

# Automatisierte Backports

langweilige Routine, jetzt automatisch

# Wer bin ich?

Andreas Rogge

- hauptberuflicher Bareos-Entwickler
- Langzeit-Nutzer von Bareos
- ursprünglich Systemadministrator



# Übersicht



# Übersicht

- Worum geht es überhaupt?



# Übersicht

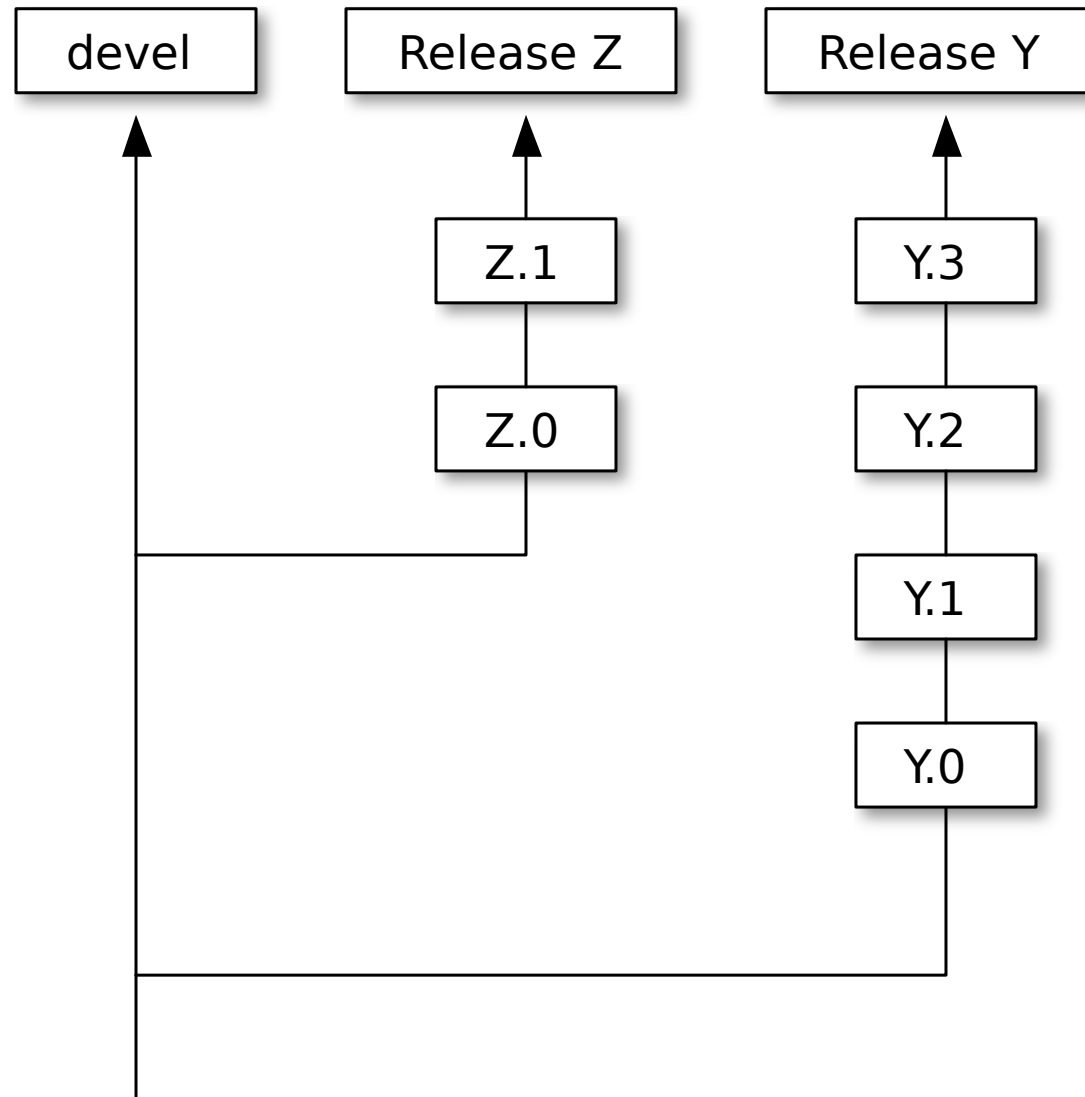
- Worum geht es überhaupt?
- Was muss man da machen?



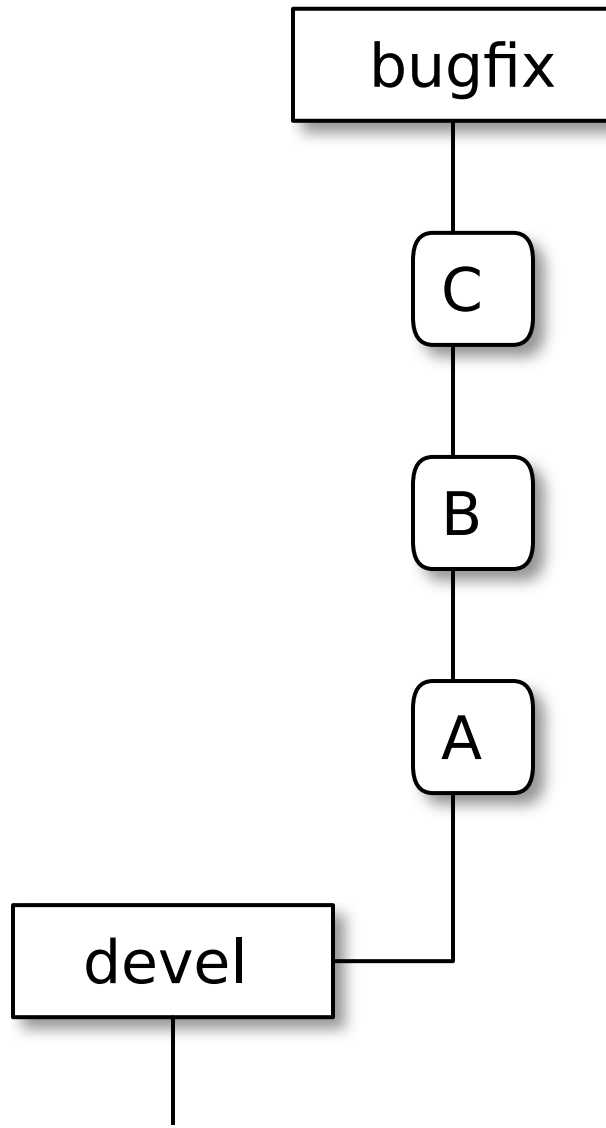
# Übersicht

- Worum geht es überhaupt?
- Was muss man da machen?
- Wie kann man das automatisieren?

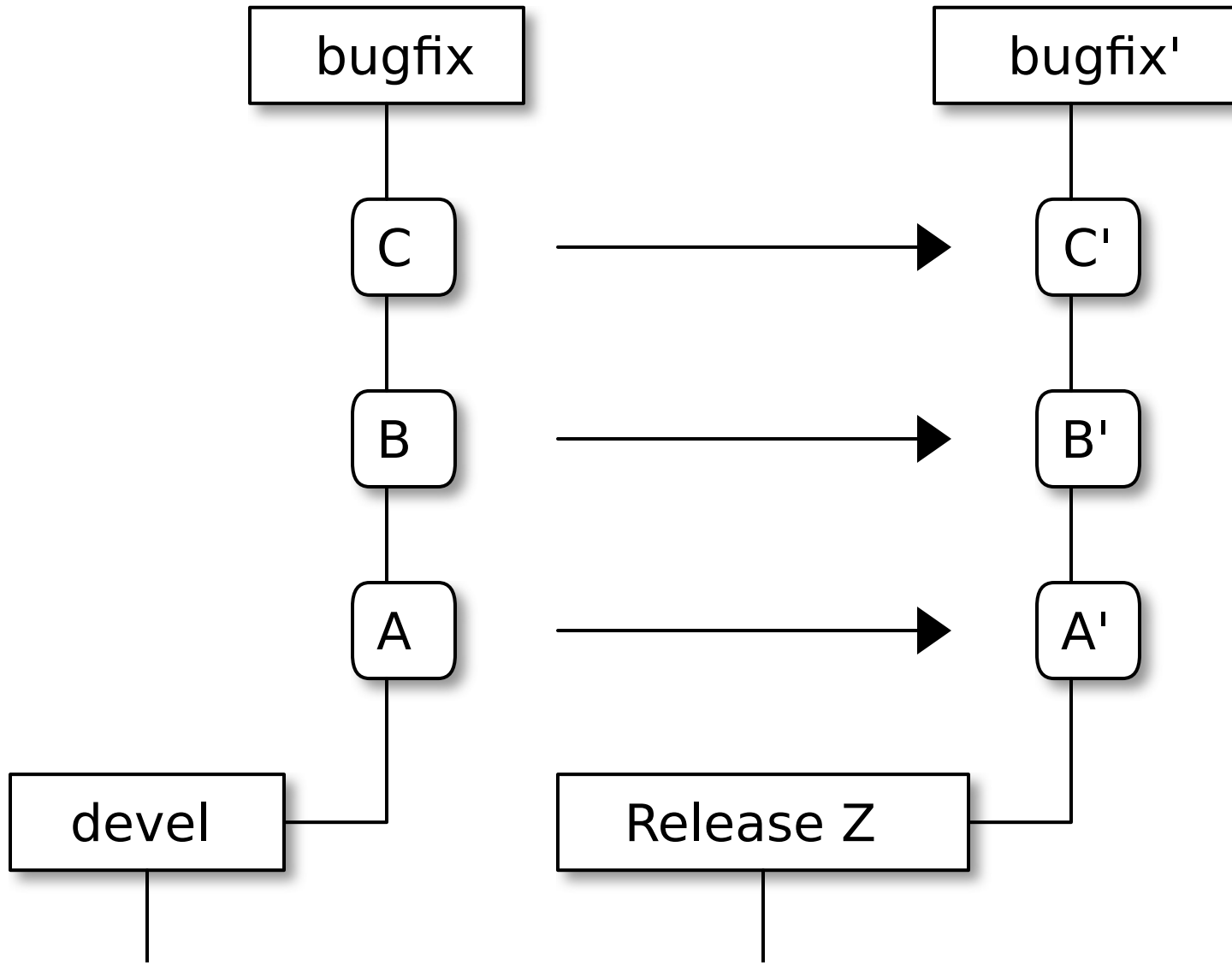
# Git Branches



# Pull Request



# Backport



# Branch vorbereiten

# Branch vorbereiten

- Liste der Commits bestimmen

# Branch vorbereiten

- Liste der Commits bestimmen
- Branch-Namen ausdenken

# Branch vorbereiten

- Liste der Commits bestimmen
- Branch-Namen ausdenken
- Backport-Branch erstellen/auschecken
  - auf Basis des richtigen Release-Branches

# Branch vorbereiten

- Liste der Commits bestimmen
- Branch-Namen ausdenken
- Backport-Branch erstellen/auschecken
  - auf Basis des richtigen Release-Branches
- `git cherry-pick -x <commit...>`

# Branch vorbereiten

- Liste der Commits bestimmen
- Branch-Namen ausdenken
- Backport-Branch erstellen/auschecken
  - auf Basis des richtigen Release-Branches
- `git cherry-pick -x <commit...>`
- Branch pushen

# Pull Request öffnen

# Pull Request öffnen

- korrekten Ziel-Branch setzen

# Pull Request öffnen

- korrekten Ziel-Branch setzen
- Labels vom Original-PR übernehmen

# Pull Request öffnen

- korrekten Ziel-Branch setzen
- Labels vom Original-PR übernehmen
- Titel und Beschreibung ausfüllen

# Automatisierung: Branch

# Automatisierung: Branch

- Ziel-Branch auschecken

# Automatisierung: Branch

- Ziel-Branch auschecken
- `pr-tool backport create <pr>`

# Automatisierung: Branch

- Ziel-Branch auschecken
- `pr-tool backport create <pr>`
  - erstellt Backport-Branch

# Automatisierung: Branch

- Ziel-Branch auschecken
- `pr-tool backport create <pr>`
  - erstellt Backport-Branch
  - bestimmt Commits aus Original-PR

# Automatisierung: Branch

- Ziel-Branch auschecken
- `pr-tool backport create <pr>`
  - erstellt Backport-Branch
  - bestimmt Commits aus Original-PR
  - bietet Liste von Commits über Editor an

# Automatisierung: Branch

- Ziel-Branch auschecken
- `pr-tool backport create <pr>`
  - erstellt Backport-Branch
  - bestimmt Commits aus Original-PR
  - bietet Liste von Commits über Editor an
  - führt `git cherry-pick` aus

# Automatisierung: PR

# Automatisierung: PR

```
pr-tool backport publish
```

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request
  - Übernimmt Titel/Labels des Original-PR

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request
  - Übernimmt Titel/Labels des Original-PR
  - Wählt den korrekten Ziel-Branch

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request
  - Übernimmt Titel/Labels des Original-PR
  - Wählt den korrekten Ziel-Branch
  - vorausgefülltes Backport-Template im Editor

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request
  - Übernimmt Titel/Labels des Original-PR
  - Wählt den korrekten Ziel-Branch
  - vorausgefülltes Backport-Template im Editor
- `git push`

# Automatisierung: PR

```
pr-tool backport publish
```

- Vorbereitung Pull Request
  - Übernimmt Titel/Labels des Original-PR
  - Wählt den korrekten Ziel-Branch
  - vorausgefülltes Backport-Template im Editor
- `git push`
- Erstellt Pull Request

# Ich will das auch!

[https://github.com/bareos/bareos/blob/master/  
devtools/pip-tools/pr\\_tool/backport.py](https://github.com/bareos/bareos/blob/master/devtools/pip-tools/pr_tool/backport.py)