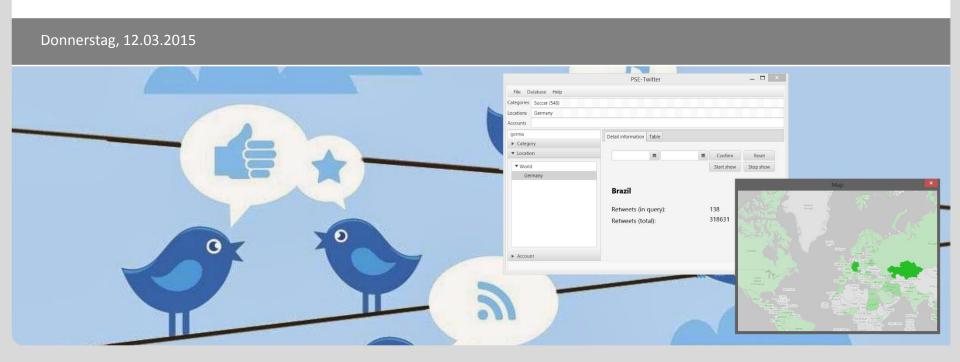


Praxis der Software Entwicklung: Visualizing Trends. Was verrät uns Twitter?



Gliederung

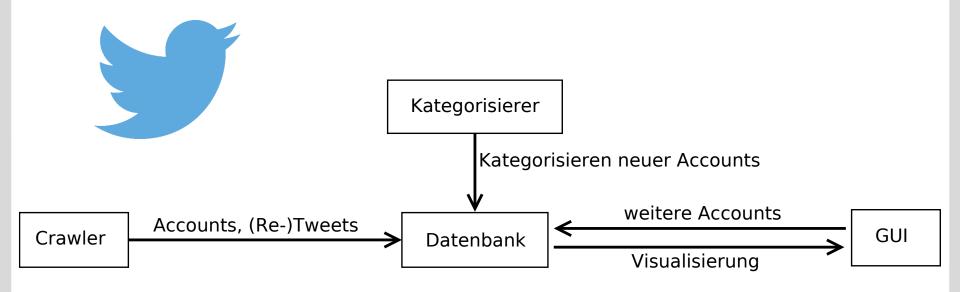


- 1. Aufgabenstellung
- 2. Entwurf
- 3. Implementierung
- 4. Validierung
- 5. Fazit

1. Aufgabenstellung



- Sammeln von Tweets mittels der Twitter-API
- Auswertung durch austauschbare Analysekomponenten
- Visualisierung der Analyse-Ergebnisse in einer graphischen Oberfläche



2. Entwurf (Datenbank)



Accounts

Location

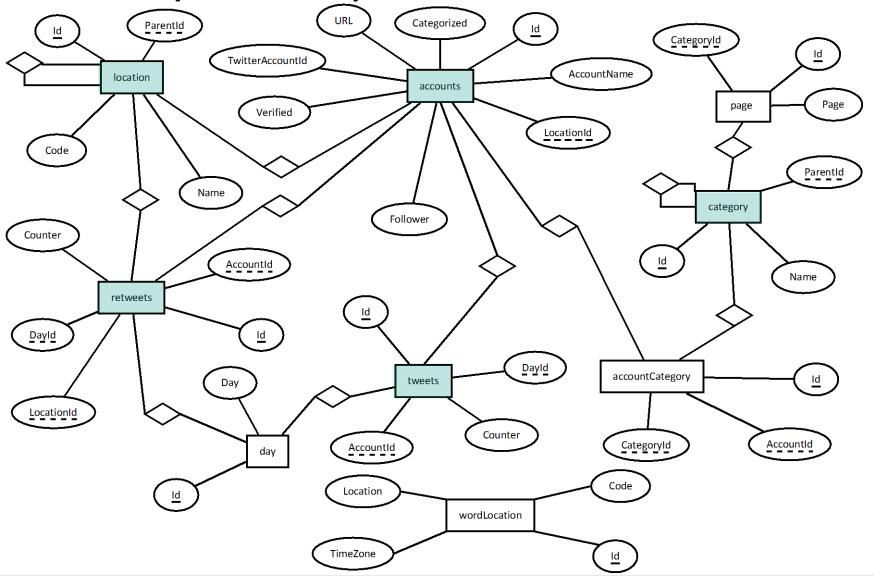
Category

Tweets / Retweets

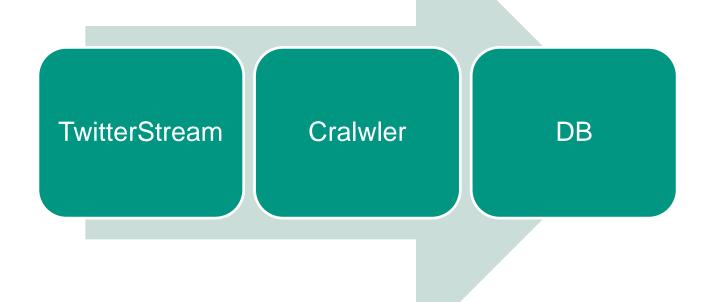
Validierung

2. Entwurf (Datenbank)









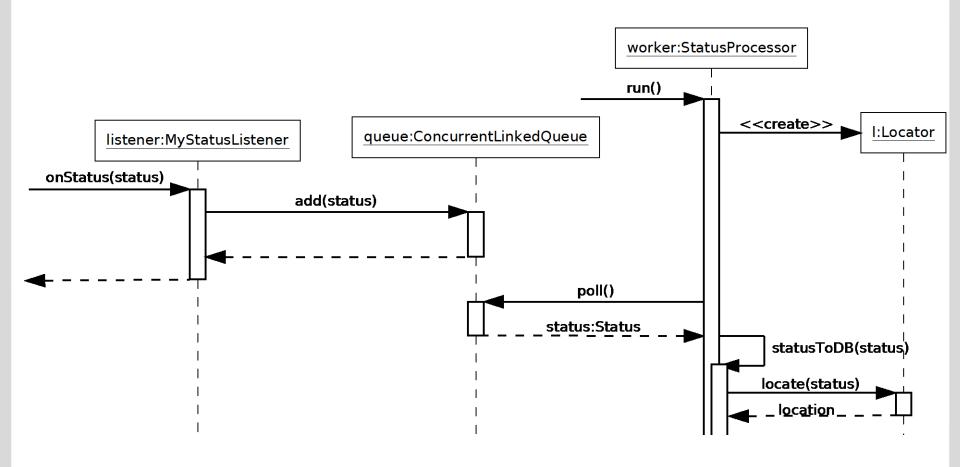


StatusProcessor

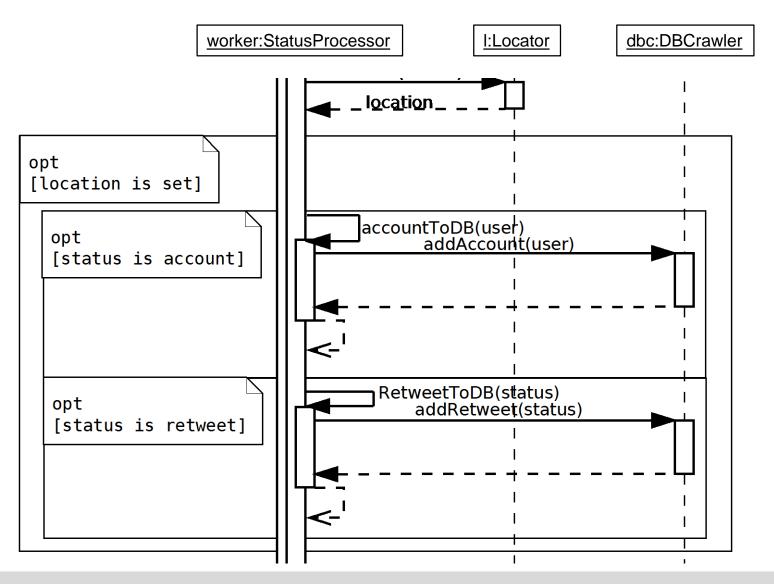
Locator

WebServiceLocator

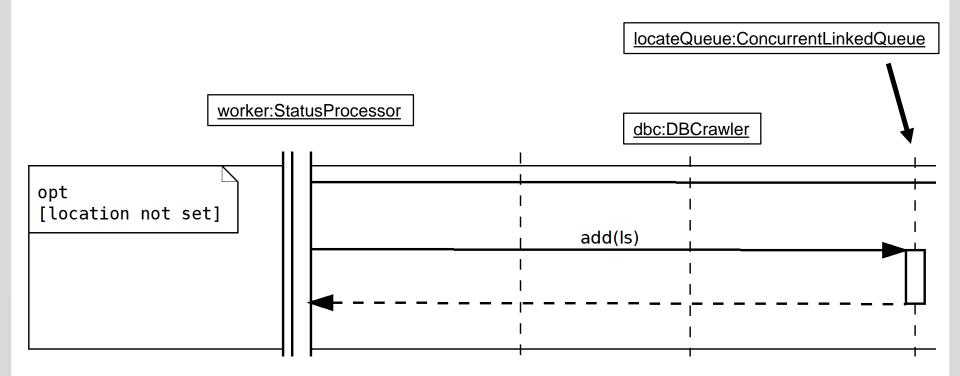






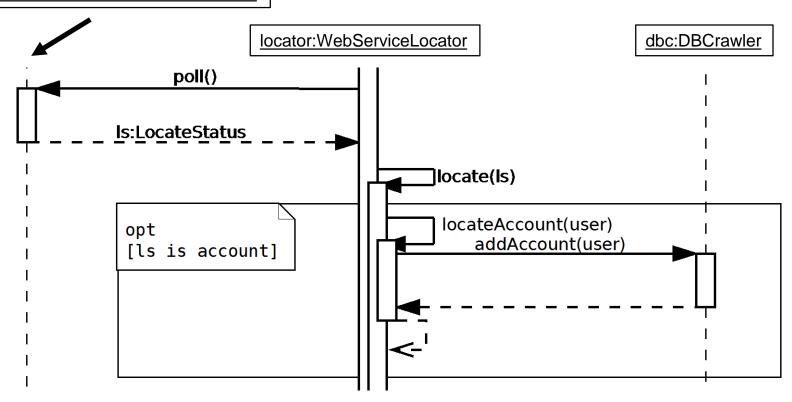




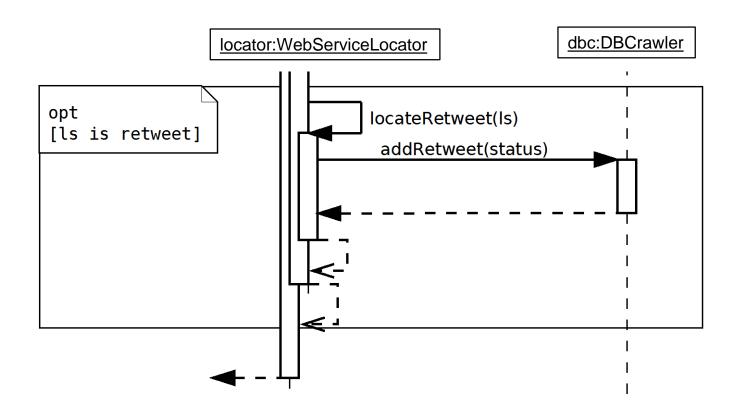




<u>locateQueue:ConcurrentLinkedQueue</u>

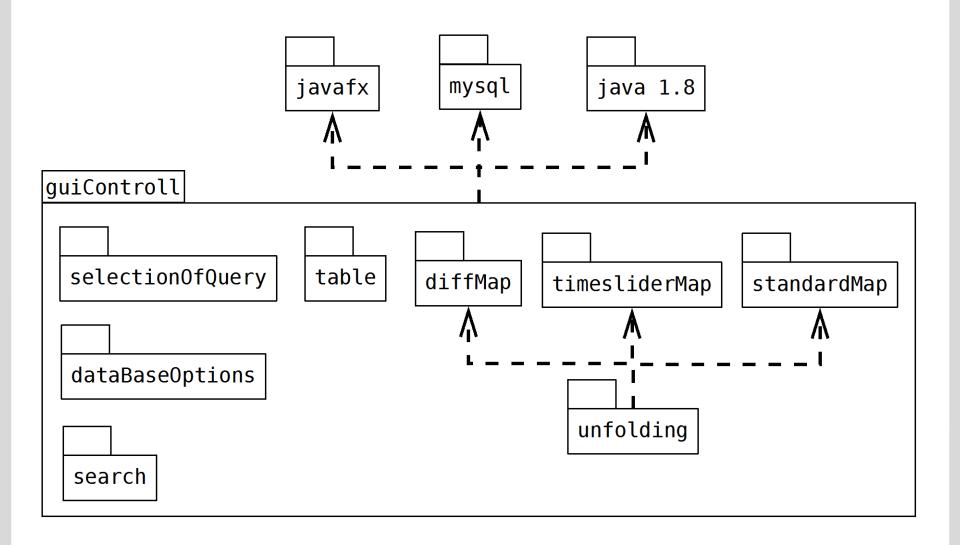






2. Entwurf (GUI)





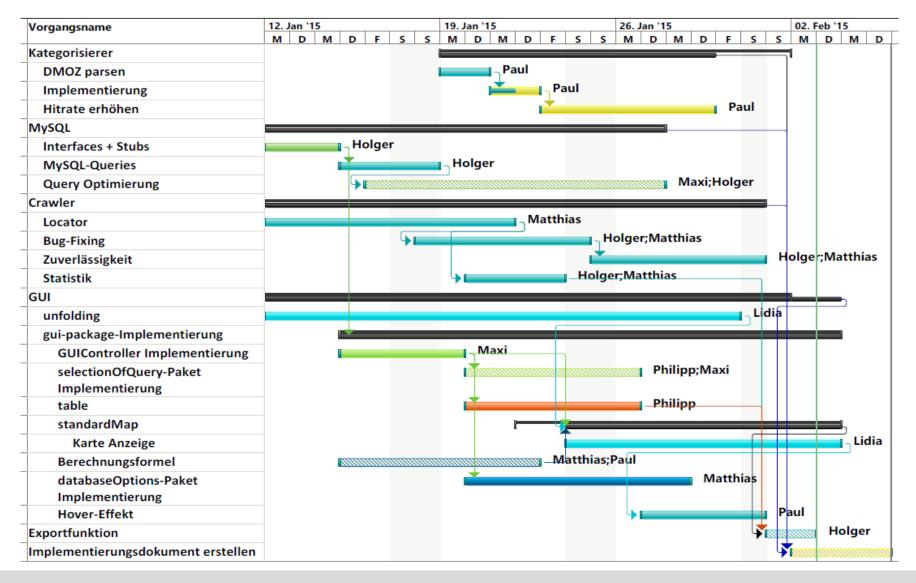
2. Entwurf (GUI)



```
?xml version="1.0" encoding="UTF-8"?>
guiControll
             ?import javafx.scene.control.*?>
             <mark>?</mark>import java.lang.*<mark>?></mark>
             <mark>?</mark>import javafx.scene.layout.*<mark>?></mark>
           <GridPane xmlns="http://javafx.com/javafx/8" xmlns:fx="http://javafx.com/</pre>
              <columnConstraints>
    GuiVie
                <ColumnConstraints hgrow="ALWAYS"/>
   +frame: Par
              </columnConstraints>
   +tabs: Tabl
              <rewConstraints>
                <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="NEVER" />
                <RowConstraints minHeight="200" prefHeight="200" vgrow="ALWAYS" />
              </re>
               <children>
                   <TextField fx:id="txtFilterSearch" onKeyReleased="#handle"/>
                   <Accordion GridPane.rowIndex="1">
  selectionOf
                     <panes>
                                       SelectionOfQueryView
                                      +categoryTree: TreeView
                                                                        <<abstract>>
       SelectionOfQueryController <
                                                                                         <<abstract>>
                                      +countryTree: TreeView
                                                                        InputElement
                                                                                        OutputElement
                                      -accountList: ListView
      +update()
                                      +selectedCategories: ListView
                                      +selectedCountries: ListView
                                      +selectedAccounts: ListView
```

3. Implementierung





4. Validierung



- Checkstyle
- JUnit

```
gui.table
                                                 99,4 %
  ContentTableController.java
                                                100,0 %
  ■ InternAccount.java
                                                 97,2 %
unfolding
                                                 86,5 %
  MyUnfoldingMap.java
                                                 86.6 %
  MyDataEntry.java
                                                 86,2 %
🖶 gui
                                                 76,3 %
  InputElement.java
                                                100,0 %
  OutputElement.java
                                                100,0 %
  PPPPRunnable.java
                                                100,0 %
  PPRunnable.java
                                                100,0 %
  PRunnable.java
                                                100,0 %
  GUIElement.java
                                                 97,0 %
  InfoRunnable.java
                                                 95,8 %
  Util.java
                                                 92,1 %
```

```
@Test
public void testSubscribe() throws InterruptedException {
    TestGUIElement e = new TestGUIElement();
    guiController.subscribe(e);
    guiController.setDontLoadFromDB(true);
    Thread.sleep(1000);
    assertTrue(e.isUpdated());
    guiController.setDontLoadFromDB(false);
}
```

5. Fazit



- GitHub
- Treffen / Videokonferenzen
- Beobachter Muster in GUI
 - leicht erweiterbar
- Entkoppelung einzelner Komponenten
 - Änderungen unproblematisch
- "Prototyp" zu jeder Phase
 - Problem: unfolding ⇔ JavaFX