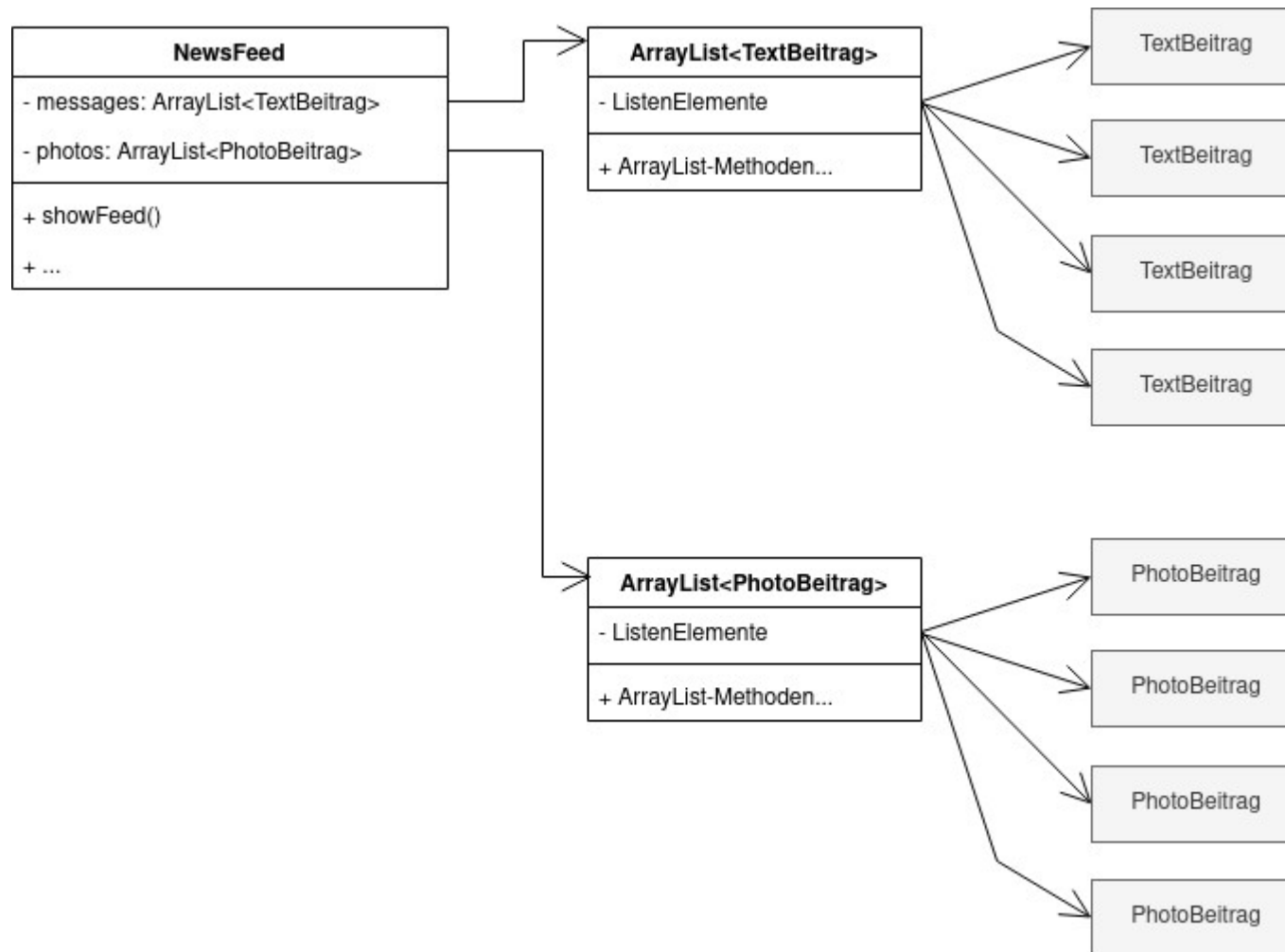


# Vererbung



# Vererbung

```
public class TextBeitrag
{
    private String username;
    private String message;
    private long timestamp;
    private int likes;
    private ArrayList<String> comments;

    public TextBeitrag(String author, String text)
    {
        username = author;
        message = text;
        timestamp = System.currentTimeMillis();
        likes = 0;
        comments = new ArrayList<>();
    }

    public void addComment(String text) ...

    public void like() ...

    public void display() ...

    ...
}
```

## TextBeitrag

- username  
- message  
- timestamp  
- likes  
- comments

+ like  
+ unlike  
+ addComment  
+ getText  
+ getTimestamp  
+ display

# Vererbung

```
public class PhotoBeitrag
{
    private String username;
    private String dateiname;
    private String undertitel;
    private long timestamp;
    private int likes;
    private ArrayList<String> comments;

    public PhotoPost(String author, String dateiname,
                     String undertitel)
    {
        username = author;
        this.dateiname = dateinamenname;
        this.undertitel = undertitel;
        timestamp = System.currentTimeMillis();
        likes = 0;
        comments = new ArrayList<>();
    }

    public void addComment(String text) ...
    public void like() ...
    public void display() ...
    ...
}
```

## PhotoBeitrag

- username  
- dateiname  
- undertitel  
- timestamp  
- likes  
- comments

+ like  
+ unlike  
+ addComment  
+ getBilddatei  
+ getUntertitel  
+ getTimestamp  
+ display

# Vererbung

```
public class NewsFeed
{
    private ArrayList<MessagePost> textPosts;
    private ArrayList<PhotoPost> photoPosts;
    ...
    public void showFeed()
    {
        for(MessagePost text : textPosts) {
            text.display();
        }

        for(PhotoPost photo : photoPosts) {
            photo.display();
        }
    }
}
```

NewsFeed
- messages: ArrayList<TextBeitrag>
- photos: ArrayList<PhotoBeitrag>
+ showFeed()
+ ...



Bei unserem ersten Entwurf gibt es starke **Codedoppelung**. Die Klassen **PhotoPost** und **TextPost** teilen sich zahlreiche Attribute und Methoden, z.T. ist der benötigte Code identisch.

**Problem?** Bei Änderungen muss immer mehrfach angepasst werden → **Aufwand** & **Bugs!**

Nicht offensichtlich: „Codedoppelung“ in **NewsFeed**?!

Das ist eigentlich auch zwei mal genau dasselbe!

```
for(MessagePost text : textPosts) {  
    text.display();  
}  
  
for(PhotoPost photo : photoPosts) {  
    photo.display();  
}
```