

# Getting more out of Git

Technologieplauscherl - 23.01.2018

# Johannes Pichler

- Lead Web Developer @ karriere.at
- Entwickler seit 2006
- Backend Stuff (PHP, Laravel, ...)

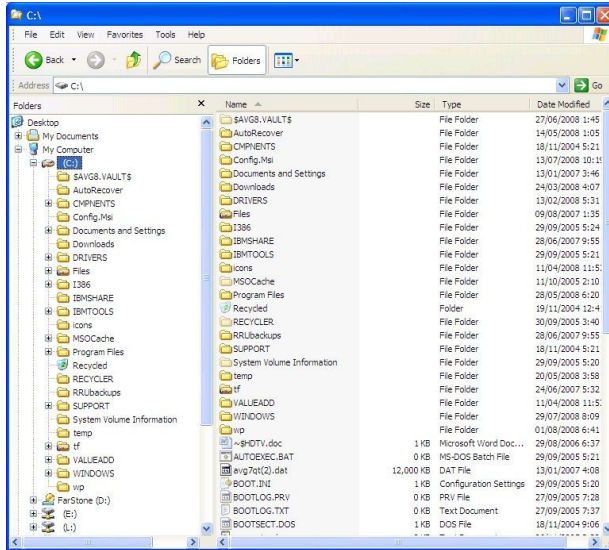


# Mein Weg zur Versionskontrolle

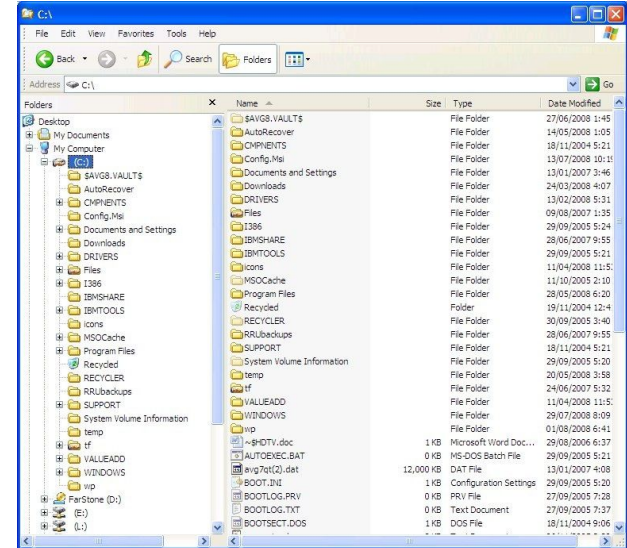


# Drag & Drop Deploys

Lokal



Produktivsystem



# Drag & Drop Deploys - Probleme

- Ständiges Überschreiben der Änderungen
- Keine Informationen über Änderungen
- Abhängigkeiten direkt im Projekt
- Manuelles Informieren der Kollegen bei größeren Änderungen

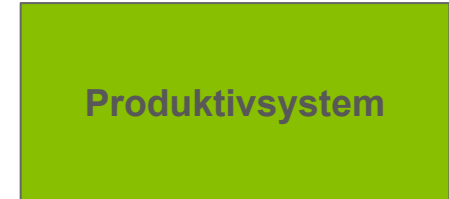
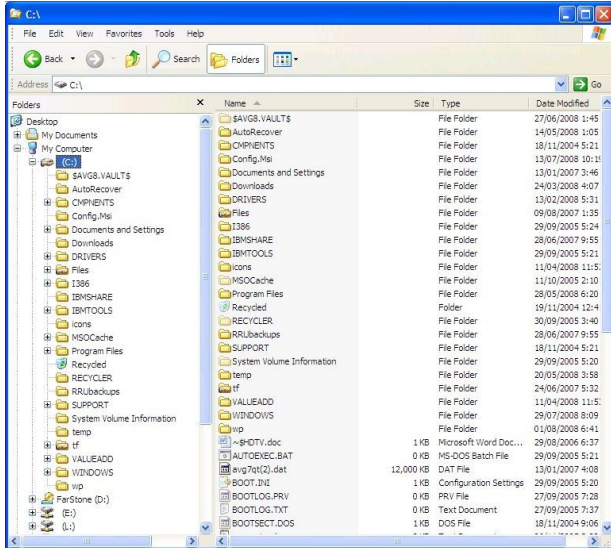
# Drag & Drop Deploys - Changelog

- Multiline Comment am Beginn von jedem File
- Jeder Dev trägt seine Änderungen mit Kürzel ein



# Subversion

Lokal





# Subversion - Probleme

- Kein Plan wozu
- Unregelmäßige Commits
- Verwendung als "Sicherheit" des Source Codes
- Deployment weiterhin manuell über Drag & Drop

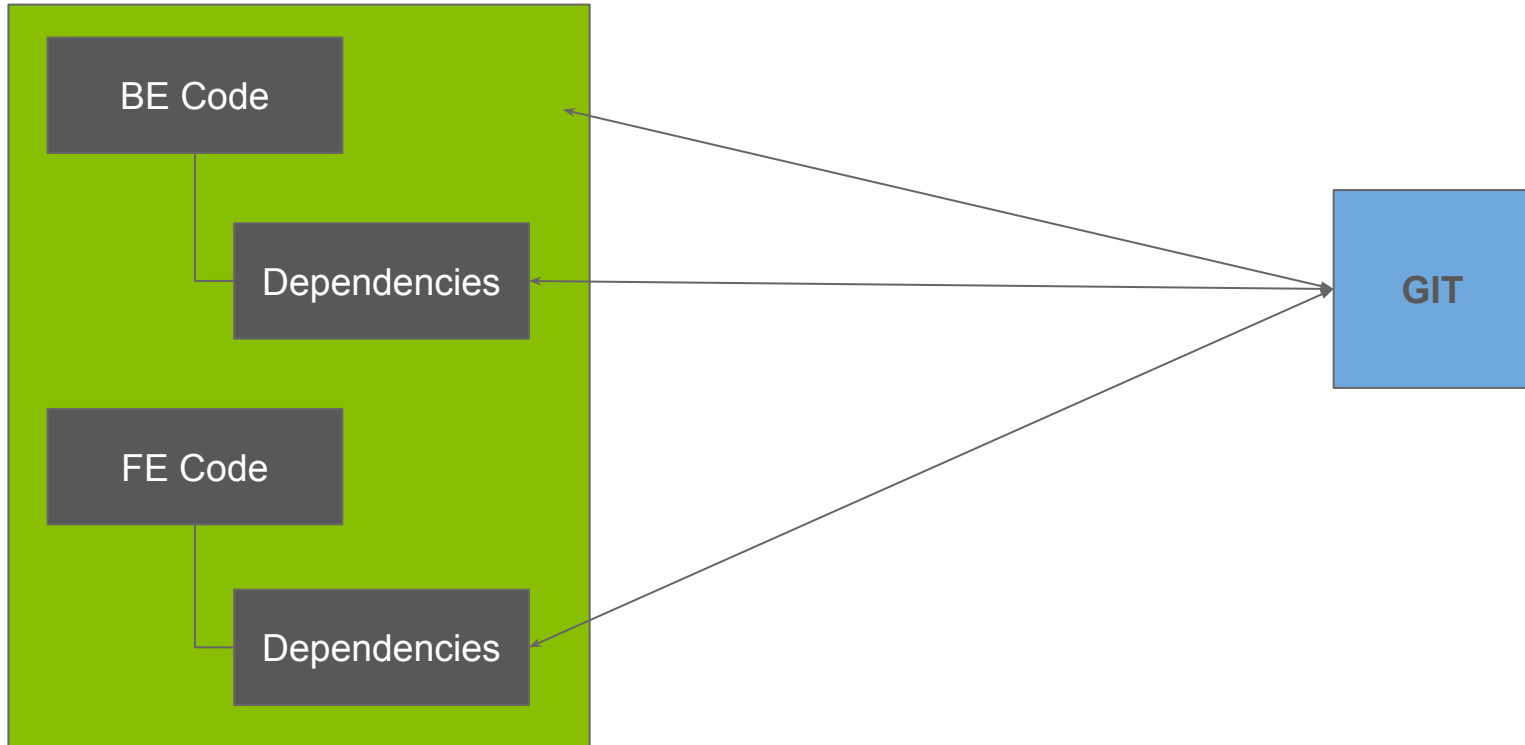
# Git



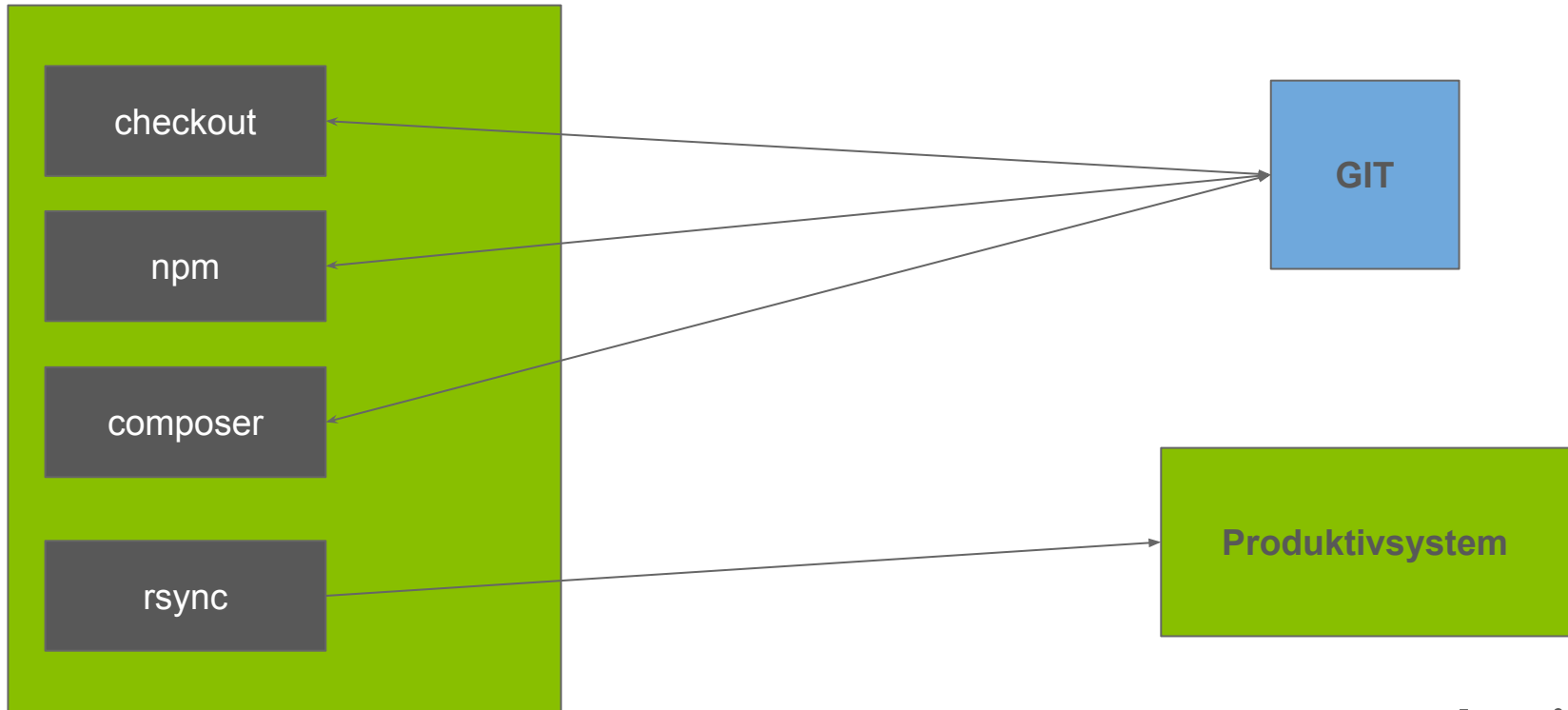
```
→ ~ git:(master) X git add .git*
→ ~ git:(master) X git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   .gitconfig
#       new file:   .gitignore_global
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
```

# Git & Deployment

# Ein Projekt bei karriere.at



# Deployment bei karriere.at





**MAKING GIT  
COMMITTS**



**FEELING MORE  
ORGANIZED**

# Subject

```
$ git log --oneline -5 --author cbeams --before "Fri  
e5f4b49 Re-adding ConfigurationPostProcessorTests a  
2db0f12 fixed two build-breaking issues: + reverted  
147709f Tweaks to package-info.java files  
22b25e0 Consolidated Util and MutableAnnotationUtil  
7f96f57 polishing
```

```
$ git log --oneline -5 --author pwebb --before "Sat  
5ba3db6 Fix failing CompositePropertySourceTests  
84564a0 Rework @PropertySource early parsing logic  
e142fd1 Add tests for ImportSelector meta-data  
887815f Update docbook dependency and generate epub  
ac8326d Polish mockito usage
```

**WHEN I SEE  
MY COMMIT HISTORY**





# Subject

- Maximale Länge von 50 Zeichen
- Starte mit einem Großbuchstaben
- Keine Satzzeichen am Ende des Subjects
- Verwende die Befehlsform

# Ticketnummern im Subject

```
KSWAT-1234: Refactor StaticDataService to use new...
```



```
Refactor StaticDataService to use new Resource API
```

```
Add reference to new Resource API to eliminate hardcoded data  
usage
```

```
Resolves: KSWAT-1234
```

# Typ im Subject

[BUGFIX] Fix a typo in sample module



**Fix** a typo in sample module

# Subject soll den Satz vervollständigen

If applied, this commit will ...

```
Fix a typo in sample module
```

```
Refactor StaticDataService to use new Resource API
```

# Body

- Zeilenumbruch nach 72 Zeichen
- Soll die Fragen What und Why beantworten
- Kann eine Ticketnummer beinhalten

# Beispiel

```
Refactor StaticDataService to use new Resource API
```

```
Add reference to new Resource API to eliminate hardcoded data  
usage.
```

```
Resolves: KSWAT-1234
```

# Message Template

```
# If this commit is applied, it will ...

# Why was this change made and what does it actually change?

# Ticket reference

# HINTS:
# Limit the subject line to 50 characters
# Capitalize the subject line
# Do not end the subject line with a punctuation mark
# Use the imperative mood in the subject
# Wrap the body at 72 characters
```

# Message Template

Verwendung über git config

```
git config --global commit.template ~/.git-commit-message
```



# VI Konfiguration

~/.vimrc

```
set textwidth=72  
set colorcolumn=+1  
set colorcolumn+=51
```

# BetterGitBot @ karriere.at

- Automatisiert über Gitlab Hook
- Es werden alle Formatierungsregeln geprüft
- Im Fehlerfall wird ein Fehlermeldung über Slack an den Dev gesendet

**MY FACE....**



**Memepile.com**

**WHEN SETTING THE HOOK.**

# Serverseitige Hooks

- pre-receive
- post-receive

# Clientseitige Hooks

- pre-commit
- commit-msg
- ...

# pre-commit

- wird direkt nach Aufruf von `git commit` ausgeführt
- wird zum Ausführen von Tests und Lintern verwendet

# Demo

# Zusammenfassung

- Commit Messages sind für die Ewigkeit
- Einheitliche Form hilft dir und deinen Kollegen
- Stehen nicht in Konkurrenz zum Ticket-System
  
- git cli benutzen



# Zusammenfassung

- Hooks können "Fix..." Commits reduzieren
- Tragen zur Code/Commit Qualität bei

# Weitere Informationen

- How to Write a Git Commit Message  
<https://chris.beams.io/posts/git-commit/>
- Git Hooks  
(<https://git-scm.com/book/gr/v2/Customizing-Git-Git-Hooks>)

Fragen?

# We are hiring



karriere.at

EINBLICKE

VIDEOS

## Web Backend Developer (m/w)

karriere.at · Linz am 11.1.2018

Aufgaben: Als Teil des Development-Teams bist du an der Weiterentwicklung unserer Recruiting-Portale beteiligt. Du arbeitest an wesentlichen Kernkomponenten und Services und stellst dich...

☆ Job merken



JOBS.AT

## Web Frontend Developer (m/w)

jobs.at Recruiting GmbH · Linz am 22.1.2018

Aufgaben: Du bist verantwortlich für das Frontend von jobs.at und hast den Mobile-first-Gedanken verinnerlicht. Du arbeitest eng mit dem ganzen Team zusammen, um unsere Vision umzusetzen. D...

☆ Job merken

# Danke

Johannes Pichler - @fetzi\_io