

L^AT_EX-Toolchain zur Erstellung technisch-wissenschaftlicher Arbeiten

Michael Hammer & Stefan Sollerer
michael.hammer@tugraz.at
stefan.sollerer@tugraz.at

Grazer Linuxtag 2007
<http://www.linuxtage.at>

20. Mai 2007

Kurze Vorstellung

Profil Stefan Sollerer

Mein Name ist Stefan Sollerer (geboren 1979 in Innsbruck)

About me...

- Wirtschaftsingenieur-Maschinbau-Mechatronik (TU Graz)
- Wiss. Assistent am Institut für Festigkeitslehre
- Gentoo, L^AT_EX, Java

Kurze Vorstellung

Profil Michael Hammer

Mein Name ist Michael Hammer (geboren 1981 in Linz)

Mein Werdegang

- Studium: Maschinbau-Mechatronik (TU Graz)
- ab Juni Wiss. Assistent am Institut für Festigkeitslehre
- Numerische Simulation - Finite Elemente in der Festkörpermechanik
- Erstes Linux 1997 SuSE - heute begeisterter Gentoo User

Übersicht

- 1 Einführung
 - latex vs. pdflatex
 - Möglichkeiten zur Fusionierung
- 2 pst-pdf
 - Was kann pst-pdf?
- 3 T_EXToolKit
 - T_EXToolKit im Überblick
 - Preprocessing
 - Mainprocessing
 - Compile Vorgang
 - Übersicht
- 4 Offene Aufgaben

Übersicht

- 1 Einführung
 - latex vs. pdflatex
 - Möglichkeiten zur Fusionierung
- 2 pst-pdf
 - Was kann pst-pdf?
- 3 T_EXToolKit
 - T_EXToolKit im Überblick
 - Preprocessing
 - Mainprocessing
 - Compile Vorgang
 - Übersicht
- 4 Offene Aufgaben

Übersicht

- 1 Einführung
 - latex vs. pdflatex
 - Möglichkeiten zur Fusionierung
- 2 pst-pdf
 - Was kann pst-pdf?
- 3 T_EXToolKit
 - T_EXToolKit im Überblick
 - Preprocessing
 - Mainprocessing
 - Compile Vorgang
 - Übersicht
- 4 Offene Aufgaben

Übersicht

- 1 Einführung
 - latex vs. pdflatex
 - Möglichkeiten zur Fusionierung
- 2 pst-pdf
 - Was kann pst-pdf?
- 3 T_EXToolKit
 - T_EXToolKit im Überblick
 - Preprocessing
 - Mainprocessing
 - Compile Vorgang
 - Übersicht
- 4 Offene Aufgaben

Unterschied von latex zu pdflatex

Viele Gerüchte sind im Umlauf - was soll man nun verwenden?
Macht es einen Unterschied?

Gar nicht so unterschiedlich ...

- **Beide basieren mittlerweile auf dem Satzsystem pdf_ET_EX**
- pdf_ET_EX ist in der Lage sowohl pdf als auch dvi files zu erzeugen
- Wie sieht das nun aus?
- Irgendwo muss doch nun ein Vorteil sein?

Unterschied von latex zu pdflatex

Viele Gerüchte sind im Umlauf - was soll man nun verwenden?
Macht es einen Unterschied?

Gar nicht so unterschiedlich ...

- Beide basieren mittlerweile auf dem Satzsystem pdf T_EX
- pdf T_EX ist in der Lage sowohl pdf als auch dvi files zu erzeugen
- Wie sieht das nun aus?
- Irgendwo muss doch nun ein Vorteil sein?

Unterschied von latex zu pdflatex

Viele Gerüchte sind im Umlauf - was soll man nun verwenden?
Macht es einen Unterschied?

Gar nicht so unterschiedlich ...

- Beide basieren mittlerweile auf dem Satzsystem pdf T_EX
- pdf T_EX ist in der Lage sowohl pdf als auch dvi files zu erzeugen
- **Wie sieht das nun aus?**
- Irgendwo muss doch nun ein Vorteil sein?

Unterschied von latex zu pdflatex

Viele Gerüchte sind im Umlauf - was soll man nun verwenden?
Macht es einen Unterschied?

Gar nicht so unterschiedlich ...

- Beide basieren mittlerweile auf dem Satzsystem pdf \TeX
- pdf \TeX ist in der Lage sowohl pdf als auch dvi files zu erzeugen
- Wie sieht das nun aus?
- **Irgendwo muss doch nun ein Vorteil sein?**

Unterschied von latex zu pdflatex

... aber dann wieder doch sehr verschieden

- **pdf_ET_EX erzeugt direkt ein pdf**
- pdf_ET_EX wird auch typografisch weiter verbessert (z.B. character protruding) - das modernere System
- es unterstützt mehr und modernere Grafikformate
- es unterstützt eine Vielzahl an features die nur in pdf möglich sind (z.b. Autor, Copyright, usw.)
- **ABER** : eps Grafiken und ps-orientierte Pakete können nur im dvi mode verwendet werden :(

Letzter Punkt ist essentiell für technische Zeichnungen in denen mit L^AT_EX gesetzt werden soll oder pstricks Grafiken!

Unterschied von latex zu pdflatex

... aber dann wieder doch sehr verschieden

- pdf $\text{T}_{\text{E}}\text{X}$ erzeugt direkt ein pdf
- pdf $\text{T}_{\text{E}}\text{X}$ wird auch typografisch weiter verbessert (z.B. character protruding) - das modernere System
- es unterstützt mehr und modernere Grafikformate
- es unterstützt eine Vielzahl an features die nur in pdf möglich sind (z.b. Autor, Copyright, usw.)
- **ABER** : eps Grafiken und ps-orientierte Pakete können nur im dvi mode verwendet werden :(

Letzter Punkt ist essentiell für technische Zeichnungen in denen mit $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ gesetzt werden soll oder pstricks Grafiken!

Unterschied von latex zu pdflatex

... aber dann wieder doch sehr verschieden

- pdf_ET_EX erzeugt direkt ein pdf
- pdf_ET_EX wird auch typografisch weiter verbessert (z.B. character protruding) - das modernere System
- **es unterstützt mehr und modernere Grafikformate**
- es unterstützt eine Vielzahl an features die nur in pdf möglich sind (z.b. Autor, Copyright, usw.)
- **ABER** : eps Grafiken und ps-orientierte Pakete können nur im dvi mode verwendet werden :(

Letzter Punkt ist essentiell für technische Zeichnungen in denen mit L^AT_EX gesetzt werden soll oder pstricks Grafiken!

Unterschied von latex zu pdflatex

... aber dann wieder doch sehr verschieden

- pdf_ET_EX erzeugt direkt ein pdf
- pdf_ET_EX wird auch typografisch weiter verbessert (z.B. character protruding) - das modernere System
- es unterstützt mehr und modernere Grafikformate
- **es unterstützt eine Vielzahl an features die nur in pdf möglich sind (z.b. Autor, Copyright, usw.)**
- **ABER** : eps Grafiken und ps-orientierte Pakete können nur im dvi mode verwendet werden :(

Letzter Punkt ist essentiell für technische Zeichnungen in denen mit L^AT_EX gesetzt werden soll oder pstricks Grafiken!

Unterschied von latex zu pdflatex

... aber dann wieder doch sehr verschieden

- pdf_ET_EX erzeugt direkt ein pdf
- pdf_ET_EX wird auch typografisch weiter verbessert (z.B. character protruding) - das modernere System
- es unterstützt mehr und modernere Grafikformate
- es unterstützt eine Vielzahl an features die nur in pdf möglich sind (z.b. Autor, Copyright, usw.)
- **ABER** : eps Grafiken und ps-orientierte Pakete können nur im dvi mode verwendet werden :(

Letzter Punkt ist essentiell für technische Zeichnungen in denen mit L^AT_EX gesetzt werden soll oder pstricks Grafiken!

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- **Umgebung pspicture und postscript**
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die pst-pdf bietet
- aufwendige Abfolge von Hilfskommandos (wrapper auto-pst-pdf)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung `pspicture` und `postscript`
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die `pst-pdf` bietet
- aufwendige Abfolge von Hilfskommandos (wrapper `auto-pst-pdf`)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung pspicture und postscript
- `< \jobname>-pics.pdf`
- **eingeschränkt auf die Möglichkeiten die pst-pdf bietet**
- aufwendige Abfolge von Hilfskommandos (wrapper auto-pst-pdf)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung `pspicture` und `postscript`
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die `pst-pdf` bietet
- **aufwendige Abfolge von Hilfskommandos (wrapper `auto-pst-pdf`)**

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung pspicture und postscript
- < \jobname>-pics.pdf
- eingeschränkt auf die Möglichkeiten die pst-pdf bietet
- aufwendige Abfolge von Hilfskommandos (wrapper auto-pst-pdf)

Eigene toolchain - T_EXToolKit

- **Extraktion der Einzelaufrufe - getrennte pdf files**
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar (eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung pspicture und postscript
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die pst-pdf bietet
- aufwendige Abfolge von Hilfskommandos (wrapper auto-pst-pdf)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- **Verpacken in Skripte und automatisierter Ablauf**
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung `pspicture` und `postscript`
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die `pst-pdf` bietet
- aufwendige Abfolge von Hilfskommandos (wrapper `auto-pst-pdf`)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- **extrem flexibel und erweiterbar**
(eps-Preprocessing)
- Skripte in das Projekt einbetten

Möglichkeiten zur Fusionierung

Wie nutze ich die Vorteile beider Welten?

pst-pdf

- Umgebung pspicture und postscript
- `< \jobname>-pics.pdf`
- eingeschränkt auf die Möglichkeiten die pst-pdf bietet
- aufwendige Abfolge von Hilfskommandos (wrapper auto-pst-pdf)

Eigene toolchain - T_EXToolKit

- Extraktion der Einzelaufrufe
- getrennte pdf files
- Verpacken in Skripte und automatisierter Ablauf
- extrem flexibel und erweiterbar
(eps-Preprocessing)
- **Skripte in das Projekt einbetten**

Was kann p_st-pdf?

Ein Paket zur Verwendung von ps-Code in PDF-Dokumenten.

Funktionsweise

- Man kapselt den ps-Code in `pspicture` oder `postscript` Umgebungen
- In diesen Umgebungen können nun `\psfrag` und `\pstricks` wie gewohnt verwendet werden
- p_st-pdf erzeugt eine Containerdatei `< \jobname>-pics.pdf`
- Die Grafiken aus der Containerdatei werden zum Erstellen des Enddokuments verwendet

<http://www.ctan.org/tex-archive/macros/latex/contrib/pst-pdf/pst-pdf-DE.pdf>

Was kann pst-pdf?

Ein Paket zur Verwendung von ps-Code in PDF-Dokumenten.

Funktionsweise

- Man kapselt den ps-Code in `pspicture` oder `postscript` Umgebungen
- In diesen Umgebungen können nun `\psfrag` und `\pstricks` wie gewohnt verwendet werden
- `pst-pdf` erzeugt eine Containerdatei `< \jobname>-pics.pdf`
- Die Grafiken aus der Containerdatei werden zum Erstellen des Enddokuments verwendet

<http://www.ctan.org/tex-archive/macros/latex/contrib/pst-pdf/pst-pdf-DE.pdf>

Was kann pst-pdf?

Ein Paket zur Verwendung von ps-Code in PDF-Dokumenten.

Funktionsweise

- Man kapselt den ps-Code in `pspicture` oder `postscript` Umgebungen
- In diesen Umgebungen können nun `\psfrag` und `\pstricks` wie gewohnt verwendet werden
- **pst-pdf erzeugt eine Containerdatei `< \jobname >-pics.pdf`**
- Die Grafiken aus der Containerdatei werden zum Erstellen des Enddokuments verwendet

<http://www.ctan.org/tex-archive/macros/latex/contrib/pst-pdf/pst-pdf-DE.pdf>

Was kann pst-pdf?

Ein Paket zur Verwendung von ps-Code in PDF-Dokumenten.

Funktionsweise

- Man kapselt den ps-Code in `pspicture` oder `postscript` Umgebungen
- In diesen Umgebungen können nun `\psfrag` und `\pstricks` wie gewohnt verwendet werden
- `pst-pdf` erzeugt eine Containerdatei `< \jobname>-pics.pdf`
- **Die Grafiken aus der Containerdatei werden zum Erstellen des Enddokuments verwendet**

<http://www.ctan.org/tex-archive/macros/latex/contrib/pst-pdf/pst-pdf-DE.pdf>

Und nun Live ...

... einfach zum Zuschauen und Fragen stellen!

Was ist T_EXToolKit?

Ein alternativer Weg zur Verwendung von ps-Code in pdfL_AT_EX

Kurz umrissen

- **Ist entstanden aus der Notwendigkeit von L_AT_EX Satzanweisungen in eps Grafiken**
- Soll ein universeller Mechanismus zur Erstellung technisch-wissenschaftlicher Arbeiten sein
- Beinhaltet eps Pre- und Mainprocessing und die Einbindung der Grafiken in das Hauptdokument
- Soll platformunabhängig unter Verwendung bestehender etablierter Werkzeuge arbeiten
- Ist in Python geschrieben

Was ist T_EXToolKit?

Ein alternativer Weg zur Verwendung von ps-Code in pdfL_AT_EX

Kurz umrissen

- Ist entstanden aus der Notwendigkeit von L_AT_EX Satzanweisungen in eps Grafiken
- **Soll ein universeller Mechanismus zur Erstellung technisch-wissenschaftlicher Arbeiten sein**
- Beinhaltet eps Pre- und Mainprocessing und die Einbindung der Grafiken in das Hauptdokument
- Soll platformunabhängig unter Verwendung bestehender etablierter Werkzeuge arbeiten
- Ist in Python geschrieben

Was ist T_EXToolKit?

Ein alternativer Weg zur Verwendung von ps-Code in pdfL_AT_EX

Kurz umrissen

- Ist entstanden aus der Notwendigkeit von L_AT_EX Satzanweisungen in eps Grafiken
- Soll ein universeller Mechanismus zur Erstellung technisch-wissenschaftlicher Arbeiten sein
- **Beinhaltet eps Pre- und Mainprocessing und die Einbindung der Grafiken in das Hauptdokument**
- Soll platformunabhängig unter Verwendung bestehender etablierter Werkzeuge arbeiten
- Ist in Python geschrieben

Was ist T_EXToolKit?

Ein alternativer Weg zur Verwendung von ps-Code in pdfL^AT_EX

Kurz umrissen

- Ist entstanden aus der Notwendigkeit von L^AT_EX Satzanweisungen in eps Grafiken
- Soll ein universeller Mechanismus zur Erstellung technisch-wissenschaftlicher Arbeiten sein
- Beinhaltet eps Pre- und Mainprocessing und die Einbindung der Grafiken in das Hauptdokument
- Soll **plattformunabhängig unter Verwendung bestehender etablierter Werkzeuge arbeiten**
- Ist in Python geschrieben

Was ist T_EXToolKit?

Ein alternativer Weg zur Verwendung von ps-Code in pdfL^AT_EX

Kurz umrissen

- Ist entstanden aus der Notwendigkeit von L^AT_EX Satzanweisungen in eps Grafiken
- Soll ein universeller Mechanismus zur Erstellung technisch-wissenschaftlicher Arbeiten sein
- Beinhaltet eps Pre- und Mainprocessing und die Einbindung der Grafiken in das Hauptdokument
- Soll platformunabhängig unter Verwendung bestehender etablierter Werkzeuge arbeiten
- **Ist in Python geschrieben**

Wie funktioniert T_EXToolKit?

- **Preproc. Module:** Deren Aufgabe ist die Konvertierung von eps Grafiken aus Drittsystemen (z.B. AutoCAD) in ein psfrag taugliches eps Format. [ttk_acad2psfrag]
- Mainproc. Module: Erstellen jene pdf Dateien die ins Hauptdokument eingefügt werden. [ttk_eps2pdf und ttk_tex2pdf]
- Compile Vorgang: L^AT_EX muss jetzt so oft aufgerufen werden, bis alle Referenzen richtig gesetzt sind (dabei sind spezieller Anwendungen wie bibT_EX zu beachten).

Die Zuordnung der source Files erfolgt nach der Namensgebung der Dateien innerhalb einer vorgegebenen Verzeichnisstruktur.

Wie funktioniert T_EXToolKit?

- Preproc. Module: Deren Aufgabe ist die Konvertierung von eps Grafiken aus Drittsystemen (z.B. AutoCAD) in ein psfrag taugliches eps Format. [ttk_acad2psfrag]
- Mainproc. Module: Erstellen jene pdf Dateien die ins Hauptdokument eingefügt werden. [ttk_eps2pdf und ttk_tex2pdf]
- Compile Vorgang: L^AT_EX muss jetzt so oft aufgerufen werden, bis alle Referenzen richtig gesetzt sind (dabei sind spezieller Anwendungen wie bibT_EX zu beachten).

Die Zuordnung der source Files erfolgt nach der Namensgebung der Dateien innerhalb einer vorgegebenen Verzeichnisstruktur.

Wie funktioniert T_EXToolKit?

- Preproc. Module: Deren Aufgabe ist die Konvertierung von eps Grafiken aus Drittsystemen (z.B. AutoCAD) in ein psfrag taugliches eps Format. [ttk_acad2psfrag]
- Mainproc. Module: Erstellen jene pdf Dateien die ins Hauptdokument eingefügt werden. [ttk_eps2pdf und ttk_tex2pdf]
- **Compile Vorgang: L^AT_EX muss jetzt so oft aufgerufen werden, bis alle Referenzen richtig gesetzt sind (dabei sind spezieller Anwendungen wie bibT_EX zu beachten).**

Die Zuordnung der source Files erfolgt nach der Namensgebung der Dateien innerhalb einer vorgegebenen Verzeichnisstruktur.

Wie funktioniert T_EXToolKit?

- Preproc. Module: Deren Aufgabe ist die Konvertierung von eps Grafiken aus Drittsystemen (z.B. AutoCAD) in ein psfrag taugliches eps Format. [ttk_acad2psfrag]
- Mainproc. Module: Erstellen jene pdf Dateien die ins Hauptdokument eingefügt werden. [ttk_eps2pdf und ttk_tex2pdf]
- Compile Vorgang: L^AT_EX muss jetzt so oft aufgerufen werden, bis alle Referenzen richtig gesetzt sind (dabei sind spezieller Anwendungen wie bibT_EX zu beachten).

Die Zuordnung der source Files erfolgt nach der Namensgebung der Dateien innerhalb einer vorgegebenen Verzeichnisstruktur.

ttk_acad2psfrag

AutoCAD Konvertierung

- **Markiere in AutoCAD die Schrifteinfügestellen mit einer Linie auf definiertem Layer und exportiere die Grafik als eps nach `src_images`**
- Lage von Start zu Endpunkt zur Ausrichtung der Schrift, Länge der Linie könnte als Größenbezug verwendet werden
- Ersetze jenen Strich durch eine psfrag taugliche Textanweisung in jeder `*_acad.eps` und erzeuge daraus eine `*_img.eps`
- Text mit fortlaufenden Buchstaben kann nun mit psfrag im Mainprocessing ersetzt werden

ttk_acad2psfrag

AutoCAD Konvertierung

- Markiere in AutoCAD die Schrifteinfügestellen mit einer Linie auf definiertem Layer und exportiere die Grafik als eps nach `src_images`
- Lage von Start zu Endpunkt zur Ausrichtung der Schrift, Länge der Linie könnte als Größenbezug verwendet werden
- Ersetze jenen Strich durch eine psfrag taugliche Textanweisung in jeder `*_acad.eps` und erzeuge daraus eine `*_img.eps`
- Text mit fortlaufenden Buchstaben kann nun mit psfrag im Mainprocessing ersetzt werden

ttk_acad2psfrag

AutoCAD Konvertierung

- Markiere in AutoCAD die Schrifteinfügestellen mit einer Linie auf definiertem Layer und exportiere die Grafik als eps nach `src_images`
- Lage von Start zu Endpunkt zur Ausrichtung der Schrift, Länge der Linie könnte als Größenbezug verwendet werden
- Ersetze jenen Strich durch eine psfrag taugliche Textanweisung in jeder `*_acad.eps` und erzeuge daraus eine `*_img.eps`
- Text mit fortlaufenden Buchstaben kann nun mit psfrag im Mainprocessing ersetzt werden

ttk_acad2psfrag

AutoCAD Konvertierung

- Markiere in AutoCAD die Schrifteinfügestellen mit einer Linie auf definiertem Layer und exportiere die Grafik als eps nach `src_images`
- Lage von Start zu Endpunkt zur Ausrichtung der Schrift, Länge der Linie könnte als Größenbezug verwendet werden
- Ersetze jenen Strich durch eine psfrag taugliche Textanweisung in jeder `*_acad.eps` und erzeuge daraus eine `*_img.eps`
- **Text mit fortlaufenden Buchstaben kann nun mit psfrag im Mainprocessing ersetzt werden**

ttk_eps2pdf

eps zu pdf Konvertierung

- Suche in `src_images` nach `*_img.eps` und zugehöriger `*_img.tex` in der die `\psfrag` Anweisungen und die Größenangaben zu finden sind
- Führe die Anweisungen in `*_img.tex` aus und erzeuge in `images` die Datei `*.pdf`
- Diese Datei kann nun im L^AT_EX Dokument inkludiert werden

ttk_eps2pdf

eps zu pdf Konvertierung

- Suche in `src_images` nach `*_img.eps` und zugehöriger `*_img.tex` in der die `\psfrag` Anweisungen und die Größenangaben zu finden sind
- Führe die Anweisungen in `*_img.tex` aus und erzeuge in `images` die Datei `*.pdf`
- Diese Datei kann nun im L^AT_EX Dokument inkludiert werden

ttk_eps2pdf

eps zu pdf Konvertierung

- Suche in `src_images` nach `*_img.eps` und zugehöriger `*_img.tex` in der die `\psfrag` Anweisungen und die Größenangaben zu finden sind
- Führe die Anweisungen in `*_img.tex` aus und erzeuge in `images` die Datei `*.pdf`
- Diese Datei kann nun im L^AT_EX Dokument inkludiert werden

ttk_tex2pdf

tex (pstricks) zu pdf Konvertierung

- **Suche in src_images nach *_tex.tex**
- Führe die pstricks Anweisungen in *_tex.tex aus und erzeuge in images die Datei *.pdf
- Diese Datei kann nun im L^AT_EX Dokument inkludiert werden

ttk_tex2pdf

tex (pstricks) zu pdf Konvertierung

- Suche in `src_images` nach `*_tex.tex`
- Führe die `pstricks` Anweisungen in `*_tex.tex` aus und erzeuge in `images` die Datei `*.pdf`
- Diese Datei kann nun im \LaTeX Dokument inkludiert werden

ttk_tex2pdf

tex (pstricks) zu pdf Konvertierung

- Suche in `src_images` nach `*_tex.tex`
- Führe die pstricks Anweisungen in `*_tex.tex` aus und erzeuge in `images` die Datei `*.pdf`
- Diese Datei kann nun im L^AT_EX Dokument inkludiert werden

Der Übersetzungsvorgang

- **Führe die Preprocessing Module auf die entsprechenden Dateien aus.**
- Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.
- Führe \LaTeX aus.
- Wenn sich das `*.bib` file verändert hat muss auch `bibtex` ausgeführt werden.
- Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen
- Wenn sich Referenzen geändert haben, noch einml \LaTeX um diese aufzulösen

Der Übersetzungsvorgang

- Führe die Preprocessing Module auf die entsprechenden Dateien aus.
- **Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.**
- Führe \LaTeX aus.
- Wenn sich das `*.bib` file verändert hat muss auch `bibtex` ausgeführt werden.
- Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen
- Wenn sich Referenzen geändert haben, noch einml \LaTeX um diese aufzulösen

Der Übersetzungsvorgang

- Führe die Preprocessing Module auf die entsprechenden Dateien aus.
- Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.
- **Führe \LaTeX aus.**
- Wenn sich das `*.bib` file verändert hat muss auch `bibtex` ausgeführt werden.
- Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen
- Wenn sich Referenzen geändert haben, noch einml \LaTeX um diese aufzulösen

Der Übersetzungsvorgang

- Führe die Preprocessing Module auf die entsprechenden Dateien aus.
- Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.
- Führe \LaTeX aus.
- **Wenn sich das *.bib file verändert hat muss auch bibtex ausgeführt werden.**
- Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen
- Wenn sich Referenzen geändert haben, noch einm^l \LaTeX um diese aufzulösen

Der Übersetzungsvorgang

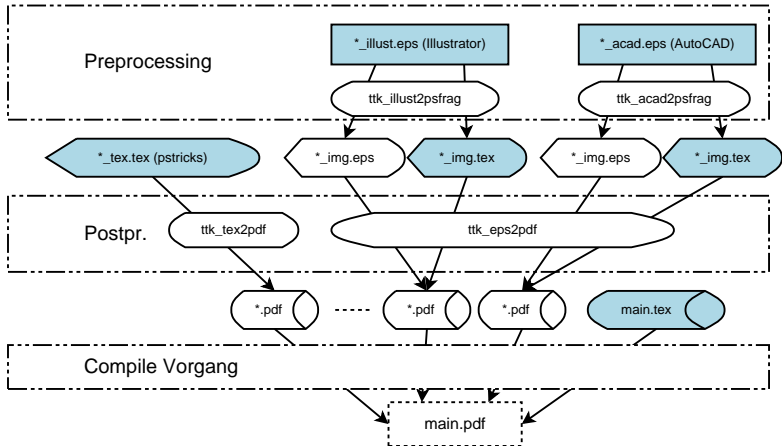
- Führe die Preprocessing Module auf die entsprechenden Dateien aus.
- Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.
- Führe \LaTeX aus.
- Wenn sich das `*.bib` file verändert hat muss auch `bibtex` ausgeführt werden.
- **Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen**
- Wenn sich Referenzen geändert haben, noch einml \LaTeX um diese aufzulösen

Der Übersetzungsvorgang

- Führe die Preprocessing Module auf die entsprechenden Dateien aus.
- Nun verarbeite die eps Dateien in `src_images` im Mainprocessing und erzeuge alle pdf Dateien in `images`.
- Führe \LaTeX aus.
- Wenn sich das `*.bib` file verändert hat muss auch `bibtex` ausgeführt werden.
- Wenn `bibtex` ausgeführt wurde, dann muss ein erneuter \LaTeX Aufruf erfolgen zum erzeugen der Referenzen
- **Wenn sich Referenzen geändert haben, noch einml \LaTeX um diese aufzulösen**

Übersicht

Die Funktionsstruktur der Toolchain



Und nun Live ...

... einfach zum Zuschauen und Fragen stellen!

Offene Aufgaben

Wir stehen erst am Anfang

- **Schreiben weiterer Preprocessing Module (z.B. für Illustrator)**
- Verbesserung des Make Prozesses (Einbindung von bestehenden L^AT_EX build Systemen wie mkl_AT_EX)
- Erstellen von Systempaketen (Gentoo ebuilds, debian Pakete, Windows Installer)
- Einbindung der Toolchain in L^AT_EX "IDE"s (Emacs - AU_CT_EX, VIM, Kile, T_EXnicCenter, WinEdt)

Offene Aufgaben

Wir stehen erst am Anfang

- Schreiben weiterer Preprocessing Module (z.B. für Illustrator)
- **Verbesserung des Make Prozesses (Einbindung von bestehenden L^AT_EX build Systemen wie mklatex)**
- Erstellen von Systempaketen (Gentoo ebuids, debian Pakete, Windows Installer)
- Einbindung der Toolchain in L^AT_EX "IDE"s (Emacs - AUCT_EX, VIM, Kile, T_EXnicCenter, WinEdt)

Offene Aufgaben

Wir stehen erst am Anfang

- Schreiben weiterer Preprocessing Module (z.B. für Illustrator)
- Verbesserung des Make Prozesses (Einbindung von bestehenden L^AT_EX build Systemen wie mkl_{at}ex)
- Erstellen von Systempaketen (Gentoo ebuilds, debian Pakete, Windows Installer)
- Einbindung der Toolchain in L^AT_EX "IDE"s (Emacs - AUCT_EX, VIM, Kile, T_EXnicCenter, WinEdt)

Offene Aufgaben

Wir stehen erst am Anfang

- Schreiben weiterer Preprocessing Module (z.B. für Illustrator)
- Verbesserung des Make Prozesses (Einbindung von bestehenden L^AT_EX build Systemen wie mkl_{at}ex)
- Erstellen von Systempaketen (Gentoo ebuilds, debian Pakete, Windows Installer)
- Einbindung der Toolchain in L^AT_EX "IDE"s (Emacs - AUCT_EX, VIM, Kile, T_EXnicCenter, WinEdt)