzt. Das Paket bietet eine Unterstützung für **color-palettes**, die Quellcode-Ausschnitte in diesem Dokument wurden mit **sopra-listings** aus der **sopra-collection** gesetzt.

Inhaltsverzeichnis

1 Ein Schnelleinstieg	1
1.1 Die Boxen	1
1.2 Stile	2
1.3 Nummerierung	3
1.4 Hinweis zu den Listen	3
2 Nutzerbefehle	3
2.1 Die Standardboxen	3
2.2 Referenzen	6
2.3 Verändern bestehender Boxen	6
2.4 Stile	7
2.5 Eigene Boxen definieren	7
2.6 Weitere Befehle	9
3 Paketoptionen	9
4 Versionierung	10
4.1 Version 1.0	10
4.2 Version 1.1	10
Befehlsübersicht	11
Beispielhafte Listen	11

1 Ein Schnelleinstieg

Bevor die genauere Betrachtung der Boxmechanik ansteht, ist hier eine Kurzübersicht über die Verwendung. Grundlegend werden durch das Paket sieben Boxen geladen, deren Bezeichner und derer Textsatz von der konfigurierten Sprache abhängen⁽¹⁾. Die Sprache beeinflusst selbst den Bezeichner der Umgebungen. Diese können durch die **Paketoptionen** allerdings auch unabhängig von der Ausgabe-Sprache generiert werden. Wie bereits zu Beginn erwähnt, wird im Rahmen der Dokumentation die deutsche Sprache verwendet, die englischen Pendanten sind jeweils angemerkt.

1.1 Die Boxen

Betrachten wir die Darstellung einer Definition durch **definition**:

⁽¹⁾Unterstützt werden die englische und die deutsche Sprache

```

1 \begin{definition}[Referenztitel]{Ich bin eine Definition}
2   \label{def:example}Dies ist der Text innerhalb der Definition.
3 \end{definition}
4
5 Ein Verweis auf \autoref{def:example}. Eine tolle \boxref{def:example}.

```

Die Struktur ist hierbei für alle Boxen dieselbe. Wir der *Referenztitel* nicht angegeben, so wird der „lange“ Titel gewählt (analog zu Befehlen wie `\section`). Das Ergebnis des eben gezeigten Listings ist das Folgende (Linien zur Trennung eingefügt)⁽²⁾:

Definition 1.1 – Ich bin eine Definition
Dies ist der Text innerhalb der Definition.

Ein Verweis auf **Definition 1.1**. Eine tolle **Definition 1.1 (Referenztitel)**.

Ebenso gibt es die Boxen `bemerkung`, `satz`, `beispiel`, `lemma`, `beweis` und `zusammenfassung`. Jede dieser Boxen besitzt eine Variante mit Sternchen in der sie als wichtig hervorgehoben wird:

```

1 \begin{satz*}[Supersatz]{Ein Beispielsatz}
2   Ich bin ein Beispielsatz.
3 \end{satz*}

```

Das Ergebnis ist das Folgende:

★ **Satz 1.1** – Ein Beispielsatz
Ich bin ein Beispielsatz.

Eine Auflistung aller Boxen gibt es mittels des Befehls `\listof<enboxpl>` wobei `enboxpl` den englischen Plural des Boxtitels bezeichnet⁽³⁾. So ist die Ausgabe von `\listofdefinitions` exemplarisch **am Ende** aufgeführt.

1.2 Stile

Die Gestaltung der Boxen kann durch Stile verändert werden. Diese können über `\LoadBoxStyle` geladen und geändert werden. Es können eigene Stile definiert werden (`\libx@UpdateAllBoxStyles`). In diesem Dokument ist der Stil `limerence` geladen. Der Standard-Stil `default` muss nicht manuell geladen werden, erlaubt es aber den Stil im Dokument zu ändern:

★ **Definition 1.2**

Definition im standard Stil

Ich fasse Fässer zusammen.

Die Stile können durch normale L^AT_EX-Gruppen begrenzt werden. Nach dem Verlassen wird der letzte Stil wiederhergestellt:

Definition 1.3 – Wieder eine normale Definition
Back to normal.

Darüber hinaus besitzt jede Box eine „Plain“-Variante (diese kann und wird auch vom jeweiligen Stil geändert). Für mehr Informationen über die Verwendung siehe `\SetBoxPlain` oder die `Paketoption` `autostyle`.

⁽²⁾Der Befehl `\autoref` entstammt dem Paket `hyperref` welches nicht geladen werden oder sein muss um die Boxen zu verwenden. Für weitere Befehle siehe `\boxref` oder `\boxnameref`.

⁽³⁾Der Befehl ändert sich auch nicht durch eine andere Sprache oder Optionen, da Bezeichner wie `\listofbemerkungen` sehr seltsam wirken.

Zusammenfassung 2.1 – Ich bin eine fliegende Zusammenfassung

Dies ist der Text einer fliegenden Zusammenfassung.

1.3 Nummerierung

Die Nummerierung der Boxen kann durch `Paketoptionen` kontrolliert werden. So sorgt `lockhigh` zum Beispiel dafür, dass die Nummerierung sich an der höheren Gliederungsebene im Dokument orientiert. Diese Option wird auch in dieser Dokumentation verwendet. Mit `nolock` können diese Sperren deaktiviert werden.

1.4 Hinweis zu den Listen

Das Paket schiebt ein paar Abstraktionen zwischen dem Setzen einer Box und dem Eintrag in die Liste. Dies sorgt dafür, dass auch im Falle von „schwebenden“ (floating) Boxen die Reihenfolge im L^AT_EX-Code eingehalten wird.

2 Nutzerbefehle

2.1 Die Standardboxen

Hinweis für die Beispiele: Die Dokumentation macht *nicht* von der `Paketoptionen` `autostyle` gebrauch. Die Farben werden allerdings durch die Palette „Peach“.

★ Bemerkung 2.1 – Bezüglich des Klammerarguments

Jede Box unterstützt auch das Argument `Boxargs`. Dieses ist ebenfalls optional, erlaubt es aber weitere Argumente an die zugrundeliegende Box zu übergeben! Möchte man zum Beispiel eine Floating-zusammenfassung haben, so genügt:

```
1 \begin{zusammenfassung}(float){Ich bin eine fliegende Zusammenfassung}
2   Dies ist der Text einer fliegenden Zusammenfassung.
3 \end{zusammenfassung}
```

Die Ausgabe findet sich in [Zusammenfassung 2.1 \(Ich bin eine fliegende Zusammenfassung\)](#). Möchte man in einem Zweispaltigen Dokument eine Definition die beide Spalten überspannt so gilt es einfach `float*` anstelle von `float` zu verwenden^(a). Über `floatplacement` oder `float=htbp`, können Platzierungshinweise analog zum optionalen Argument bei Umgebungen wie `figure` angegeben werden. Für mehr Informationen empfiehlt sich die Dokumentation von `tcolorbox`. Natürlich kann so auch ein Referenztitel übergeben werden:

```
1 \begin{zusammenfassung}(float*=b, width=\textwidth)[Und ich der Referenztitel]
2   {Ich bin auch eine fliegende Zusammenfassung}
3   Dies ist der Text einer fliegenden Zusammenfassung.
4 \end{zusammenfassung}
```

Für das Ergebnis siehe hier: [Zusammenfassung 2.2 \(Und ich der Referenztitel\)](#).

Wichtige Anmerkung: Der Schlüssel `hypertarget` sollte nur mit äußerster Vorsicht modifiziert werden, da er für interne Mechaniken des Pakets verwendet werden!

^(a)Wichtig! Wie auch die Dokumentation von `tcolorbox` darauf hinweist, ist im Falle einer wirklich über beide Spalten gehende Box auch noch `width=\textwidth` anzufügen.

`definition (Boxargs) [Referenztitel] {Titel}`

v1.0

Setzt eine „Definition“-Box. Kommandos im Titel müssen robust sein. Wird kein `Referenztitel` angegeben, so wird der `Titel` als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von `Boxargs` wird in [Bemerkung 2.1](#) erklärt.

Zusammenfassung 2.2 – Ich bin auch eine fliegende Zusammenfassung

Dies ist der Text einer fliegenden Zusammenfassung.

Die englische Umgebung trägt den Namen `definition`. Der Bezeichner für `\autoref` befindet sich in `\definitionsname` (Definition), die Liste aller Beispiele kann durch `\listofdefinitions` gesetzt werden. Der Titel der Liste befindet sich in `\definitionslistname` (Alle Definitionen).

Der interne Stil der Box trägt den Bezeichner `Definition`, die interne Id ist `def`, die Datei aller Boxen dieses Typs trägt das Suffix `ldefs`.

Beispiel:

Definition 2.1 – Trooper-Box für Definition

Ich bin eine Beispiel-Box-Fox. Für die „Definition“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

`bemerkung (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Bemerkung“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `remark`. Der Bezeichner für `\autoref` befindet sich in `\remarksname` (Bemerkung), die Liste aller Beispiele kann durch `\listofremarks` gesetzt werden. Der Titel der Liste befindet sich in `\remarkslistname` (Alle Bemerkungen).

Der interne Stil der Box trägt den Bezeichner `Remark`, die interne Id ist `bem`, die Datei aller Boxen dieses Typs trägt das Suffix `lrem`.

Beispiel:

Bemerkung 2.2 – Trooper-Box für Bemerkung

Ich bin eine Beispiel-Box-Fox. Für die „Bemerkung“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

`satz (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Satz“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `theorem`. Der Bezeichner für `\autoref` befindet sich in `\theoremsname` (Satz), die Liste aller Beispiele kann durch `\listoftheorems` gesetzt werden. Der Titel der Liste befindet sich in `\theoremslistname` (Alle Sätze).

Der interne Stil der Box trägt den Bezeichner `Theorem`, die interne Id ist `sat`, die Datei aller Boxen dieses Typs trägt das Suffix `ltheo`.

Beispiel:

Satz 2.1 – Trooper-Box für Satz

Ich bin eine Beispiel-Box-Fox. Für die „Satz“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

`beispiel (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Beispiel“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `example`. Der Bezeichner für `\autoref` befindet sich in `\examplesname` (Beispiel), die Liste aller Beispiele kann durch `\listofexamples` gesetzt werden. Der Titel der Liste befindet sich in `\exampleslistname` (Alle Beispiele).

Der interne Stil der Box trägt den Bezeichner `Example`, die interne Id ist `bei`, die Datei aller Boxen dieses Typs trägt das Suffix `lexa`.

Beispiel:

Beispiel 2.1 – Trooper-Box für Beispiel

Ich bin eine Beispiel-Box-Fox. Für die „Beispiel“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

`lemma (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Lemma“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `lemma`. Der Bezeichner für `\autoref` befindet sich in `\lemmasname` (Lemma), die Liste aller Beispiele kann durch `\listoflemmas` gesetzt werden. Der Titel der Liste befindet sich in `\lemmaslistname` (Alle Lemmata).

Der interne Stil der Box trägt den Bezeichner `Lemma`, die interne Id ist `lem`, die Datei aller Boxen dieses Typs trägt das Suffix `llem`.

Beispiel:

Lemma 2.1 – Trooper-Box für Lemma

Ich bin eine Beispiel-Box-Fox. Für die „Lemma“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

`beweis (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Beweis“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `proof`. Der Bezeichner für `\autoref` befindet sich in `\proofsname` (Beweis), die Liste aller Beispiele kann durch `\listofproofs` gesetzt werden. Der Titel der Liste befindet sich in `\proofslistname` (Alle Beweise).

Der interne Stil der Box trägt den Bezeichner `Proof`, die interne Id ist `bew`, die Datei aller Boxen dieses Typs trägt das Suffix `lpro`.

Beispiel:

Beweis 2.1 – Trooper-Box für Beweis

Ich bin eine Beispiel-Box-Fox. Für die „Beweis“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält. \square

`zusammenfassung (Boxargs) [Referenztitel] { Titel }`

v1.0

Setzt eine „Zusammenfassung“-Box. Kommandos im Titel müssen robust sein. Wird kein *Referenztitel* angegeben, so wird der *Titel* als Referenztitel für Querverweise verwendet. Die gesternte Variante der Umgebung wird durch das Emblem „★“ von den normalen Boxen abgesetzt. Die Bedeutung von *Boxargs* wird in [Bemerkung 2.1](#) erklärt.

Die englische Umgebung trägt den Namen `summary`. Der Bezeichner für `\autoref` befindet sich in `\summariesname` (Zusammenfassung), die Liste aller Beispiele kann durch `\listofsummaries` gesetzt werden. Der Titel der Liste befindet sich in `\summarieslistname` (Alle Zusammenfassungen).

Der interne Stil der Box trägt den Bezeichner `Summary`, die interne Id ist `zsf`, die Datei aller Boxen dieses Typs trägt das Suffix `lsum`.

Beispiel:

Zusammenfassung 2.3 – Trooper-Box für Zusammenfassung

Ich bin eine Beispiel-Box-Fox. Für die „Zusammenfassung“-Box. Ach ja und ich bin Text um zu Erzwingen, dass ein Zeilenumbruch in der Box stattfindet und man auch sieht wie sich das mit nem Zeilenumbruch so verhält.

2.2 Referenzen

Um eine Unabhängigkeit vom Befehl `\autoref` zu erzielen gibt es im Folgenden Referenzbefehle die in der Lage sind, die selbe Ausgabe zu erzeugen. Diese sind allerdings nicht von den `\<Box>name`-Befehlen beeinflusst⁽⁴⁾.

`\boxref{*}{anchor}` v1.0

Erzeugt einen Querverweis analog zu `\nameref` oder `\ref`. Angezeigt wird der Boxbezeichner samt Nummer wie auch des Titels: `\boxref{def:example}` erzeugt: „Definition 1.1 (Referenztitel)“. Mit dem Stern wird allerdings nur die Kurzform ausgegeben (gleich `\autoref`). So ergibt `\boxref*{def:example}` die Ausgabe: „Definition 1.1“. Der Befehl ist robust.

`\boxnameref{*}{anchor}` v1.0

Erzeugt einen Querverweis analog zu `\nameref` oder `\ref`. Nach aktuellem Stand entspricht die Variante ohne Stern der von `\boxref`. Mit Stern ergibt sich allerdings nur noch der eigentliche Titel `\boxnameref*{def:example}` der Box als Ausgabe: „Referenztitel“. Der Befehl ist robust.

2.3 Verändern bestehender Boxen

Die folgenden Befehle sind in sofern noch unvollständig, als dass sie es (noch) nicht erlauben, alle Eigenschaften einer Box zu modifizieren. Sie bieten (hauptsächlich) ein freundliches Interface internen Befehle `\libx@update@boxdata` und `\libx@update@boxif`. Hinweis: Die eigentliche Signatur mancher Befehle mag hier (was optionale Argumente angeht) von der präsentierten Abweichen. Dies liegt daran, dass manche Optionen für die Zukunft bereits eingebaut wurden, ohne dass sie einen Zweck erfüllen oder abgesichert wurden (Beispiel: `\SetBoxPlain`).

`\SetBoxPlain{*}{Boxbezeichner}` v1.0

Sorgt dafür, dass die Box in der „Plain“-Variante gesetzt wird. Interessant ist, dass die „gesternte“ Box-Variante unabhängig von der normalen auf Plain gesetzt werden kann (und umgekehrt). Liefert man aber den optionalen Stern für `\SetBoxPlain`, so wird die Variante für beide Umgebungen aktualisiert. Betrachten wir ein Beispiel:

Definition 2.2 – Ich bin die Plaine-Ihräine

Und ich bin der Text der das Böxoli ziert.

Ich bin die Sonne, die Wonne, der Glanz, da um mich tanzt; die Welt in der Hand und es trifft sich zum Schank; ach dann und wann ein jed-wed jeder der Lächeln noch kann.]

Wie leicht sichtbar ist, wird im geladenen Stil (default) ein Endsymbol gesetzt um zu signalisieren, dass die Box hier vorbei ist. Möchte man wieder zur „normalen“ Variante so ist dies aktuell möglich, in dem man ein leeres optionales Argument übergibt. Also: `\SetBoxPlain[]{definition}` (als Beispiel).

`\SetBoxesPlain{*}{Boxliste}` v1.0

Dieser Befehl arbeitet analog zu `\SetBoxPlain`, erlaubt es aber mittels Kommata separiert, mehrere Boxbezeichner zu übergeben. Für jeden einzelnen wird dann `\SetBoxPlain` entsprechend ausgeführt.

`\SetBoxDisabled{*}{true/false}{Boxbezeichner}` v1.0

Sorgt dafür, dass die Box nicht gesetzt wird. Interessant ist, dass die „gesternte“ Box-Variante unabhängig von der normalen auf versteckt werden kann (und umgekehrt). Liefert man aber den optionalen Stern für `\SetBoxDisabled`, so werden beide Umgebungen versteckt. Möchte man die Boxen wieder anzeigen so genügt das Übergeben von `true`⁽⁵⁾: `\SetBoxDisabled[true]{definition}`.

`\SetBoxesDisabled{*}{true/false}{Boxliste}` v1.0

Dieser Befehl arbeitet analog wie `\SetBoxesPlain` zu `\SetBoxPlain`, erlaubt es aber mittels Kommata separiert, mehrere Boxbezeichner zu übergeben. Für jeden einzelnen wird dann `\SetBoxDisabled` entsprechend ausgeführt.

`\SetBoxColor{*}{Boxbezeichner}{Farbe}` v1.0

Ändert die Farbe der angegebenen Box. Interessant ist, dass die „gesternte“ Box-Variante unabhängig von der

⁽⁴⁾Das heißt, dass eine um-definition von `\definitionsname` die Ausgabe von `\autoref` verändert, aber nicht die von `\boxref`. Beispiel (`\definitionsname` wurde auf „Walter“ geändert): `Walter 1.1 (\autoref)`, `Definition 1.1 (\boxref)`.

⁽⁵⁾Natürlich wirkt das nicht intuitiv, `true` zu übergeben, damit die Box wieder angezeigt wird. Dies kann sich in kommenden Versionen noch ändern.

normalenverändert werden kann (und umgekehrt). Liefert man aber den optionalen Stern für `\SetBoxPlain`, so wird die Farbe für beide Umgebungen aktualisiert. Setzen wir zum Beispiel die Farbe der Definitionen auf ein knalliges gift-grün:

Definition 2.3 – Beispieldefinition
Hilfe ich bin ein verzauberter Giftzweig!

2.4 Stile

`\LoadBoxStyle { Bezeichner }` v1.0

Lädt eine Datei mit der Bezeichnung: „*Bezeichner*.style“. Gesucht wird an den `\input`-Pfad.

`\CurrentBoxStyle` v1.0

Dieses Makro sollte nicht neu definiert werden. Es enthält den Bezeichner des aktuell geladenen Stils (default).

`\RefreshBoxStyle` v1.1

Lädt den `\CurrentBoxStyle` noch einmal. Kann dazu verwendet werden um nachträglich definierte Boxen mit einzugliedern.

2.5 Eigene Boxen definieren

Hinweis: das definieren eigener Boxen ist aktuell noch etwas umständlich, sofern man „abgefahrene Sachen“ machen will (wie der Block am Ende eines Beweises; siehe [beweis](#)). In diesem Fall kann aktuell nur der Quellcode des Pakets selbst als Beispiel herhalten.

`\RegisterBox { Boxkeys } { Tikzkeys } { Boxname } { Titel } { GroupId }` v1.0

Erzeugt eine neue Box mit den entsprechenden Parametern. Die (wichtigsten) gültigen Parameter für *Boxkeys* sind [Tabelle 1](#) zu entnehmen. Die *Tikzkeys* erlauben es beliebige Tikz-Kommandos an die Box zu übergeben. Der *Boxname* bezeichnet den späteren Namen der Umgebung, durch die die Box erzeugt werden soll. Der *Titel* bezeichnet nicht den Titel der jeweiligen Box, sondern das Präfix (wie [Definition](#)). Die *GroupId* muss ein eindeutiger (am Besten möglichst kurzer) Bezeichner sein, der intern für die Persistierung und Handhabung der Box verwendet wird. *Eine Boxdefinition muss in der Präambel erfolgen um ungewünschte Seiteneffekte zu vermeiden!*

Betrachten wir einmal ein Beispiel, wie eine Box erzeugt werden kann⁽⁶⁾. Da wir noch nicht wissen, wie mit `\RegisterNewDefaultStyle` eigene Stile definiert werden können, verwenden wir hier den Stil der Definition.

```

1 \newlistof{codeliste}{lcode}{Alle Codeboxen}%
2 \RegisterBox{%
3   list name=codeliste,%
4   list men=lcode,%
5   has emblem=true,%
6   in code=\noindent,%
7   emblem=\protect\faCode,%
8   box color=purple,%
9   use style=libx@box@Definition}%
10 {}{code}{Code}{cde}%

```

Setzen wir doch mal unsere Code-Box (für die Umgebung `java` ist das Paket `sopra-listings` aus der `sopra-collection` vonnöten):

```

1 \begin{code}{Ich bin eine Codebox!}
2 Super Code nicht wahr?
3   \begin{java}
4 System.out.println("Hallo Welt!");
5   \end{java}
6 \end{code}

```

Wir erhalten:

⁽⁶⁾Dieses Beispiel ist im Verhältnis zu den anderen Boxen etwas einfacher gehalten

</> **Code 2.4** – Ich bin eine Codebox!
 Super Code nicht wahr?

```
1 System.out.println("Hallo_Welt!");
```

Wir haben also den Nummer der Definition benutzt, sind aber nicht in der Liste eingetragen, da wir unseren eigenen verwenden! Dafür können wir nun mit `\listofcode` die Liste aller Codeboxen anzeigen:

Alle Codeboxen

2.4 <> Ich bin eine Codebox! 8

Schlüssel	Typ	Default	Beschreibung
<code>box color</code>	color	black	Die Farbe der Box. Muss einige gültige Farbe sein.
<code>pre code</code>	code		(\LaTeX -)Code der vor dem Beginn der Box ausgeführt werden soll.
<code>in code</code>	code		(\LaTeX -)Code der innerhalb der Box vor dem Beginn des eigentlichen Inhalts (aber nach dem Titel) ausgeführt werden soll.
<code>out code</code>	code		(\LaTeX -)Code der vor dem Verlassen der Box ausgeführt werden soll (aber nach dem eigentlichen Inhalt).
<code>post code</code>	code		(\LaTeX -)Code der nach dem Ende der Box ausgeführt werden soll.
<code>use style</code>	style		Der durch <code>\RegisterNewDefaultStyle</code> definierte Stil. Dies kann auch einer der bereits bestehenden Stile sein. Auch wenn der Parameter optional erscheint, so ist er dennoch verpflichtend. Es gilt zu beachten, dass der Stil auch die (volle) Kontrolle über den Zähler besitzt.
<code>emblem</code>	code		Das zu setzende Emblem (wird zum Beispiel bei der gesternten Variante von <code>definition</code> als <code>\faStar</code> gesetzt.). Der Code sollte durch <code>\protect</code> abgesichert werden, wenn es sich um problematische Makros handelt.
<code>has emblem</code>	bool	false	Eigentliche Flagge, die angibt ob das Emblem auch gesetzt werden soll. Dies sorgt auch dafür, dass in den Listen Platz für ein Emblem gehalten wird. So wird zum Beispiel dafür gesorgt, dass die Einrückung auch für Definitionen auch ohne Emblem korrekt ist.
<code>custom list</code>	bool	true	Gibt an ob sich die Box in einer Liste registrieren soll. Ist dieser Wert auf „wahr“, so ist es unabdingbar <code>list name</code> , <code>list text</code> und <code>list men</code> zu setzen und mit den Informationen der zugrundeliegenden Liste zu füllen.
<code>list name</code>	text		Der Name der zugrundeliegenden Liste (muss extra mit z.B. <code>\newlistof</code> aus dem Paket <code>tocloft</code> definiert werden). Es können auch bestehende Listen verwendet werden.
<code>list men</code>	text		Die Dateibezeichnung der Liste. Die Regeln gelten analog zu <code>list name</code> .

Tabelle 1: Optionen für die Boxkeys in `\RegisterBox`. Hier sind nur die wichtigsten Schlüssel aufgeführt. Manchmal andere wie `name` werden durch andere Befehle bereits gesetzt.

`\RegisterNewDefaultStyle {Name} {Options}`

v1.0

Erzeugt einen neuen Stil der zum Beispiel von `\RegisterBox` verwendet werden kann. Die *Options* bezeichnen weitere `tcolorbox`-Argumente. Erzeugt wird der Stil `libx@box@Name`.

2.6 Weitere Befehle

`\libxlistofheading` { *Command* }

v1.0

Dieser Befehl ist nur dann von Relevanz wenn die Koma-Script-Kompatibilität aktiv ist (sonst gibt er eine Warnung aus und bleibt ohne Bedeutung). Er erlaubt es den Titel eines `\listof<enboxpl>`-Befehls zu konfigurieren. Die Standardkonfiguration ist die Folgende:

```
\libxlistofheading{\section*{#1 \@mkboth{#1}{#1}}}%
```

Hierbei wird #1 automatisch durch den jeweiligen Titel ersetzt. Möchte man also nun einfach, dass die Listen sich als `\subsection` auch im `\tableofcontents` registrieren und davor noch „Hallo Welt“ schreiben, so wäre folgender Befehl möglich:

```
\libxlistofheading{\subsection{#1} Hallo Welt}%
```

Ein Beispiel dazu lässt sich in der `example.pdf` finden, die sich [hier](#) befinden sollte.

3 Paketoptionen

Das Paket erlaubt einige Optionen zur Konfiguration. Die Dokumentation wurde mit den Folgenden Optionen geladen:

```
\usepackage[lockhigh,cpalette]{lithie-boxes}
\LoadBoxStyle{limerence}
```

Nummerierungskontrolle Mit den Optionen `lockhigh` und `locklow` kann konfiguriert werden, wie die Nummerierung des Dokuments von statten geht. Hierbei verwendet `lockhigh` die jeweils höhere Hierarchieebene⁽⁷⁾ und `locklow` die jeweils niedrigere⁽⁸⁾. Derartige Nummerierungen können mit der Option `no-lock` deaktiviert werden (diese Option ist auch der Standard). In diesem Dokument wird `lockhigh` verwendet.

Box-Varianten Standardmäßig werden alle Boxen in der normalen Variante gesetzt. Die Paketoption `autostyle` sorgt allerdings dafür, dass automatisch einige Boxen mit der `\SetBoxPlain`-Routine konfiguriert werden um einen schöneren Fokus zu legen. Die `autostyle`-Option aktiviert auch gleichzeitig die im Folgenden beschriebene Option `cpalette`. Die `noautostyle`-Option deaktiviert die Mechanik (lässt `cpalette` aber unverändert).

Farben Mit der Option `cpalette` muss das Paket `color-palettes` *zuvor* geladen werden. Die Option (welche in der Dokumentation ebenfalls aktiv ist) sorgt dafür, dass die Boxfarben sich an der jeweiligen Palette orientieren. Im Gegensatz zu einer Stil-Änderung beeinflusst die Option allerdings nur die Optionen, die vom Paket mitgeliefert werden. Die Option kann mit `nocpalette` deaktiviert werden.

Umgebungsbezeichner Standardmäßig orientieren sich die Bezeichner der Umgebungen an der Sprache. Dies kann aber problematisch sein, wenn man sich entweder an eine Sprache gewöhnt oder die Sprache im Dokument einfach ändert. Die Option `germanenv` sorgt dafür, dass die Umgebungen unabhängig von der Ausgabesprache die Bezeichner in deutscher Sprache erhalten. Analog verfährt die Option `englishenv` mit der englischen Sprache für die Umgebungen. Die Option `autoenv` wechselt wieder zur Standardkonfiguration, welche den Namen anhand der Dokumentsprache herausfindet⁽⁹⁾.

Koma-Script Das Paket versucht automatisch zu erkennen ob das Koma-Script geladen wird und dann entsprechende Maßnahmen zu ergreifen um bestmöglich kompatibel zu sein (mock von `tocloft`, ...). Sollte die Erkennung allerdings Schief laufen oder zu ungewollten Problemen führen (oder andere Pakete ohnehin einige Optionen vernichten), so kann mit `no-koma` die Anwendung der Kompatibilitätsmaßnahmen deaktiviert werden. Analog lassen sie sich mit `koma` erzwingen.

⁽⁷⁾Also wenn vorhanden: `\chapter`, sonst `\section`.

⁽⁸⁾Ist also `\chapter` vorhanden, wird `\section` gewählt, sonst wird `\subsection` gewählt.

⁽⁹⁾Wichtig: Es ist die konfigurierte Sprache zum Ladezeitpunkt des Pakets von Relevanz. Es empfiehlt sich also Pakete wie `babel` vor diesem Paket zu laden, wenn man von der automatischen Spracherkennung Gebrauch machen möchte.

Sprache Für Übersetzungen verwendet dieses Paket die Hilfe von `translator`, welches die Sprache aus dem Befehl `\languagename` extrahiert. Je nach Installation kann es allerdings sein, dass die Sprache falsch erkannt wird⁽¹⁰⁾! In diesem Fall erzwingt `german` das Setzen in deutscher Sprache (dies beeinflusst `autoenv`, hat aber keinen Einfluss auf die Umgebungsnamen wenn diese mit `englishenv` auf Englisch gezwungen werden) und analog `english` das Setzen in englischer Sprache (mit denselben Wechselwirkungen mit `autoenv` und `englishenv`). Die Option `autolang` ist der Standard und versucht die Sprache aus `\languagename` zu extrahieren⁽¹¹⁾.

Textfluss Der Paketautor⁽¹²⁾, hat eine große Freude an rechteckigen Paragraphen. Diese können aber vor allem in Boxen zu einem schlechteren Textfluss beitragen, weswegen die Option `noparfill` (Standard) derartige Vorkehrungen „zunichte“ macht. Möchte man dennoch derartige Optionen so genügt es, die Option `parfill` anzugeben⁽¹³⁾. Ebenso funktionieren `parskip` und `noparskip` für den Abstand bei Paragraphen.

Bestehende Umgebungen Manche Pakete definieren bereits Umgebungen welche den Namen von Boxumgebungen tragen, was dann natürlich zu Konflikten führt. Mit der Option `overwrite` wird das Paket die bestehenden Umgebungen überschreiben (und keinen Fehler ausgeben, wenn noch keine Umgebung mit dem Namen existiert. In diesem Fall wird einfach eine neue erstellt). Dieses Verhalten kann mit `nooverwrite` unterdrückt werden.

4 Versionierung

4.1 Version 1.0

Initiale Version mit den Boxen `definition`, `bemerkung`, `satz`, `beispiel`, `lemma`, `beweis` und `zusammenfassung`. Auch Befehle wie `\boxref` und `\LoadBoxStyle` wurden eingeführt.

4.2 Version 1.1

Ein paar kleine Verbesserungen wie ein Befehl `\RefreshBoxStyle` wurden vorgenommen. Limerence kann nun die Titelschriftgröße mit `\libxLimerenceNormalSize` und `\libxLimerenceSmallerSize` ändern.

⁽¹⁰⁾Dies sollte nicht geschehen, wenn die gewünschte Sprache der Dokumentklasse selbst übergeben wird.

⁽¹¹⁾Das Paket `iflang` wird nicht verwendet.

⁽¹²⁾Das bin ich :P

⁽¹³⁾Hinweis: Simpel gesagt setzt `noparfill` nur das in `\@libx@noparfill` hinterlegte Makro.

Befehlsübersicht

A		N	
autoenv	9	noautostyle	9
autostyle	9	nocpalette	9
B		nokoma	9
beispiel ^(env)	4	nolock	9
bemerkung ^(env)	4	nooverwrite	10
beweis ^(env)	5	noparfill	10
\boxnameref	6	noparskip	10
\boxref	6	O	
C		overwrite	10
cpalette	9	P	
\CurrentBoxStyle	7	parfill	10
D		parskip	10
definition ^(eng. → definition)	3	proof ^(eng. → beweis)	5
definition ^(env)	3	R	
E		\RefreshBoxStyle	7
english	10	\RegisterBox	7
englishenv	9	\RegisterNewDefaultStyle	8
example ^(eng. → beispiel)	4	remark ^(eng. → bemerkung)	4
G		S	
german	10	satz ^(env)	4
germanenv	9	\SetBoxColor	6
K		\SetBoxDisabled	6
koma	9	\SetBoxesDisabled	6
L		\SetBoxesPlain	6
lemma ^(eng. → lemma)	5	\SetBoxPlain	6
lemma ^(env)	5	summary ^(eng. → zusammenfassung)	5
\libxlistofheading	9	T	
\LoadBoxStyle	7	theorem ^(eng. → satz)	4
lockhigh	9	Z	
locklow	9	zusammenfassung ^(env)	5

Alle Definitionen

1.1	Referenztitel	2
1.2	★ Definition im standard Stil	2
1.3	Wieder eine normale Definition	2
2.1	Trooper-Box für Definition	4
2.2	Ich bin die Plaine-Ihräne	6
2.3	Beispieldefinition	7

Alle Sätze

1.1 ★ Supersatz	2
2.1 Trooper-Box für Satz	4

Alle Bemerkungen

2.1 ★ Bezüglich des Klammerarguments	3
2.2 Trooper-Box für Bemerkung	4

Alle Zusammenfassungen

2.1 Ich bin eine fliegende Zusammenfassung	3
2.2 Und ich der Referenztitel	3
2.3 Trooper-Box für Zusammenfassung	5