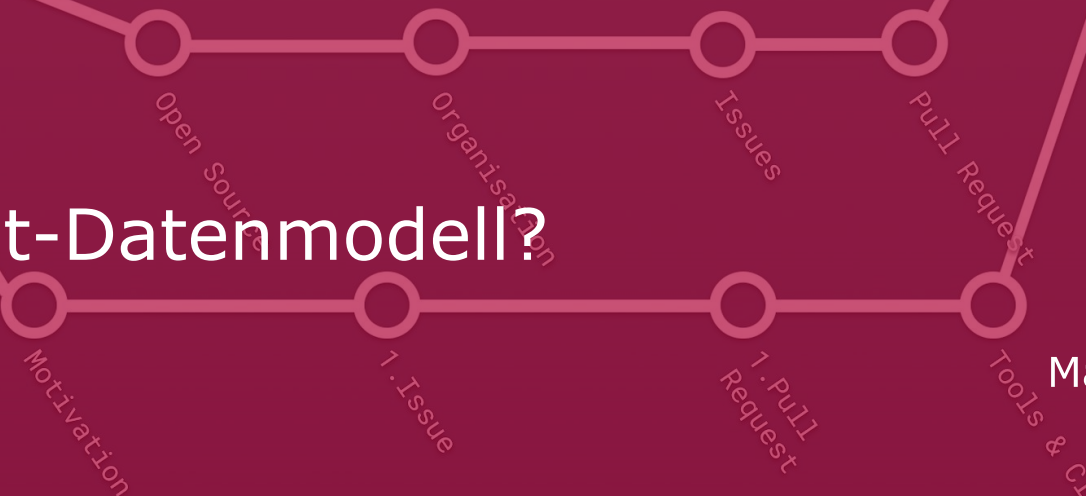


Was ist das Git-Datenmodell?

open_source

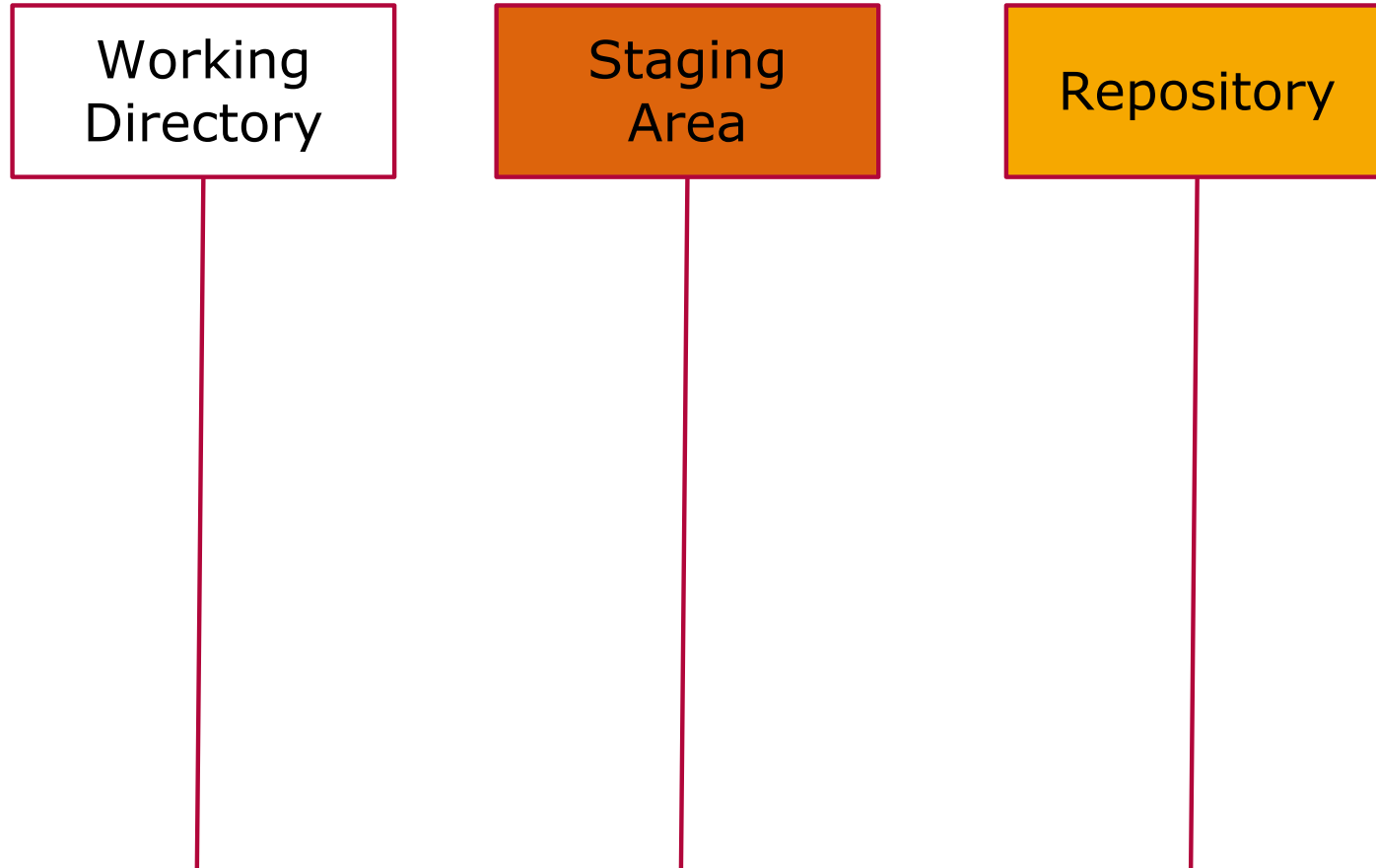
beitragen



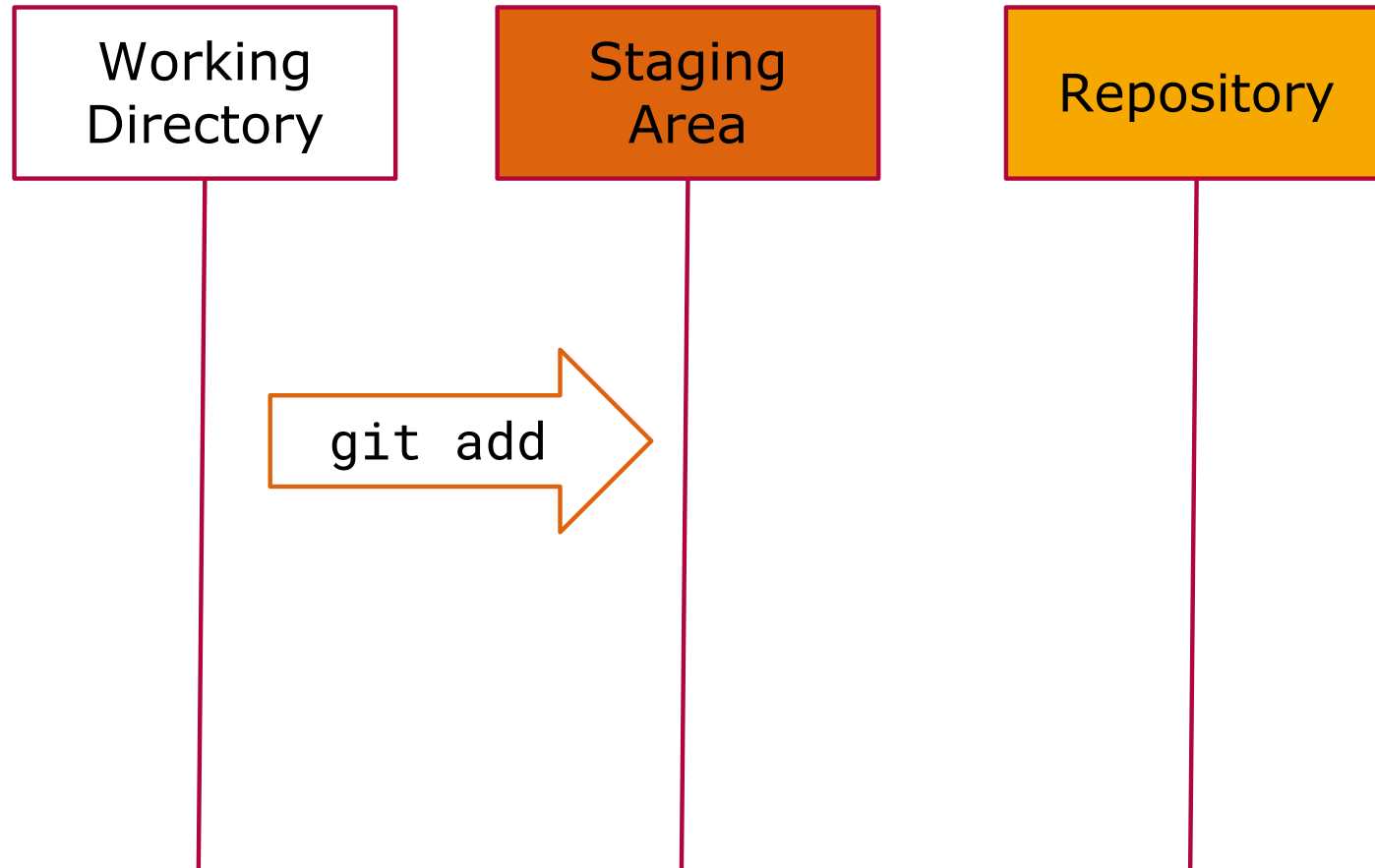
Marc Rosenau

Leo Wendt

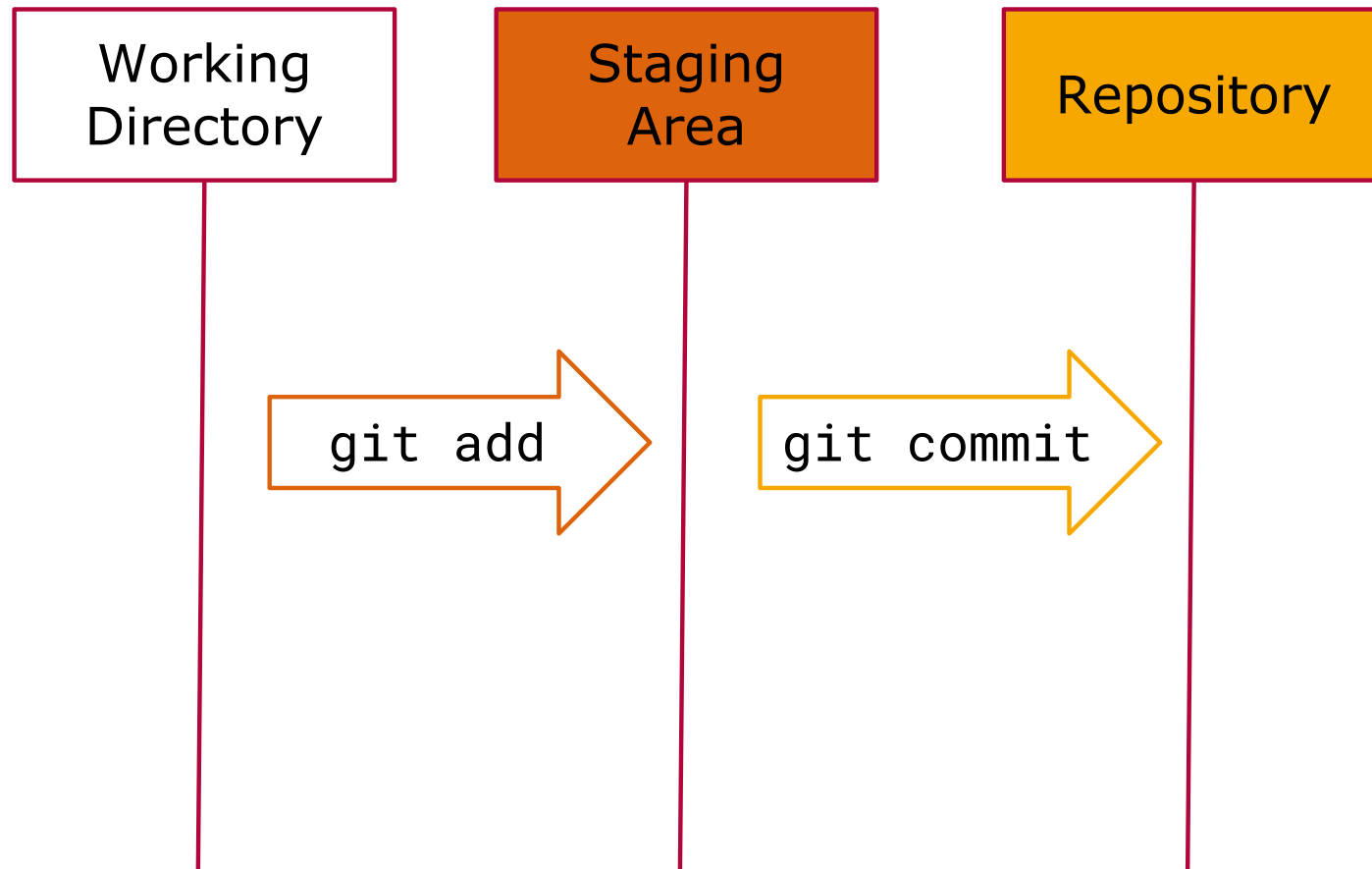
Recap



Recap



Recap



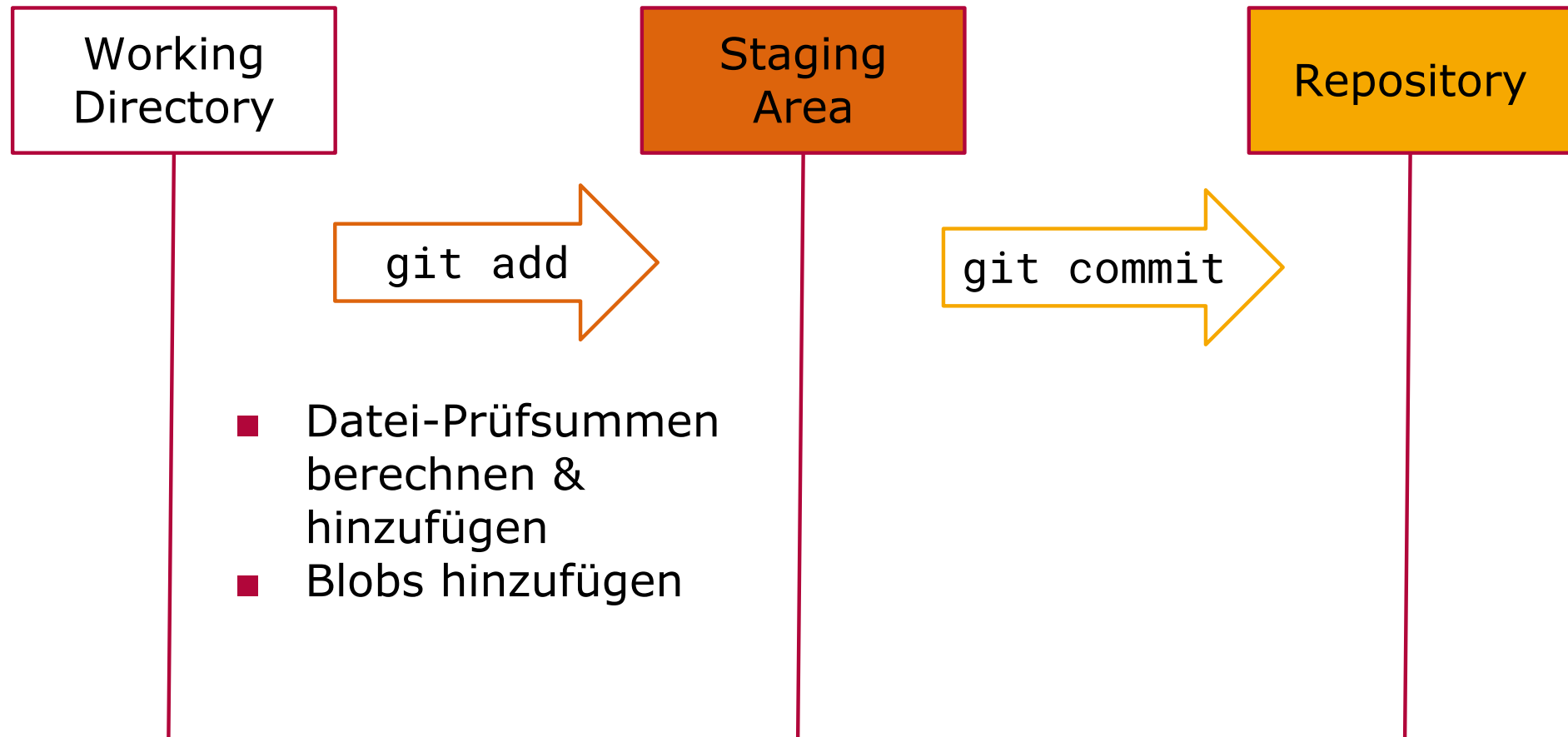
Git Baum und Datenmodell

- Stagen von Dateien
 - ☐ **speichert** die Version der Datei im Git Repository ("blobs")
 - ☐ **berechnet** Prüfsumme für jede Datei
 - ☐ fügt Prüfsummen zur Staging Area hinzu



```
$ git add liste1.txt liste2.txt
```

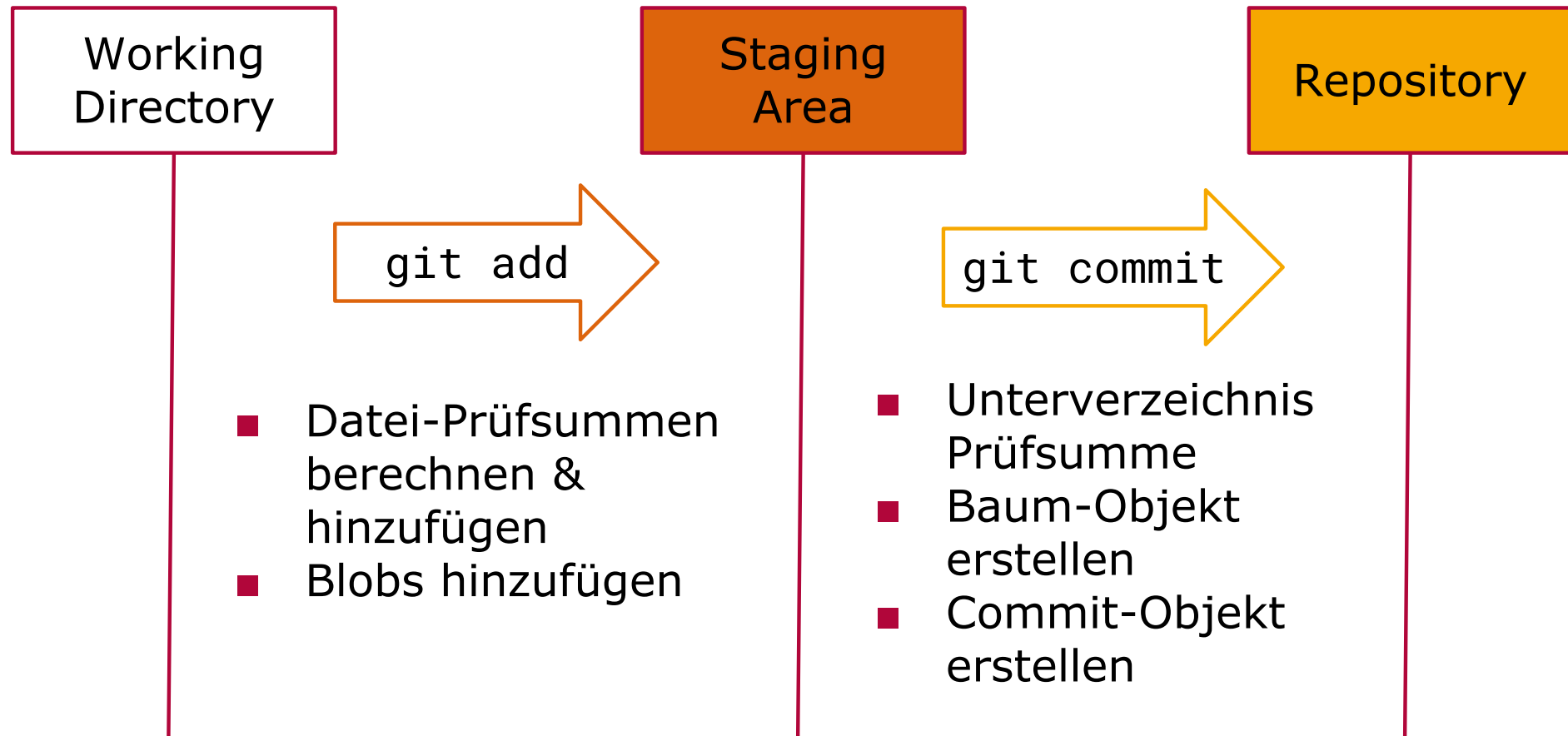
Recap



Git Baum und Datenmodell

- Committen von Dateien
 - Git erstellt Prüfsummen für Unterverzeichnisse und speichert sie als **Baum-Objekt** mit Zeigern auf gespeicherte Dateiversion
 - Git erstellt ein **Commit-Objekt** mit Metadaten und einen Zeiger auf das **Baum-Objekt**
 - Metadaten sind u.a. Namen und Email-Adresse, Commit-Nachricht und Zeiger zu den Commits, die direkt davor kamen

```
> $ git commit -m 'Die ersten 2 Listen hinzugefügt'
```



Liste1.txt

```
# Liste 1  
  
Hier sammeln wir  
Geschenke
```

Liste2.txt

```
# Liste 2  
  
Hier sammeln wir  
Rezepte
```

blob size

Liste 1

Hier sammeln wir
Geschenke

blob size

Liste 2

Hier sammeln wir
Rezepte

3b896

blob size

Liste 1

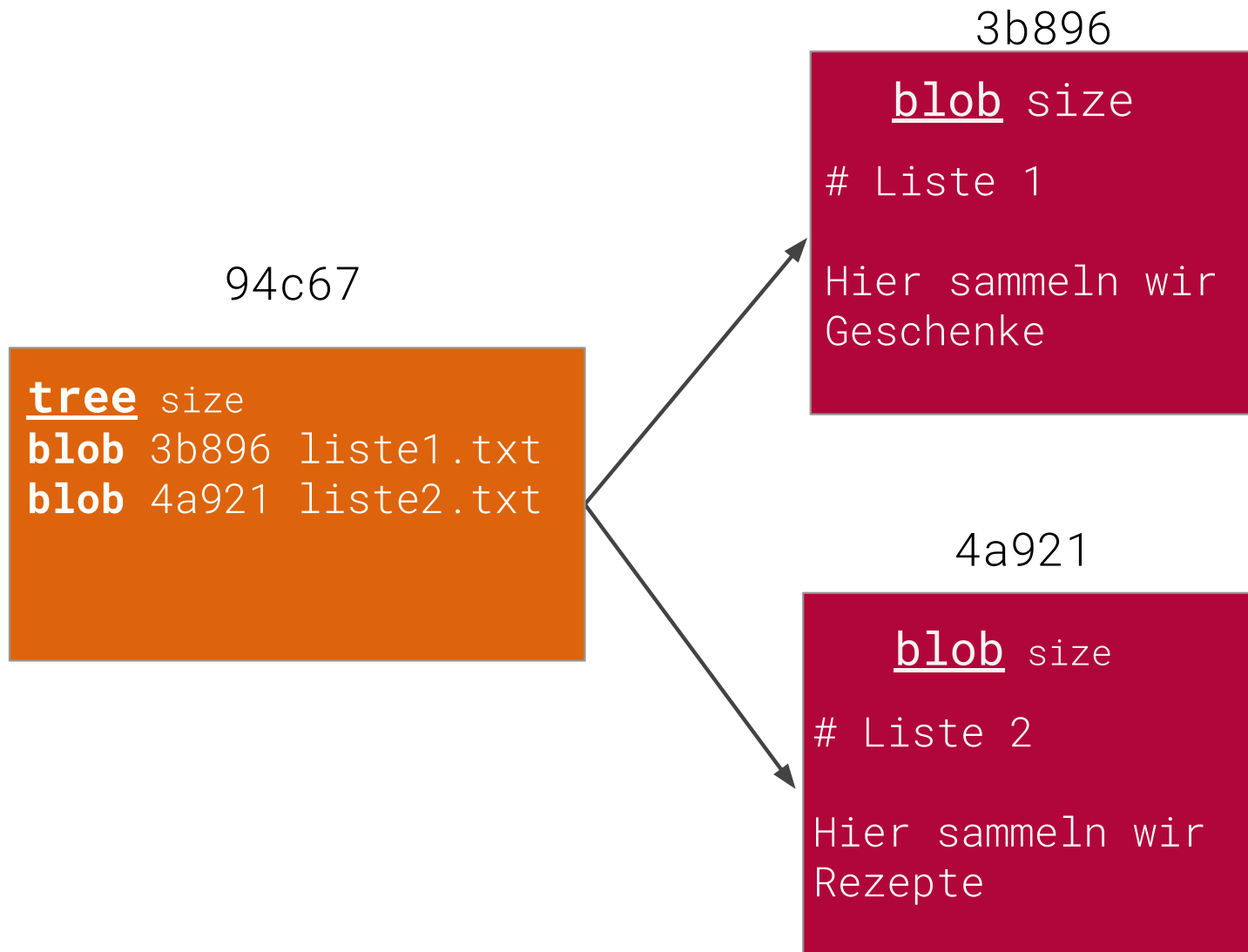
Hier sammeln wir
Geschenke

4a921

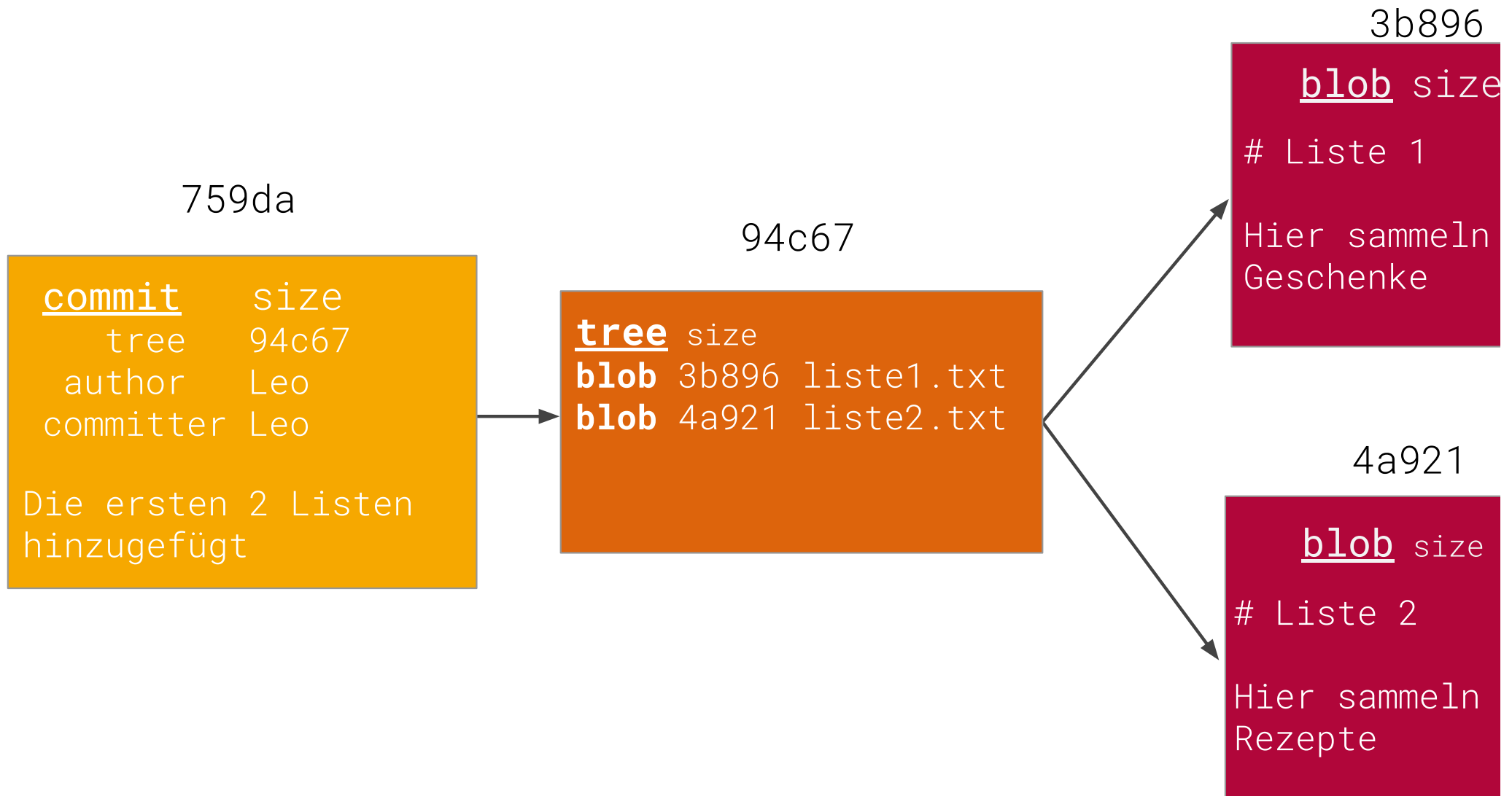
blob size

Liste 2

Hier sammeln wir
Rezepte



Beispiel



Beispiel mehrere Commits

759da

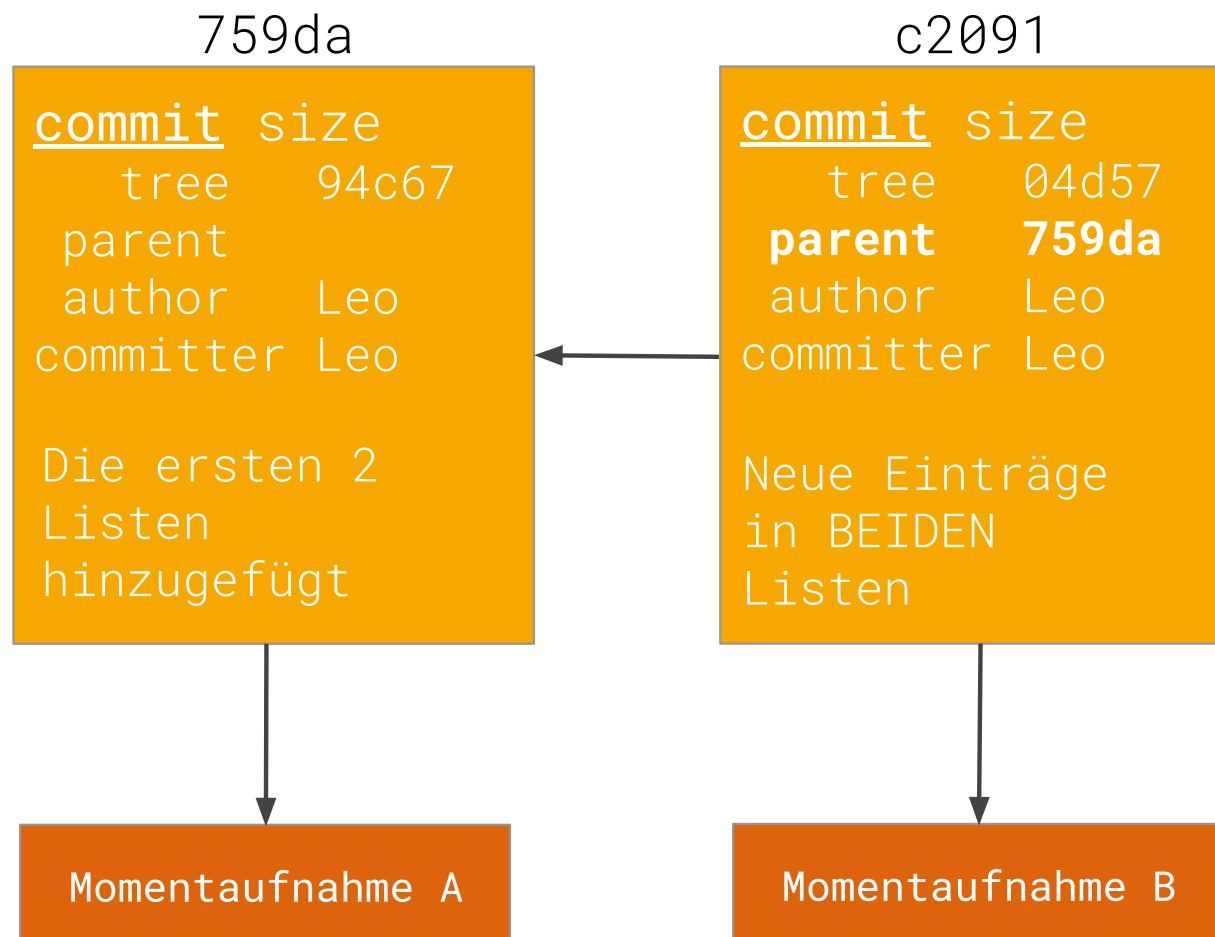
```
commit size
  tree    94c67
parent
  author   Leo
  committer Leo
```

Die ersten 2
Listen
hinzugefügt

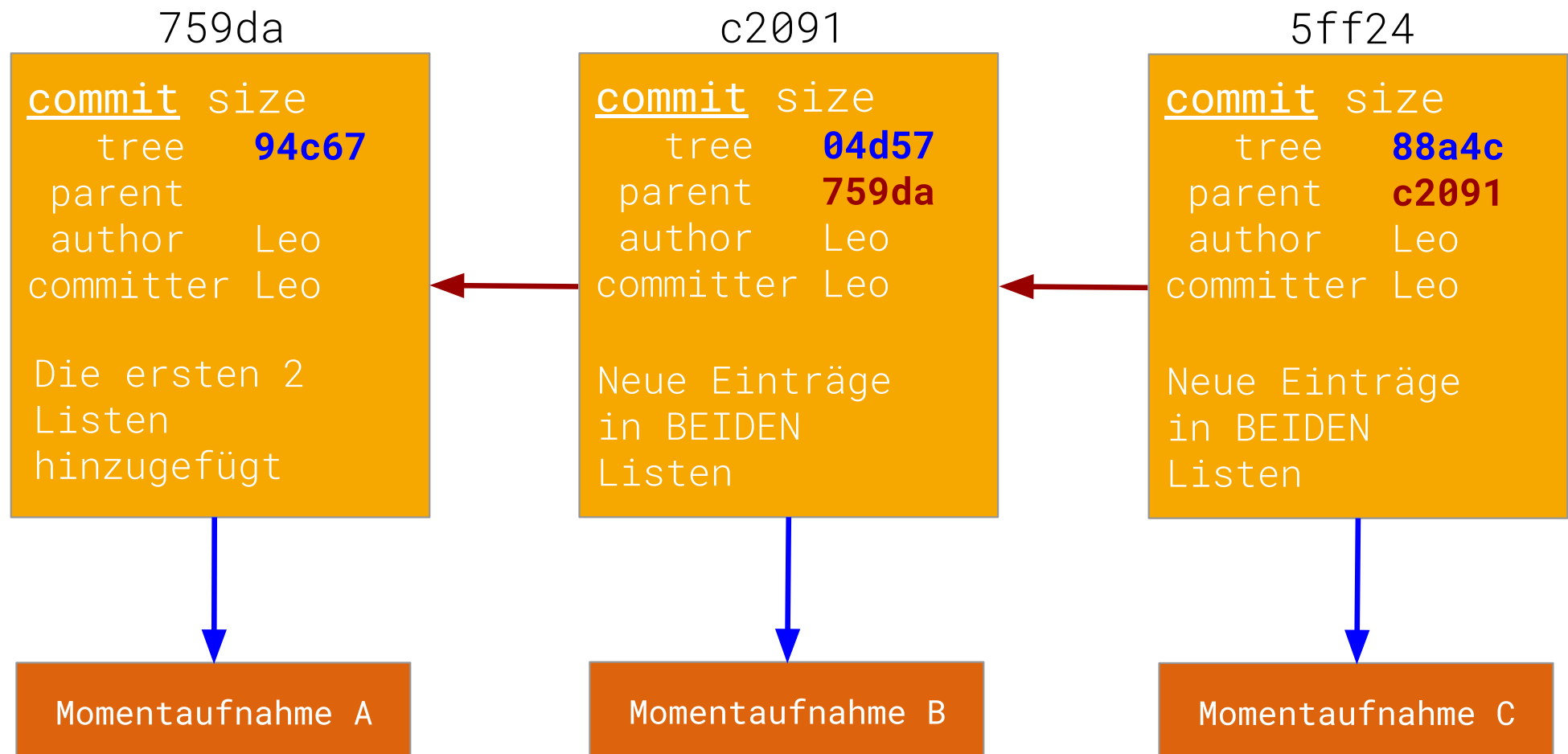


Momentaufnahme A

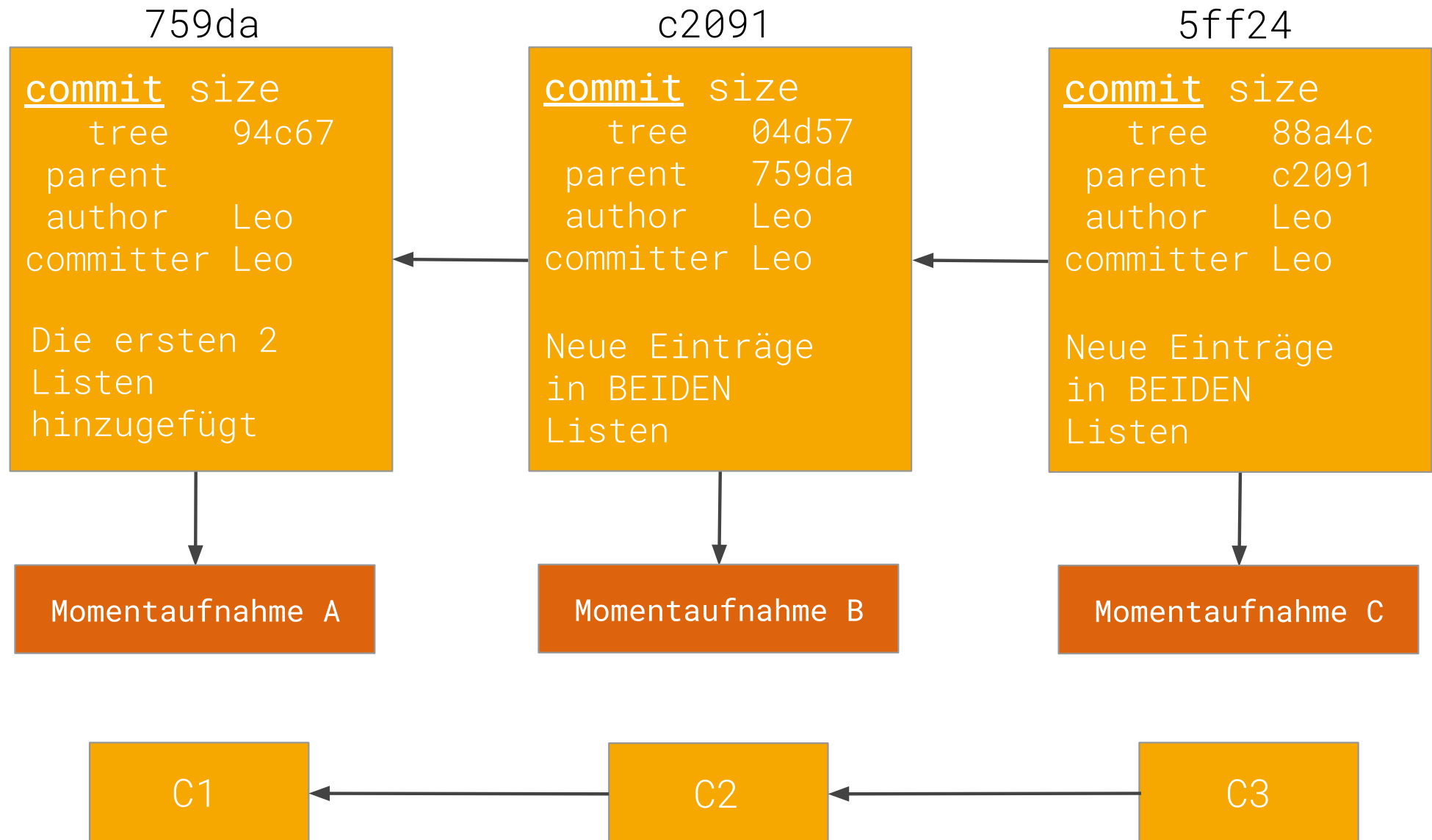
Beispiel mehrere Commits

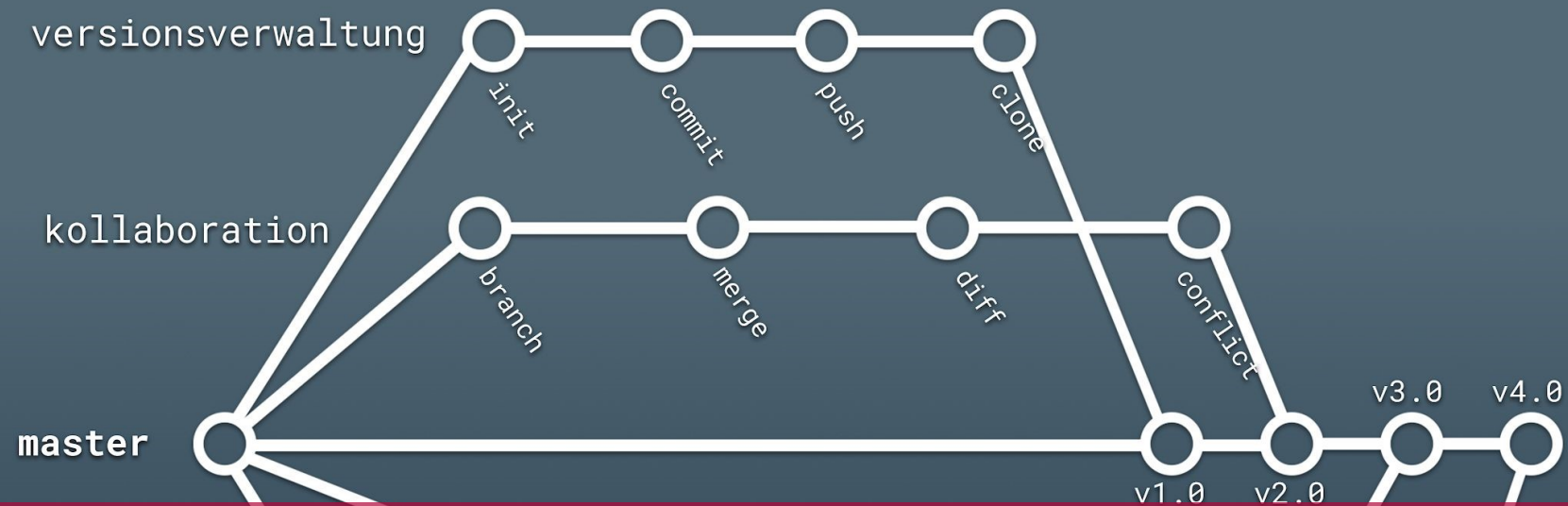


Beispiel mehrere Commits



Beispiel mehrere Commits

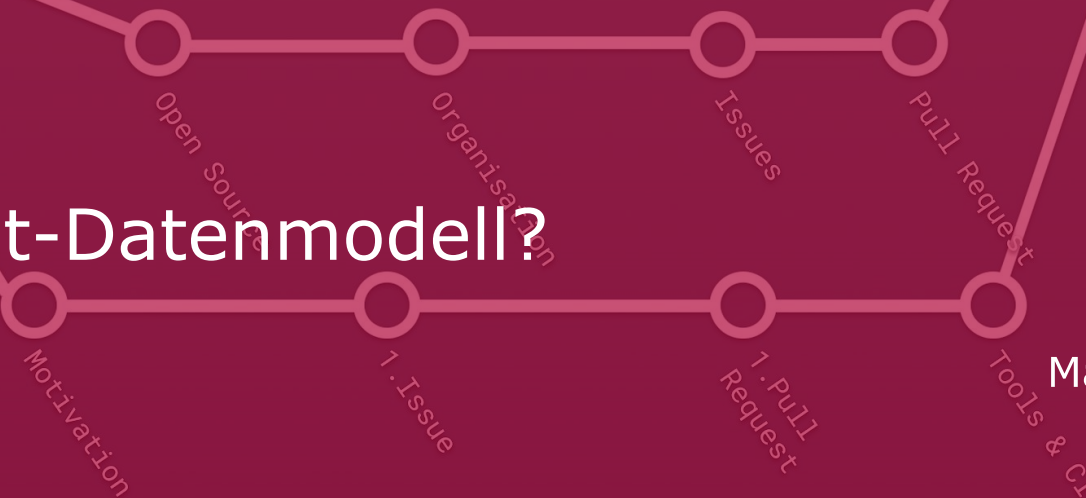




Was ist das Git-Datenmodell?

open_source

beitragen



Marc Rosenau

Leo Wendt