



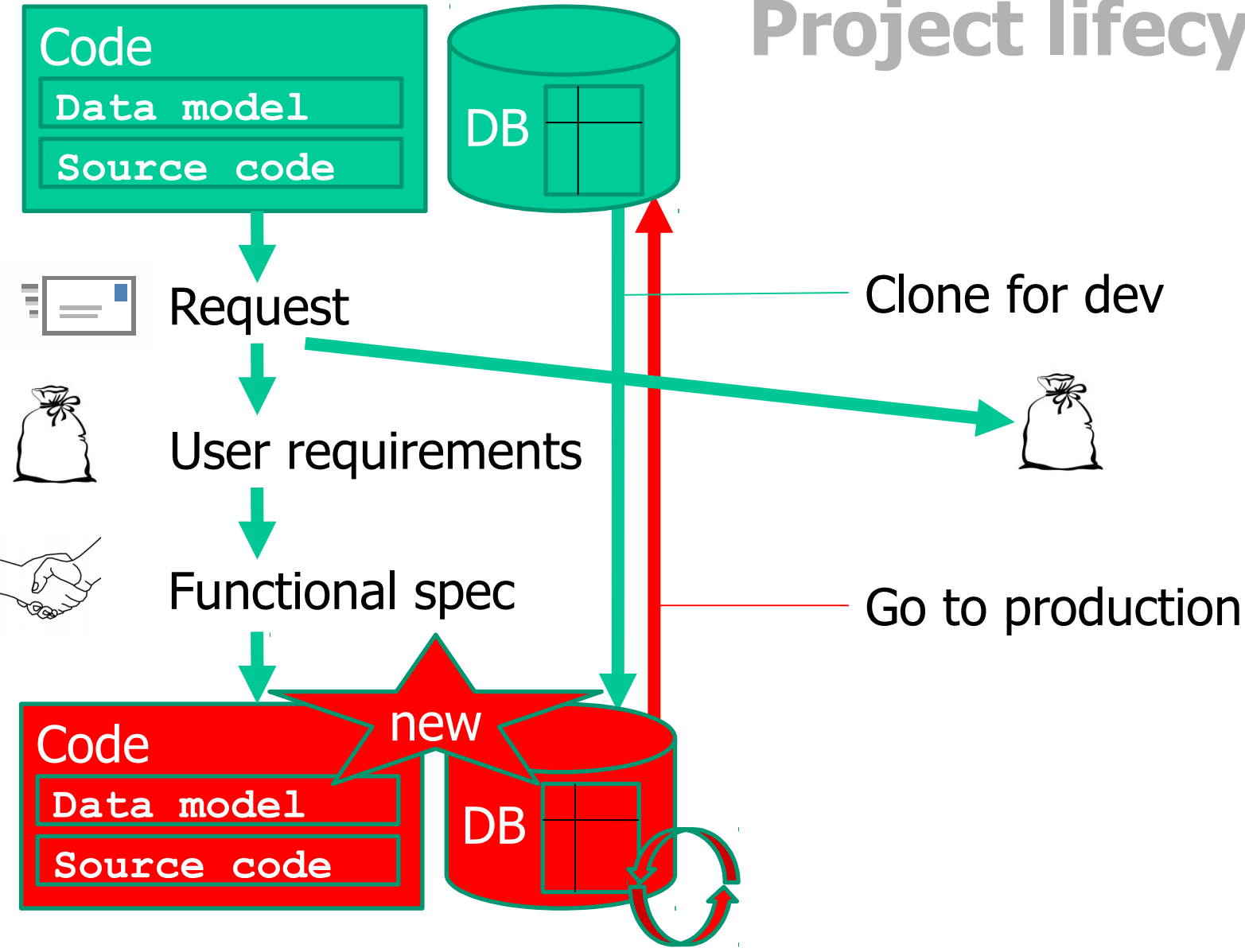
GIT

Mag. Thomas Griesmayer

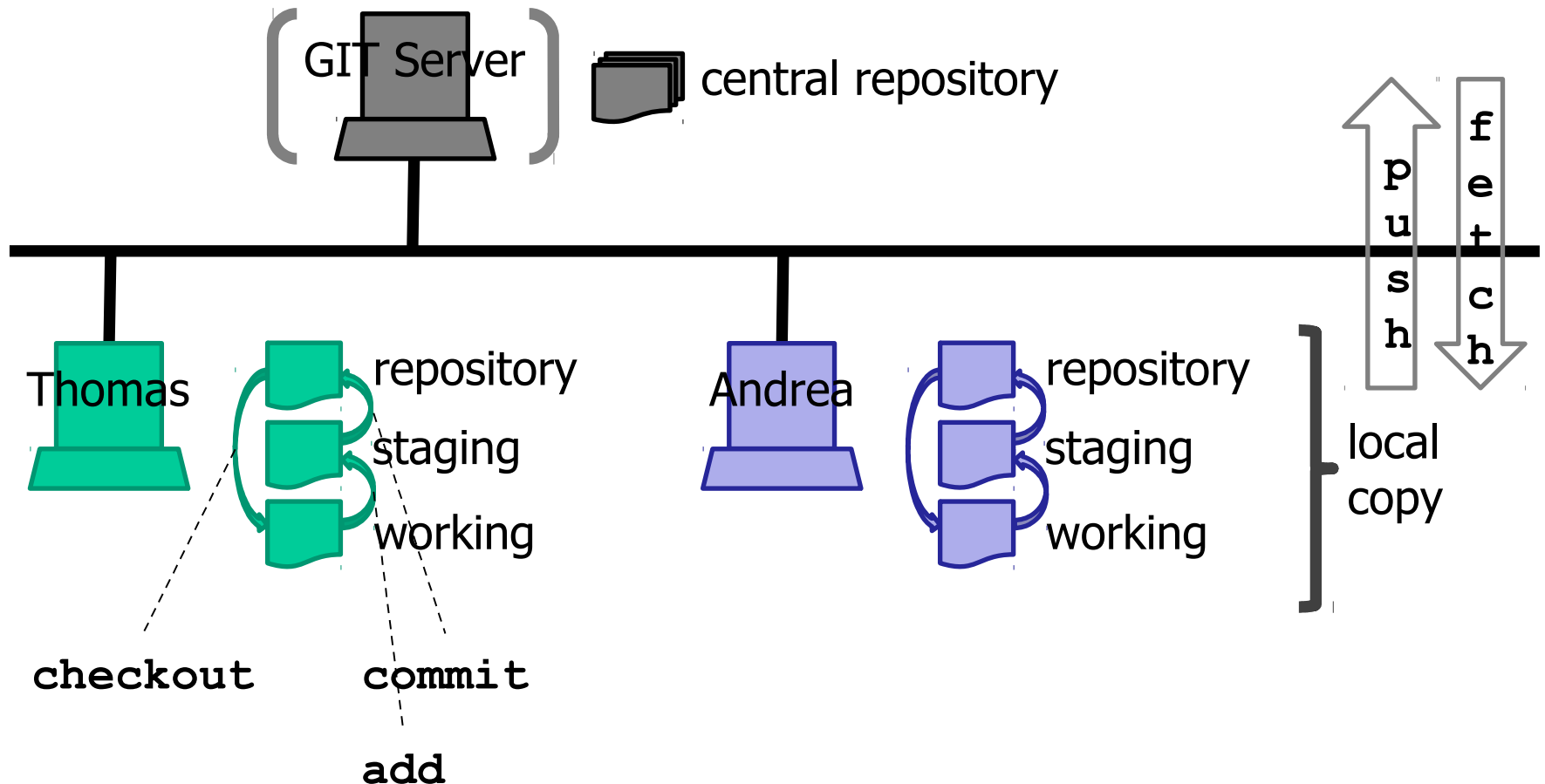
Definition

- Git was initially designed and developed in 2005 for Linux kernel development.
- It is used for software development projects as a source code management system.
- The software is independent of network access and there is no need of a central server.

Project lifecycle



Architecture



changeset

- GIT is working with the basic unit called a changeset.
- Most of the GIT commands are operations on changesets (not on files).
- Every GIT changeset is identified by a unique SHA-1 reference.

```
$ git log
```

```
commit f700508c7fc0b3d72736d6a009440908
```

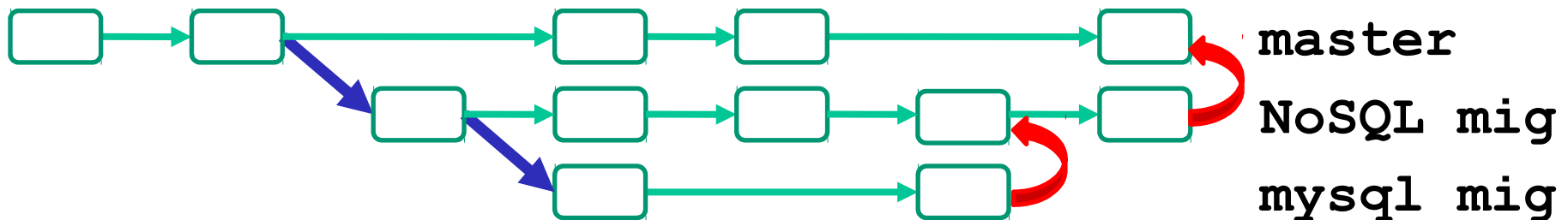
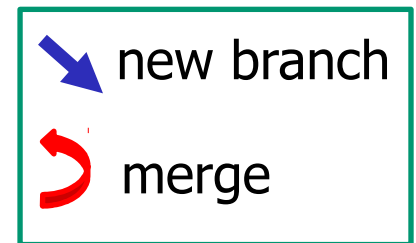
```
Author: Thomas <thomas@me.com>
```

```
Date: Fri Mar 11 13:51:31
```

Add collection

branch

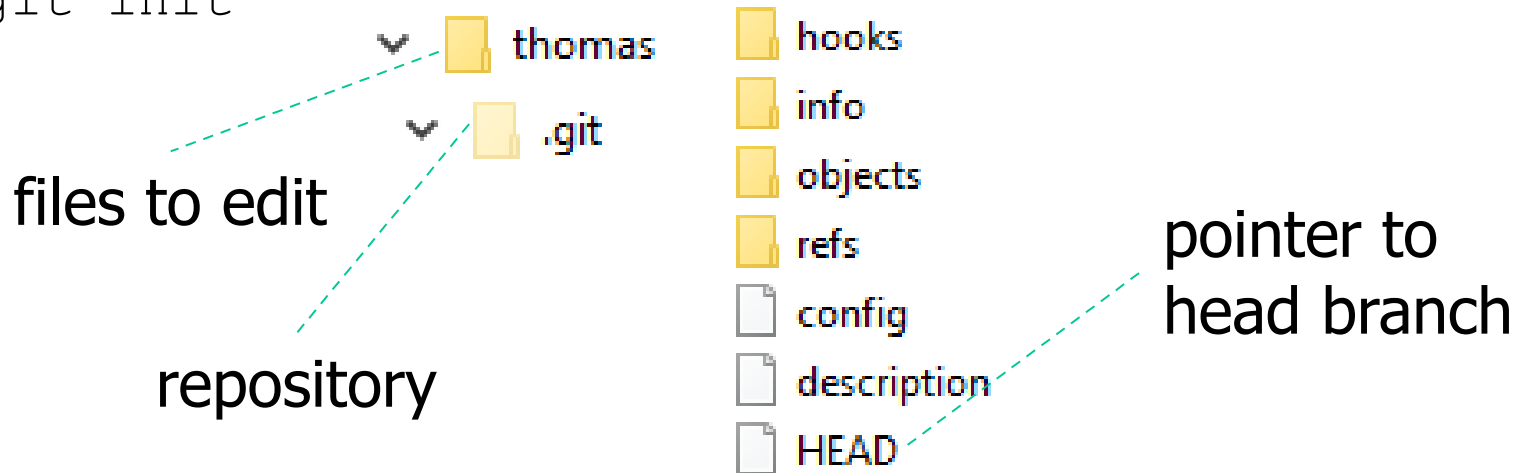
- A branch is an independent forked line of a changeset.
- The name of the default branch in GIT is **master**.
- Use the branch command to:
 - create
 - list
 - rename
 - delete **branches.**



git init

- The command `git init` creates an empty GIT repository or reinitialize an existing one.
- To use a specific path the `$GIT_DIR` environment variable has to be set.

```
$ cd H:\\Code_GIT\\thomas
$ git config --global user.name "Thomas"
$ git config --global user.email thomas@me.com
$ git init
```



local project

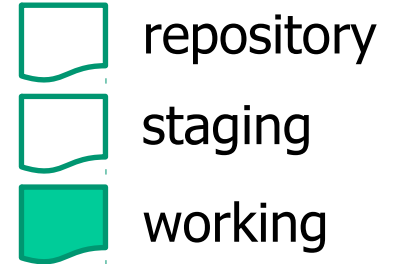
```
package at.spengergasse.bank.domain;

public class BankAccount {
    private String name;

    public BankAccount (String name) {
        setName(name);
    }

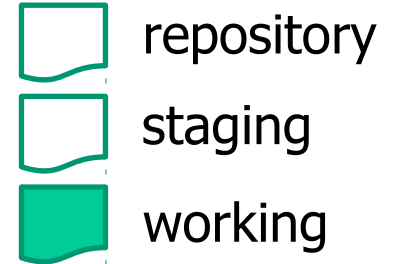
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

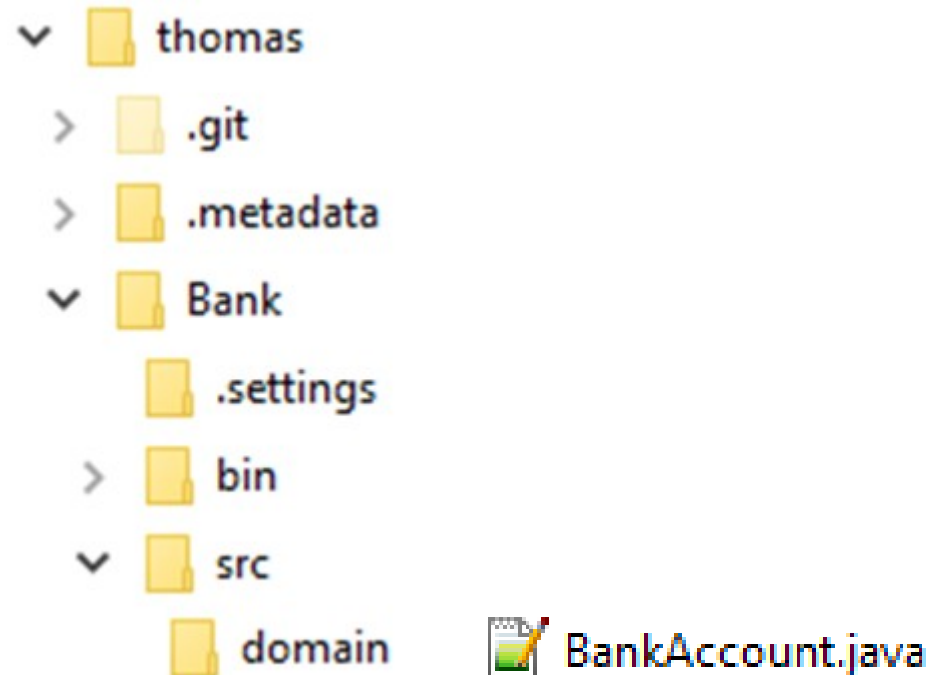


initial commit

- The `git status` command compares the working set with the staging and the repository set.

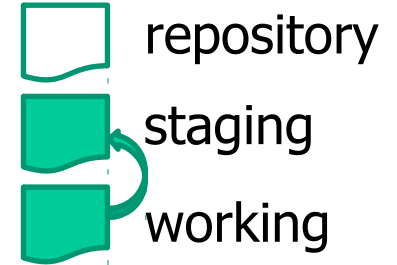


```
$ git status
On branch master
Initial commit
Untracked files:
  .metadata/
  Bank/
```



initial commit

- The `git add` command adds the files from the working set to the staging set.



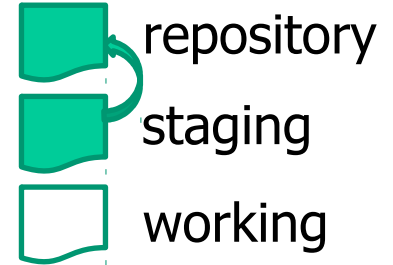
```
$ git add .
```

The file will have its original line endings in your working directory.

warning: LF will be replaced by CRLF in Bank/.gitignore.

initial commit

- The `git commit` command adds the files from the staging set to the repository.
- The `git log` command shows the different sets of the repository.

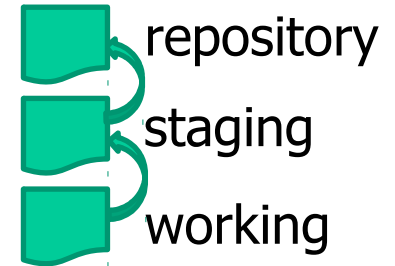


```
$ git commit -m "BankAccount with name"  
create mode 100644 BankAccount.java  
$ git log --oneline  
dcb8267 BankAccount with name
```

dcb8267

master

change



```
package at.spengergasse.bank.domain;
public class BankAccount {
    private String name;
    private int     age;
```

```
$ git status
```

```
On branch master    modified:   BankAccount.java
```

```
$ git add .
```

```
$ git commit -m "BankAccount add age"
```

```
$ git log --oneline
```

```
aa3ca22 BankAccount add age
```

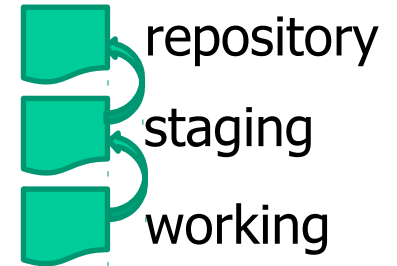
```
dcb8267 BankAccount with name
```

dcb8267

→ aa3ca22

master

add



```
package at.spengergasse.bank.domain;
import java.util.ArrayList;
public class Bank {
    private ArrayList account;
```

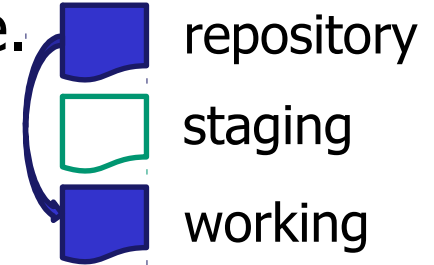
```
$ git status
On branch master    Untracked files: Bank.java
$ git add .
$ git commit -m "Bank ArrayList"
$ git log --oneline
2bdb3ef Bank ArrayList
aa3ca22 BankAccount add age
dcb8267 BankAccount with name
```



master

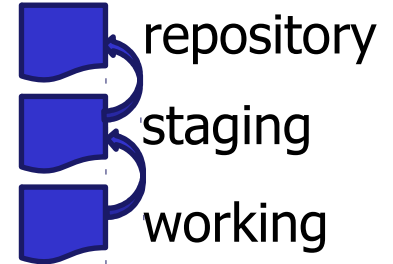
branch

- A branch is a forked new development line.
- The `git branch` command lists, creates or deletes branches.



```
$ git branch
* master
$ git branch hashmap
$ git branch
  hashmap
* master
$ git checkout hashmap
$ git branch
* hashmap
  master
```

branch



```
package at.spengergasse.bank.domain;
import java.util.HashMap;
public class Bank {
    private HashMap account;
```

```
$ git status
```

```
On branch hashmap modified: Bank.java
```

```
$ git add .
```

```
$ git commit -m "Bank changed to HashMap"
```

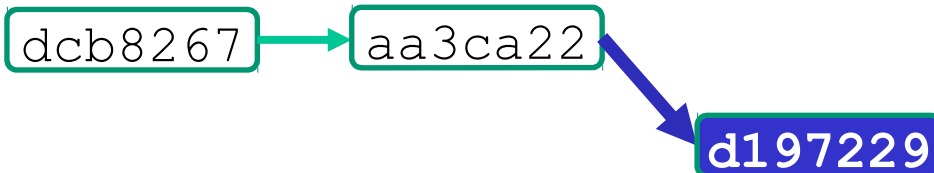
```
$ git log --oneline
```

```
d197229 Bank changed to HashMap
```

```
2bdb3ef Bank ArrayList
```

```
aa3ca22 BankAccount add age
```

```
dcb8267 BankAccount with name
```

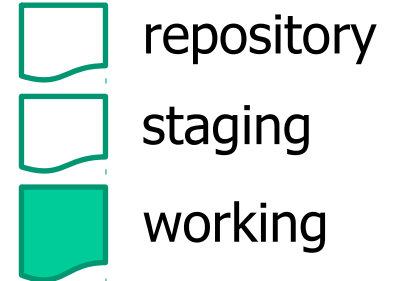


master

hashmap

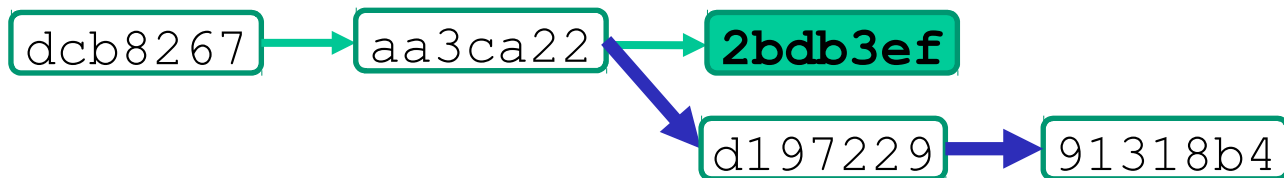
branch

```
$ git checkout master
$ git branch
  hashmap
* master
```



```
package at.spengergasse.bank.domain;
public class BankAccount {
    private String lastName;
    private int    age;
    public BankAccount(int age, String las
```

```
$ git status
On branch master    modified:   BankAccount.java
On branch master    modified:   Bank.java
```

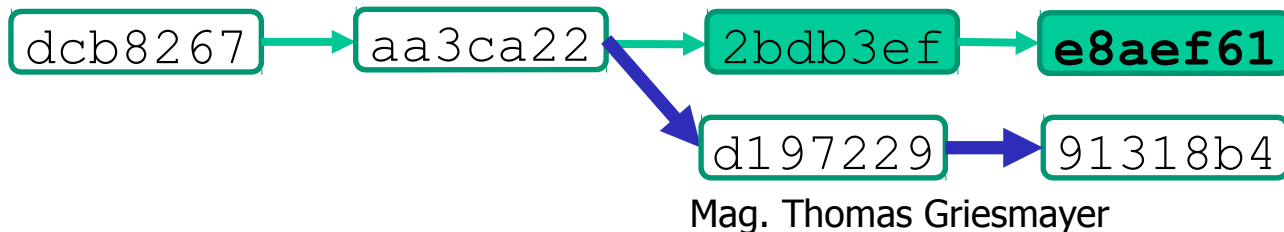
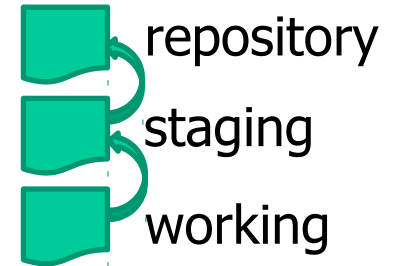


master

hashmap

branch

```
$ git add .  
$ git commit -m "BankAccount lastName"  
$ git log --oneline  
e8aef61 BankAccount lastName  
16cd82c Bank ArrayList  
7c1f7d0 BankAccount add age  
a4233dd BankAccount with name
```



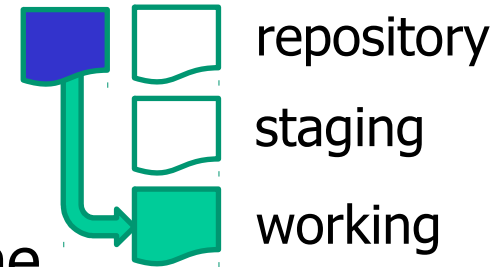
master

hashmap

merge

Merge types:

- fast forward ... master unchanged
- recursive strategy ... both changed
- conflict merge ... both changed the same line of code



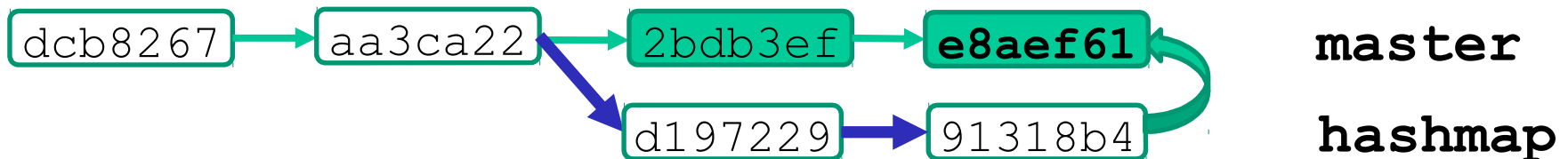
```
$ git merge hashmap
```

```
Auto-merging BankAccount.java
```

```
CONFLICT (content): Merge conflict in  
BankAccount.java
```

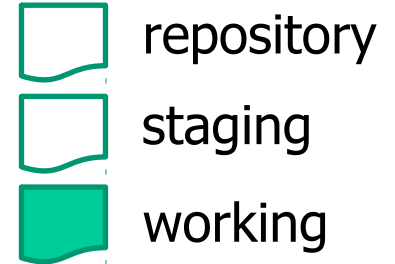
```
Auto-merging BankAccount.class
```

```
CONFLICT (content): Merge conflict in  
BankAccount.class
```

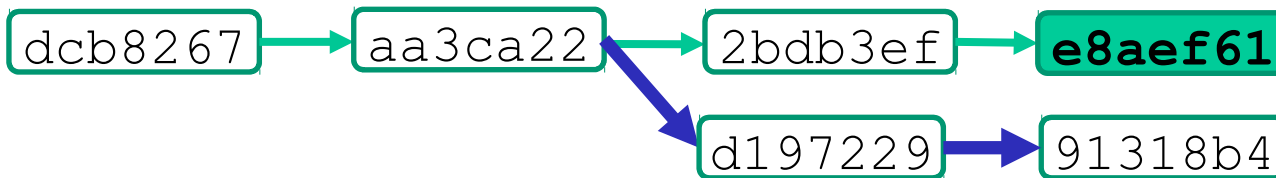


merge

```
<<<<<<< HEAD
  private String lastName;
=====
  private String firstName;
  private String lastName;
>>>>>>> hashmap
```



```
private String firstName;
private String lastName;
```

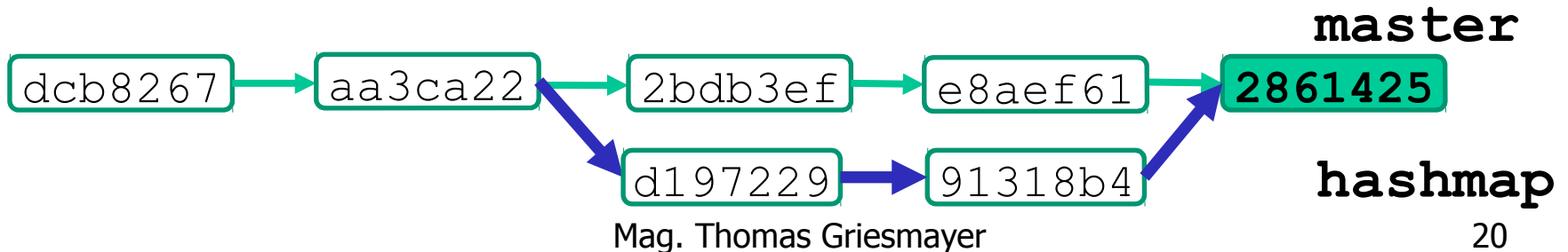
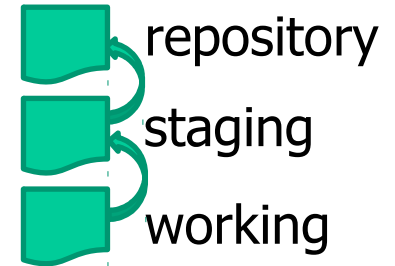


master

hashmap

merge

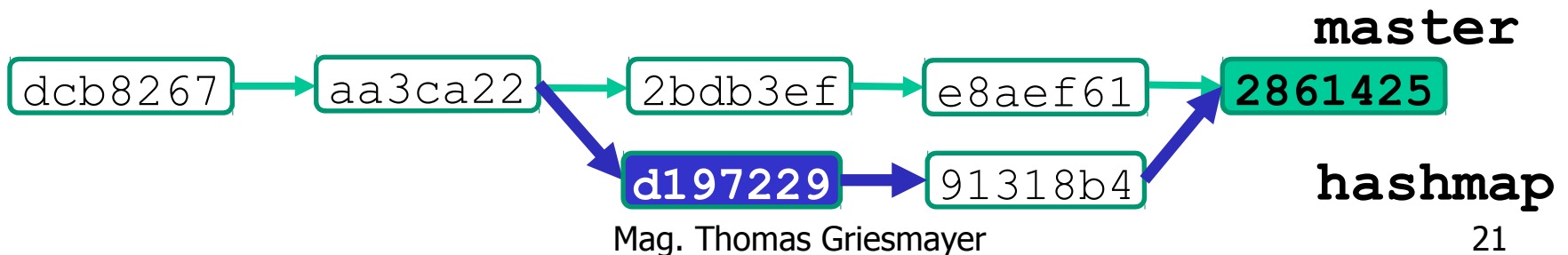
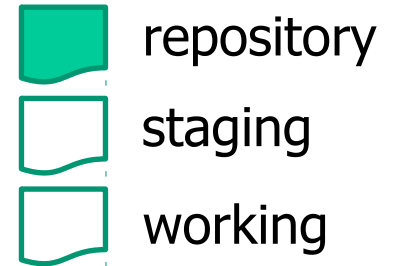
```
$ git add .
$ git commit -m "BankAccount name solved"
$ git log --oneline
2861425 BankAccount name solved
e8aef61 BankAccount only lastName
91318b4 BankAccount firstname and lastname
d197229 Bank changed to HashMap
16cd82c Bank ArrayList
7c1f7d0 BankAccount add age
a4233dd BankAccount with name
```



diff

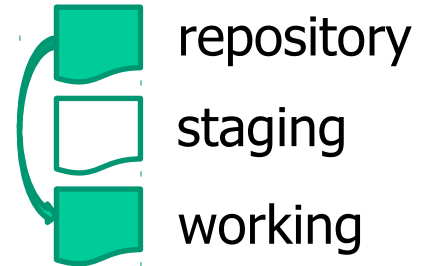
- The command `git diff` compares two repository sets.

```
$ git log --oneline
2861425 BankAccount name solved
d197229 Bank changed to HashMap
$ git diff 2861425 d197229
-<<<<<<< HEAD
- private String lastName;
- =====
- private String firstName;
- private String lastName;
->>>>>> hashmap
```



reset

- The command `git reset` moves back to an older version.



```
$ git reset --hard d197229
$ git log --oneline
d197229 Bank changed to HashMap
16cd82c Bank ArrayList
7c1f7d0 BankAccount add age
a4233dd BankAccount with name
$ git branch
  hashmap
* master
$ git reset --hard 2861425
```

