

TeX-Dokumente automatisiert mit GitHub und Travis-CI bauen

Daniel Krah

31. Oktober 2018



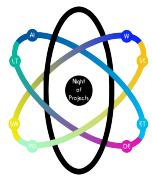
1 Warum \LaTeX bzw. LUAT \TeX ?

1.1 Vorteile von \LaTeX

- Tex Dateien kann man auch noch nach Jahren öffnen und in der Regel auch übersetzen.
- Keine Binärdateien. (Dokument auf Github direkt lesbar)
- Einfaches zusammenarbeiten (merge problemlos möglich)
- Nutzung mit Github und Travis CI relativ einfach

1.2 Warum dann LUAT \TeX ?

- Unterstützung von True-Type-Schriften
(KEIN Zugriff auf Systemschriften !!!)
- komplexeres Programmieren innerhalb von Tex möglich.



2 Warum GitHub und Travis-CI ?

Im Prinzip könnte man jeden x beliebigen Server ...

Dort:

ein Git Repo + ssh und ein CI System.

Man könnte eines selbst zu Hause hosten und hätte die volle Kontrolle.

ABER: Als Student möchte man vielleicht in Gruppen zusammen arbeiten um sich zusammen eine Mitschrift anzufertigen. Nicht jeder möchte sich eine komplette L^AT_EX- Installation an tun.

Och ne, iss mir zuviel Uffriss ...

Durfte ich mir schon oft anhören.



3 Github + Travis CI - the pdflatex way

3.1 Was wird benötigt ?

Kostenlose Variante (nur public Repo's):

- Ein Github - Account
- Ein Travis-CI (.org) Account
- Das Travis Command-line Tool

Kostenpflichtige / Studenten Variante

(Auch private Repo's / und sofortige build's):

- Ein kostenpflichtiger Github - Account (\$7/ Monat)
- Ein kostenpflichtiger Travis-CI (.com) Account (\$69/ Monat)
- Alternativ ein Student Developer Pack von Github
siehe: <https://education.github.com/pack>
- Das Travis Command-line Tool



3.2 Einrichtung

3.3 Github

Als erstes benötigen wir einen Github Account inkl. Repo.

The screenshot shows the GitHub sign-up interface. At the top, there are navigation links for Features, Business, Explore, Marketplace, and Pricing, along with a search bar and a 'Sign in or Sign up' button. The main heading 'Built for developers' is displayed, followed by a paragraph about GitHub's mission to support both open source and business needs. On the right, a sign-up form is shown with fields for Username (filled with 'NopTec'), Email (filled with 'JaChristophEsM@chtSinn.de'), and Password (filled with a masked password). Below the password field is a note: 'Use at least one letter, one numeral, and seven characters.' A large green 'Sign up for GitHub' button is at the bottom of the form. Small text at the very bottom states: 'By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.'



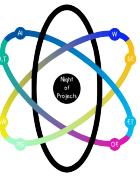
(Im Verlauf dieses Vortrages verwende ich:

<https://github.com/NOPtex/NOP>)

Dort befindet sich ein funktionierender Prototyp.
(Welcher aber noch ein paar zusätzlich Dateien beinhaltet.)

Prinzipiell reicht eine TeX-Datei und die .travis.yml.

pictures	Initial commit
.travis.yml	fix travis
README.md	Initial commit
basicff.sty	forgotten basicff
kap0.tex	Initial commit
minted.sty	Initial commit
nop.tex	Initial commit



3.4 Travis CI

3.4.1 In Travis CI einloggen und mit Github verbinden

The top screenshot shows the Travis CI homepage with a green banner 'Test and Deploy with Confidence' and a 'Sign Up' button. Below it is a list of repositories: 'green-eggs/ham' (22 builds), 'one-fish-two-fish' (266 builds), 'hep-enjep' (7001 builds), and 'harin-harizanbo' (209 builds). The bottom screenshot shows the 'Authorize Travis CI' page on GitHub, where the user grants permissions for Travis CI to access their repositories, webhooks, commit statuses, deployments, and organizations.

Da Travis nur mit Github funktioniert ist die Einrichtung recht trivial.

Travis benötigt einige Berechtigungen welche man in diesem Schritt erteilt.



3.4.2 Github - Repo aktivieren

The screenshot shows the initial setup page for Travis CI. It includes sections for activating GitHub repositories, adding a .travis.yml file, triggering a build with a git push, and a 'Sync account' button.

Activate GitHub Repositories
Once you're signed in, and we've initially synchronized your repositories from GitHub, go to your profile page for open source or for your private projects. You'll see all the organizations you're a member of and all the repositories you have access to. The ones you have administrative access to are the ones you can enable the service hook for. Flip the switch on for all repositories you'd like to enable.

Add .travis.yml file to your repository
In order for Travis CI to build your project, you need to tell the systems a little bit about it. You'll need to add a file named .travis.yml to the root of your repository. If .travis.yml is not in the repository, is misspelled or is not valid YAML, Travis CI will ignore it. Here you can find some of our basic language examples.

Trigger your first build with a git push
Once the GitHub hook is set up, push your commit that adds .travis.yml to your repository. That should add a build into one of the queues on Travis CI and your build will start as soon as one worker for your language is available. To start a build, perform one of the following:

- Commit and push something to your repository
- Go to your repository's settings page, click on "Webhooks & Services" on the left menu, choose "Travis CI" in the "Services", and use the "Test service" button.

<https://travis-ci.org/profile>

The screenshot shows the 'Sync account' step. A red arrow labeled '1.' points to the 'Sync account' button. Another red arrow labeled '2. Aktivieren' points to the 'Aktivieren' button below the repository list.

NOPtex
We're only showing your public repositories. You can find your private projects on travis-ci.com.

Repositories
NOPtex
NOPtex/NOP
NOPtex/NOP

Beta Features
See what's new!
Organizations
You are not currently a member of any organization.
Is an organization missing?
Review and add your authorized organizations.

Travis CI
©Travis CI, GmbH
Ritter Straße 8
11047 Berlin, Germany
Imprint
Jobs
Help
Documentation
Blog
Email
Twitter
Travis CI Status
Travis CI Status

Da Travis nur mit Github funktioniert ist die Einrichtung recht einfach.

Travis benötigt einige Berechtigungen welche man im nächsten Schritt erteilt.



3.4.3 Build-Einstellungen setzen



NOPtex/NOP

Über das Zahnrad kommt man zu den Einstellungen.

Ich setze alle Häkchen weil:

- nur releases Gebaut werden sollen
(Testen kann man lokal)
Alle anderen Build sollen abgebrochen werden.
- Nur wenn .travis.yml vorhanden ist Build starten.
(Einfaches deaktivieren über Git)



4 Das wars? ~ Nicht ganz!

4.1 Travis YAML erstellen/anpassen

Da die Kommunikation zwischen Travis und GitHub verschlüsselt abläuft, muss man sich einen eigenen Schlüssel erstellen.

Am einfachsten geht das mit dem Travis-Komandozeilenprogramm.
Doch zuerst sollte man sich die vorhandene .travis.yml sichern.

.travis.yml sichern

```
# sicherung der vorhandenen .travis.yml
mv .travis.yml backup.travis.yml
# falls schon vorhanden wird man gewarnt
travis setup releases
Detected repository as NOPtex/NOP, is this correct? |yes| yes
deploy section already exists in .travis.yml, run with --force to
- override
```



4.2 eine neue .travis.yml erstellen

Basis-Datei erstellen

```
# Travis Setup
travis setup releases
# natürlich ist keine .travis.yml da. Die haben wir ja umbenannt.
no .travis.yml found
# also erstellen wir eine leere:
touch .travis.yml
travis setup releases
Username: NOPtex
Password for NOPtex: *****
File to Upload: _build/nop.pdf
Deploy only from NOPtex/NOP? |yes| yes
Encrypt API key? |yes| yes
```



Danach wurde eine Datei mit folgendem Inhalt generiert:

Basis-Datei erstellen

deploy:

```
provider: releases
```

```
api_key:
```

```
  secure: <Hier könnte ihre Werbung stehen !!!>
```

```
file: _build/nop.pdf
```

```
on:
```

```
repo: NOPtex/NOP
```



Jetzt muss man denn Rest wieder einfügen:

Hinzufuegen der noetigen Aenderungen

```
sudo: required
dist: trusty
before_install:
  - sudo apt-get -qq update && sudo apt-get install -y
  - --no-install-recommends <TexLive, Pygments ... >
install:
script:
  - mkdir _build
  - pdflatex -shell-escape -interaction=nonstopmode -synctex=1
  - -halt-on-error -output-directory
    _build nop.tex
  - pdflatex -shell-escape -interaction=nonstopmode -synctex=1
  - -halt-on-error -output-directory
    _build nop.tex
```



Fortsetzung:

Hinzufuegen der nötigen Änderungen

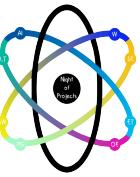
```
deploy:  
  provider: releases  
  api_key:  
    secure: <Hier könnte ihre Werbung stehen !!!>  
    file: _build/nop.pdf  
  skip_cleanup: true  
  on:  
    repo: NOPtex/NOP  
    tags: true  
    all_branches: true
```



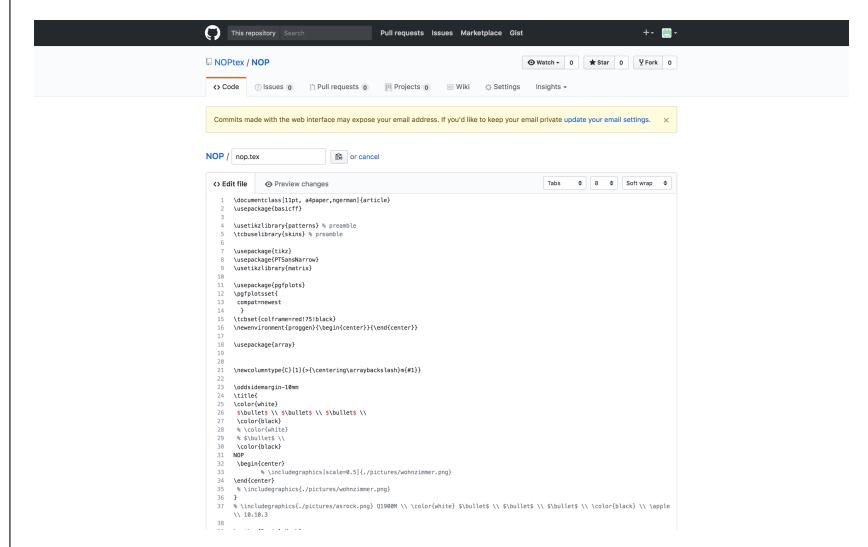
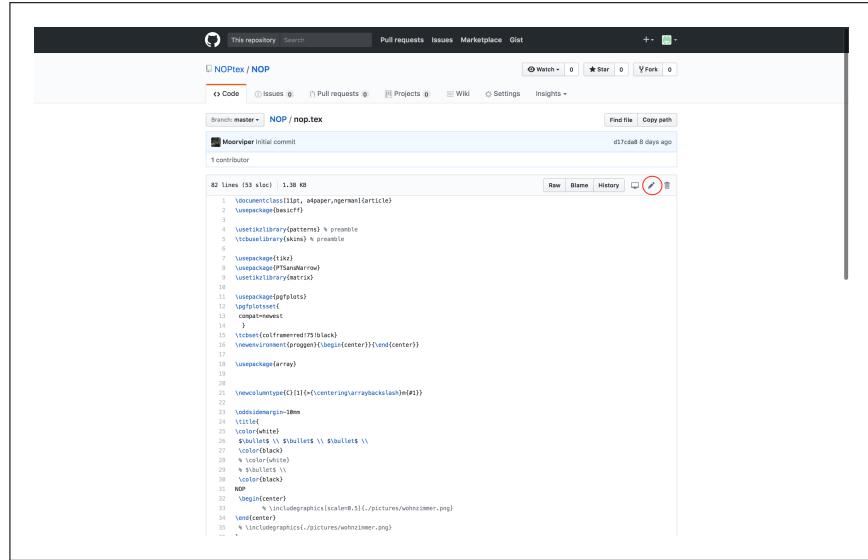
Deploy:

Hinzufuegen der nötigen Änderungen

```
# Benoetigt man nicht / Virtuelle Maschine fliegt dannach ja eh  
- in die Tonne ...  
  skip_cleanup: true  
  
# Wann soll deployed werden ?  
  tags: true  
  all_branches: true
```



5 Release mit Tag erstellen und bauen



Man klickt auf den Stift zum editieren der Datei.

Die Anzeige ändert sich geringfügig.



Text, Titel und Beschreibung hinzufügen.

```
26 % \section{converts}
27 % \end{center}
28 %
29 %
30 %
31 %
32 %
33 %
34 %
35 %
36 % \input(hardware.tex)
37 %
38 %
39 %
40 %
41 %
42 %
43 %
44 % To Do
45 % \input(todo.tex)
46 %
47 %
48 %
49 %
50 %
51 %
52 %
53 %
54 %
55 %
56 %
57 %
58 %
59 %
60 %
61 %
62 %
63 %
64 %
65 % \input(translate.tex)
66 %
67 %
68 %
69 %
70 %
71 Ein toller text
72 %
73 %
74 %
75 %
76 %
77 %
78 %
79 %
80 %
81 %
82 
```

Commit changes

ein toller text

Commit directly to the master branch.

Create a new branch for this commit and start a pull request. Learn more about pull requests.

Commit changes Cancel

Travis CI Blog Status Help

NOPtex / NOP build pending

My Repositories +

* NOPtex/NOP Duration: 5 sec

Current Branches Build History Pull Requests More options

0.8 ein toller text #7 started Running for 5 sec

Commit de8dacc Compare 0.8 Branch 0.8 NOPtex.authored GitHub committed

This job is running on our Trusty, [sudo: required] environment which was updated on Wednesday, June 21st. Read all about it on our blog and take note that you can add [group: deprecated=201702] in your .travis.yml file to use the previous version.

Job log View config

Build system information

Remove log Raw log

Kontrolle bei Travis und es baut.
(Das dauert dann so 7-8 Minuten.)



6 Erfolgreich gebaut

The top screenshot shows a failed build for the 'master' branch, while the bottom screenshot shows a successful build for the '0.8' branch.

Top Screenshot (Failed Build):

- Repository: NOPtex / NOP
- Status: build failing
- Current Branch: master
- Build #6 passed (8 minutes ago)
- Build #7 failed (8 minutes ago)
- Inactive Branches:
 - 0.8: build #7 passed (6 minutes ago)
 - 0.02: build #5 passed (8 days ago)
 - 0.001: build #3 failed (8 days ago)

Bottom Screenshot (Successful Build):

- Repository: NOPtex / NOP
- Status: build passing
- Current Branch: 0.8
- Build #7 passed (4 minutes ago)
- Commit de8dacc
- Compare 0.8
- Branch 0.8
- NOPtex authored
- Github committed
- This job ran on our Trusty environment which was updated on Wednesday, June 21st. Read all about it on our blog and take note that you can add `[group: deprecated=201702]` in your `.travis.yml` file to use the previous version.
- Job log (available):

```
Worker Information
Build system information
474 W: Failed to fetch https://packagecloud.io/basho/rabbitmq/ubuntu/dists/trusty/InRelease Failed to connect to packagecloud.io port 443: Connection timed out
475 W: Failed to fetch http://www.apache.org/dist/cassandra/debian/3.x/InRelease Could not connect to www.apache.org:80 (140.211.11.195), connection timed out [IP: 140.211.11.195:80]
476 W: Failed to fetch http://ppa.launchpad.net/chris-lea/redis-server/ubuntu/dists/trusty/InRelease Could not connect to ppa.launchpad.net:80 (91.189.95.83), connection timed out
477 W: Failed to fetch https://ppa.launchpad.net/guigui/redis55/ubuntu/dists/trusty/InRelease Unable to connect to ppa.launchpad.net:https
478 W: Failed to fetch https://packagecloud.io/computology/opt-backport/ubuntu/dists/trusty/InRelease Failed to connect to packagecloud.io port 443: Connection timed out
```

Wenn fertig sieht es so aus.

Optional kann man die Log-Ausgaben anschauen.
Aber in diesem Fall ging ja alles gut.



7 Github kontrollieren und PDF überprüfen

Auf GitHub ist nun auch die PDF unter Releases ...

PDF sieht gut aus und die Änderung ist mit drinnen.

```
1 Travis YAML
travis:
  node:
    - "node"
    - "npm"
  before_install:
    - "sudo apt-get update && sudo apt-get install -y --no-install-recommends
      - cm-super
      - texlive-latex-recommended
      - texlive-latex-extra
      - texlive-fonts-extra
      - texlive-latex-recommended
      - python3-pygments
      - texlive-lang-german
      - texlive-doc-de
      - texlive-texlive-lang-persian
      - texlive-doc-de
      - texlive-latex-extra
      - texlive-full
      - python3-pygments
      - python3-pygments
    "
  install:
    - "mkdir build"
    - "pushd build"
    - "texlive-build-shell-escape -interaction=nonstopmode -synctex=1 -halt-on-error
      - -output-directory build nop.tex
    "
    - "pdflatex -shell-escape -interaction=nonstopmode -synctex=1 -halt-on-error
      - -output-directory build nop.tex
    "
  deploy:
    - "git push"
    - "git push --tags"
  after_script:
    - "cp build/nop.pdf ."
  on:
    - tag: NOpv0.8
    - tag: true
    - all_branches: true
  env:
    - "CXX=gcc48g++7772784813300-CxxcsgplkldKthwvC70HhrigYzT3tgB0G1edHd6VbL1M9yua4Qmkk(21)af"
```

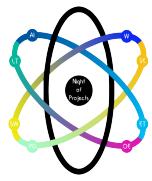


8 Github + Travis CI mittels arara + luatex

8.1 aktuelle Probleme

8.1.1 Linux

- Travis CI verwendet eine sehr alte Ubuntu-Version.
(Dadurch nur eine veraltete TexLive installation
⇒ hinderlich bei LuaT_EX)
- Aktuell verfügbare Docker-Container bekommen
das Texlive Update nicht hin.
- Arara bekommt generell keine Rechte um die Dateien zu schreiben.
(Evtl. lösbar, aber die Zeit ist grade knapp)



⇒ aktuelle Probleme

8.1.2 MacOS 10

- Unter OSX dauert die Latex Installation zu lange.
⇒ Travis CI bricht ab ...
- Docker-Nutzung schwierig, da man dafür Virtualbox + ein Linux benötigt.
- Brew (Ein Paketmanager für MacOS) enthält kein Tex-Live.
- Macports (nicht in den TravisCI - macOS10-Installationen enthalten)



Was benötigt es für arara:

```
% arara: lualatex: { shell: yes, action: nonstopmode, synctex:  
- yes}  
% arara: lualatex: { shell: yes, action: nonstopmode, synctex:  
- yes}  
\documentclass[landscape]{article}
```

Hinzufügen der nötigen Änderungen

```
- arara -v -o _build nop.tex
```



Wie gehts weiter / Nächste Schritte :

- Macports sauber auf macOS El Capitan installieren
- Texlive sauber installieren
- Rechte ordentlich setzen

Dann baut es hoffentlich