

Iterative Abschätzung von Performance-Modellparametern unter Berücksichtigung von parametrischen Abhängigkeiten

Code Review

Student: Jan-Philipp Jägers, Betreuerin: Manar Mazkatli

ARCHITECTURE-DRIVEN REQUIREMENTS ENGINEERING
INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION, FAKULTÄT FÜR INFORMATIK

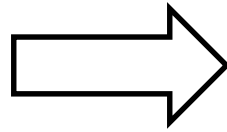


Motivation

```
class A {  
    private B externalB;  
  
    public int methodA(int a) {  
        /* computation C1 */  
        int result = 0;  
        for (int i = 0; i < a; i++) {  
            /* computation C2 */  
            result += externalB.methodB1();  
        }  
  
        return result;  
    }  
}
```



Motivation



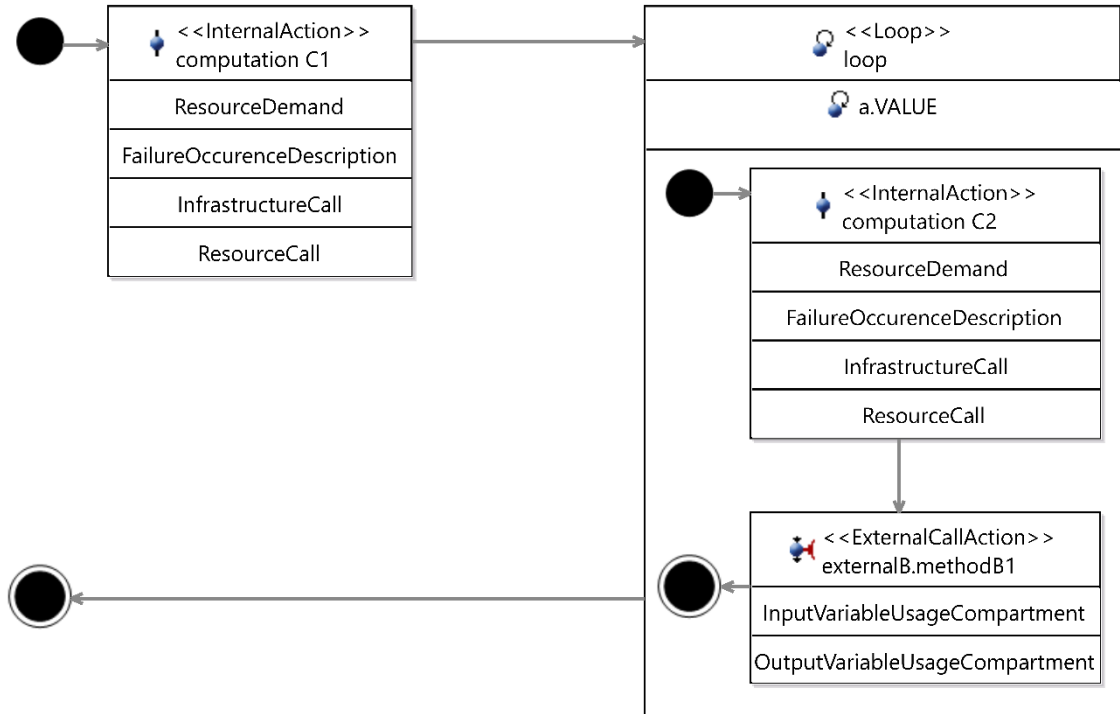
```

class A {
  private B externalB;

  public int methodA(int a) {
    /* computation C1 */
    int result = 0;
    for (int i = 0; i < a; i++) {
      /* computation C2 */
      result += externalB.methodB1();
    }

    return result;
  }
}

```

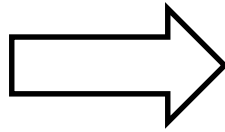


Motivation

```
class A {  
    private B externalB;  
  
    public int methodA(int a) {  
        /* computation C1 */  
        int result = 0;  
        for (int i = 0; i < a; i++) {  
            /* computation C2 */  
            result += externalB.methodB1();  
        }  
        /* ++++ new code block start ++++ */  
        if (result < CONST.A) {  
            externalB.methodB2();  
        }  
        /* —— new code block end —— */  
        return result;  
    }  
}
```



Motivation

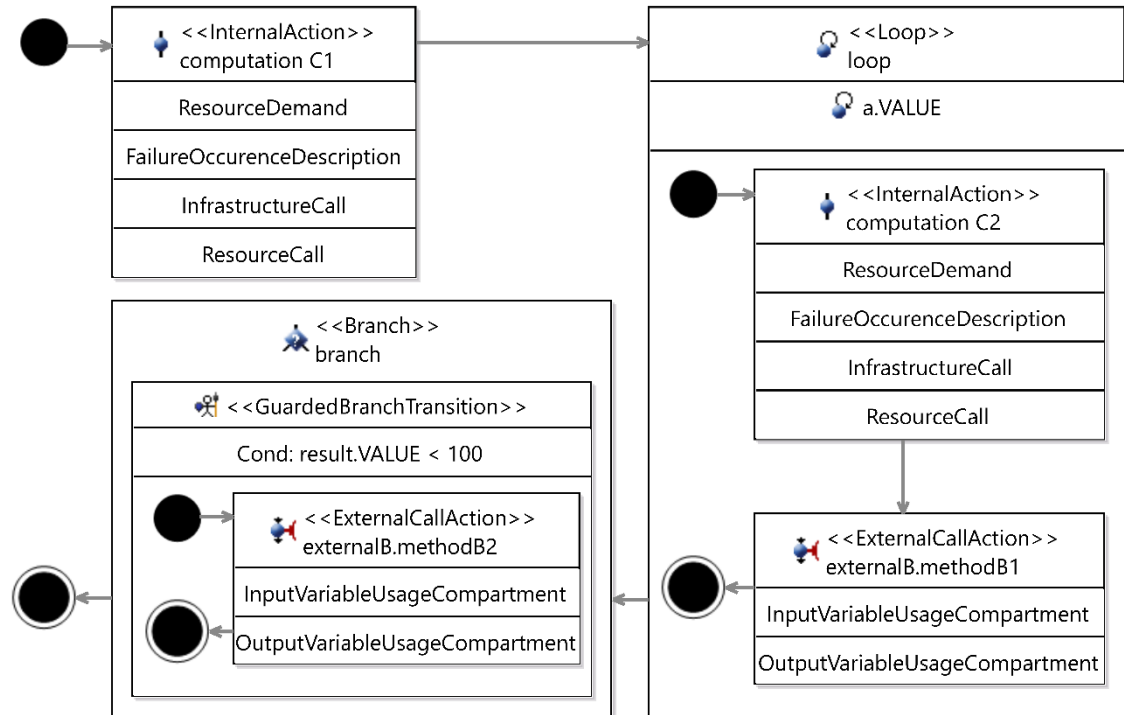


```

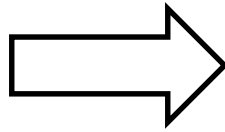
class A {
  private B externalB;

  public int methodA(int a) {
    /* computation C1 */
    int result = 0;
    for (int i = 0; i < a; i++) {
      /* computation C2 */
      result += externalB.methodB1();
    }
    /* ++++ new code block start ++++ */
    if (result < CONST.A) {
      externalB.methodB2();
    }
    /* —— new code block end —— */
    return result;
  }
}

```

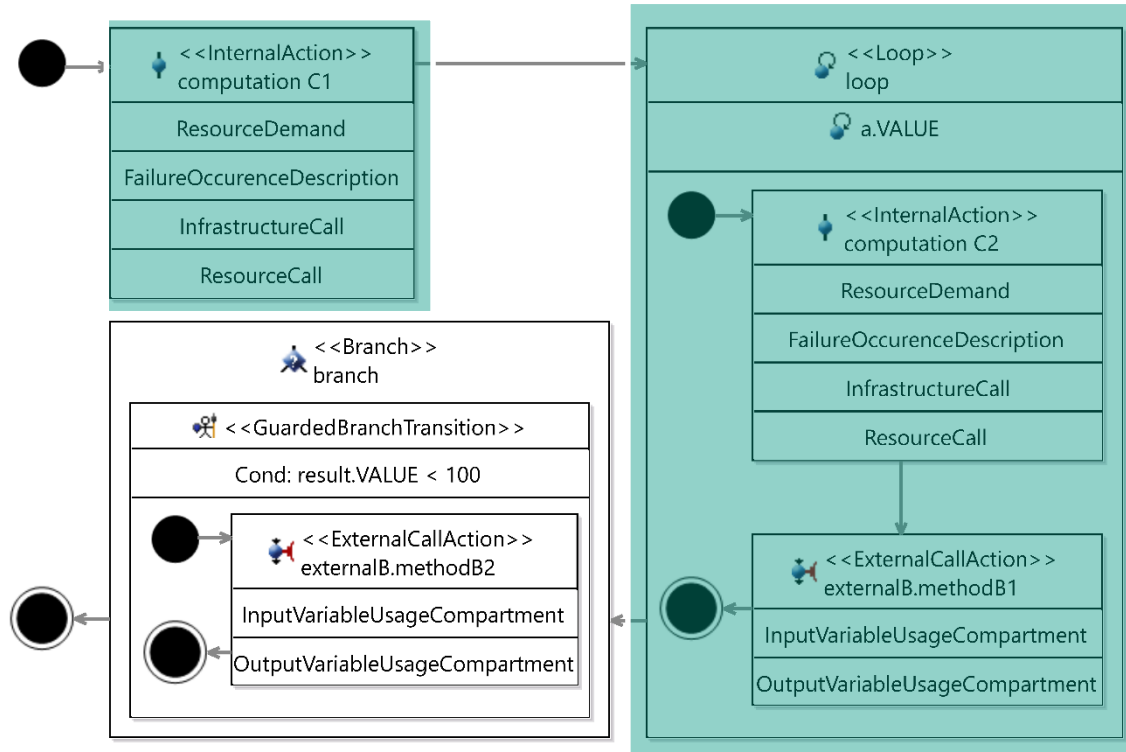


Motivation



```
class A {
  private B externalB;

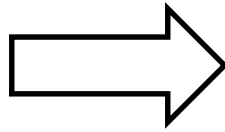
  public int methodA(int a) {
    /* computation C1 */
    int result = 0;
    for (int i = 0; i < a; i++) {
      /* computation C2 */
      result += externalB.methodB1();
    }
    /* ++++ new code block start ++++ */
    if (result < CONST.A) {
      externalB.methodB2();
    }
    /* ——— new code block end ——— */
    return result;
  }
}
```



[Langhammer, 2017]



Motivation

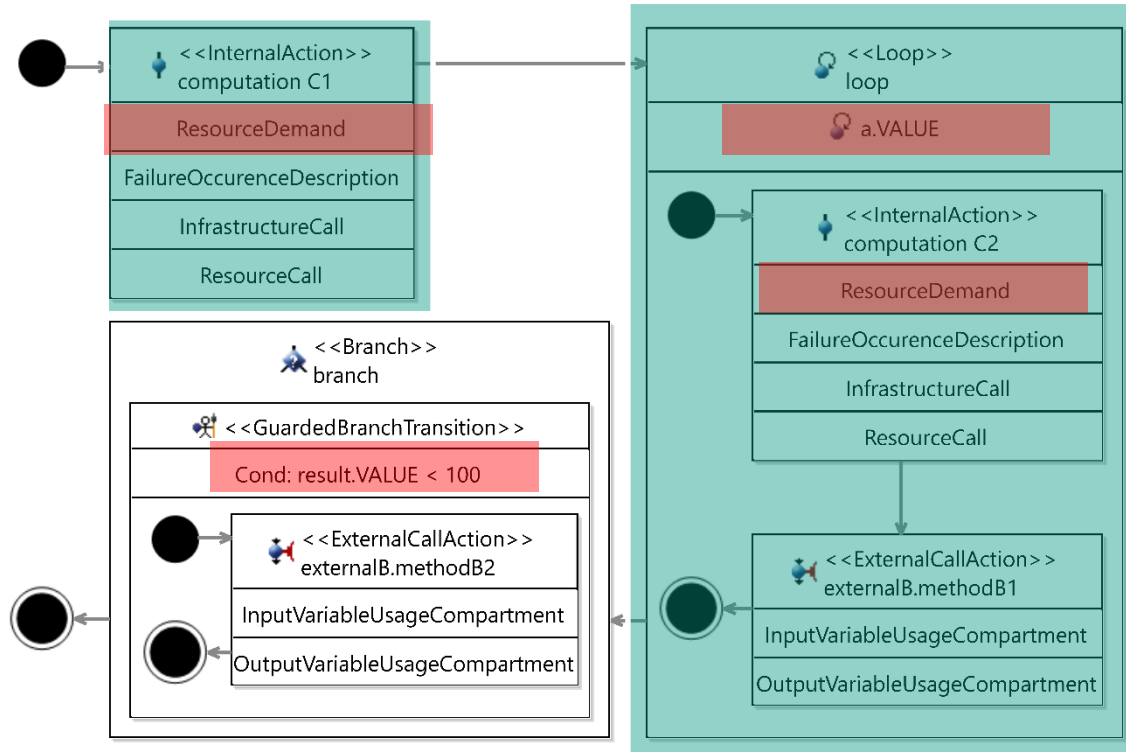


```

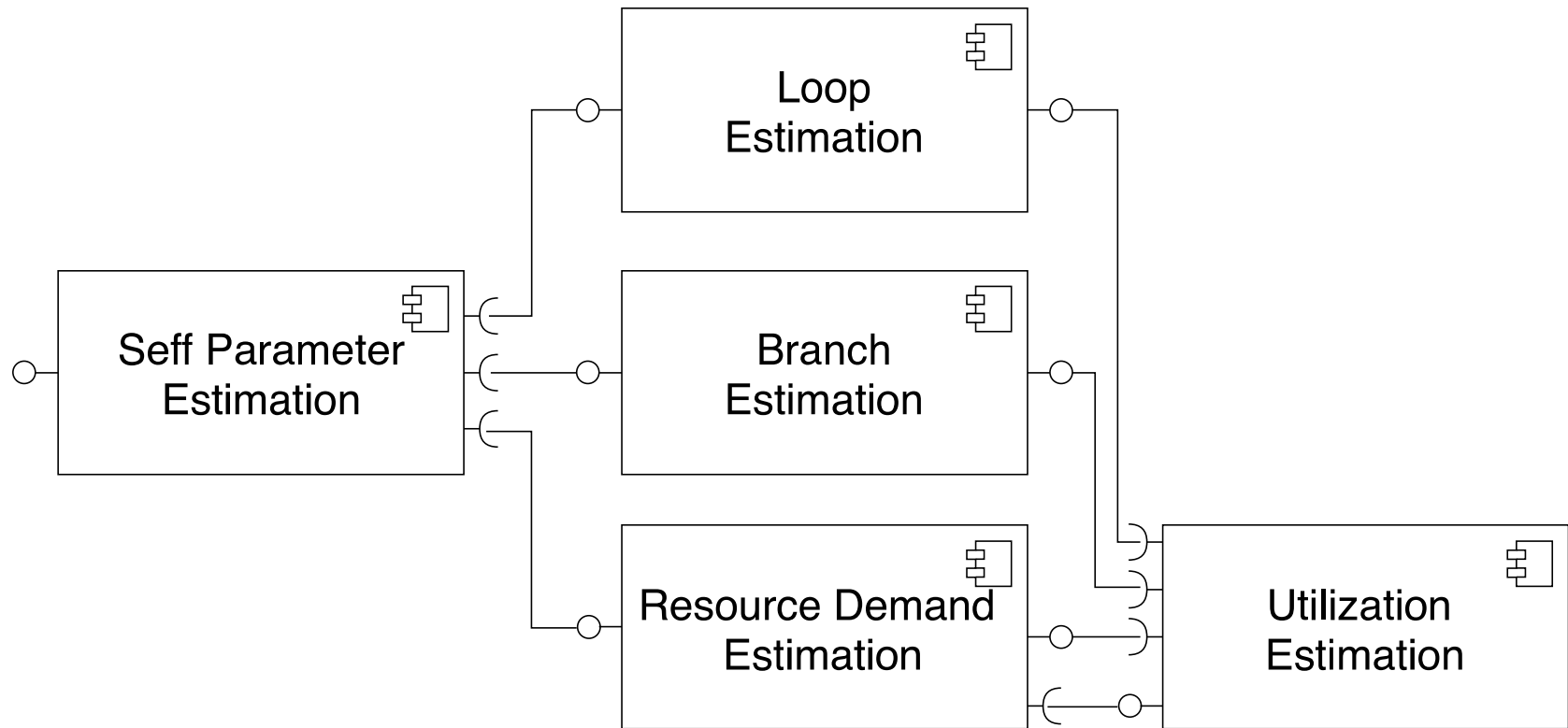
class A {
  private B externalB;

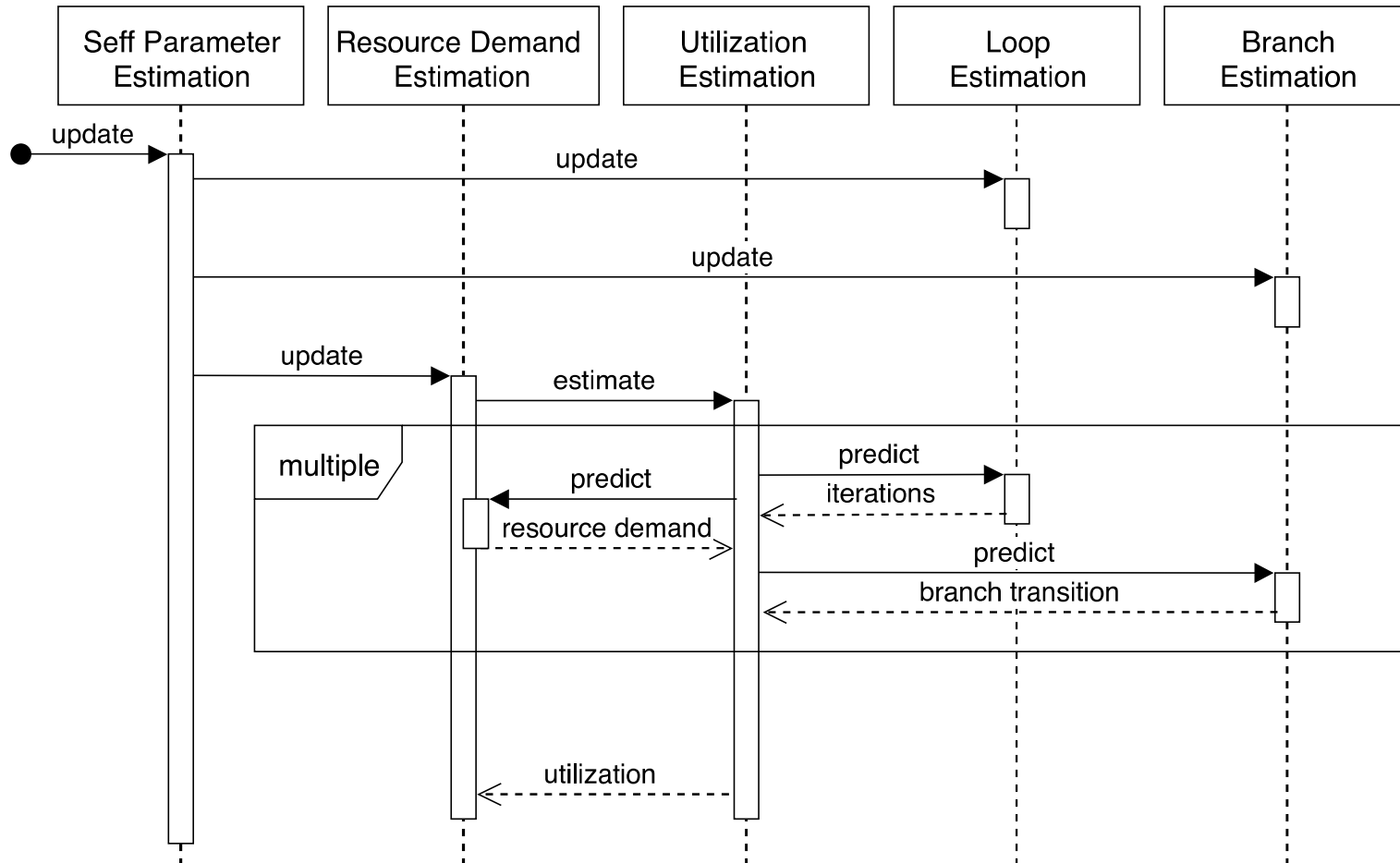
  public int methodA(int a) {
    /* computation C1 */
    int result = 0;
    for (int i = 0; i < a; i++) {
      /* computation C2 */
      result += externalB.methodB1();
    }
    /* ++++ new code block start ++++ */
    if (result < CONST.A) {
      externalB.methodB2();
    }
    /* —— new code block end —— */
    return result;
  }
}

```

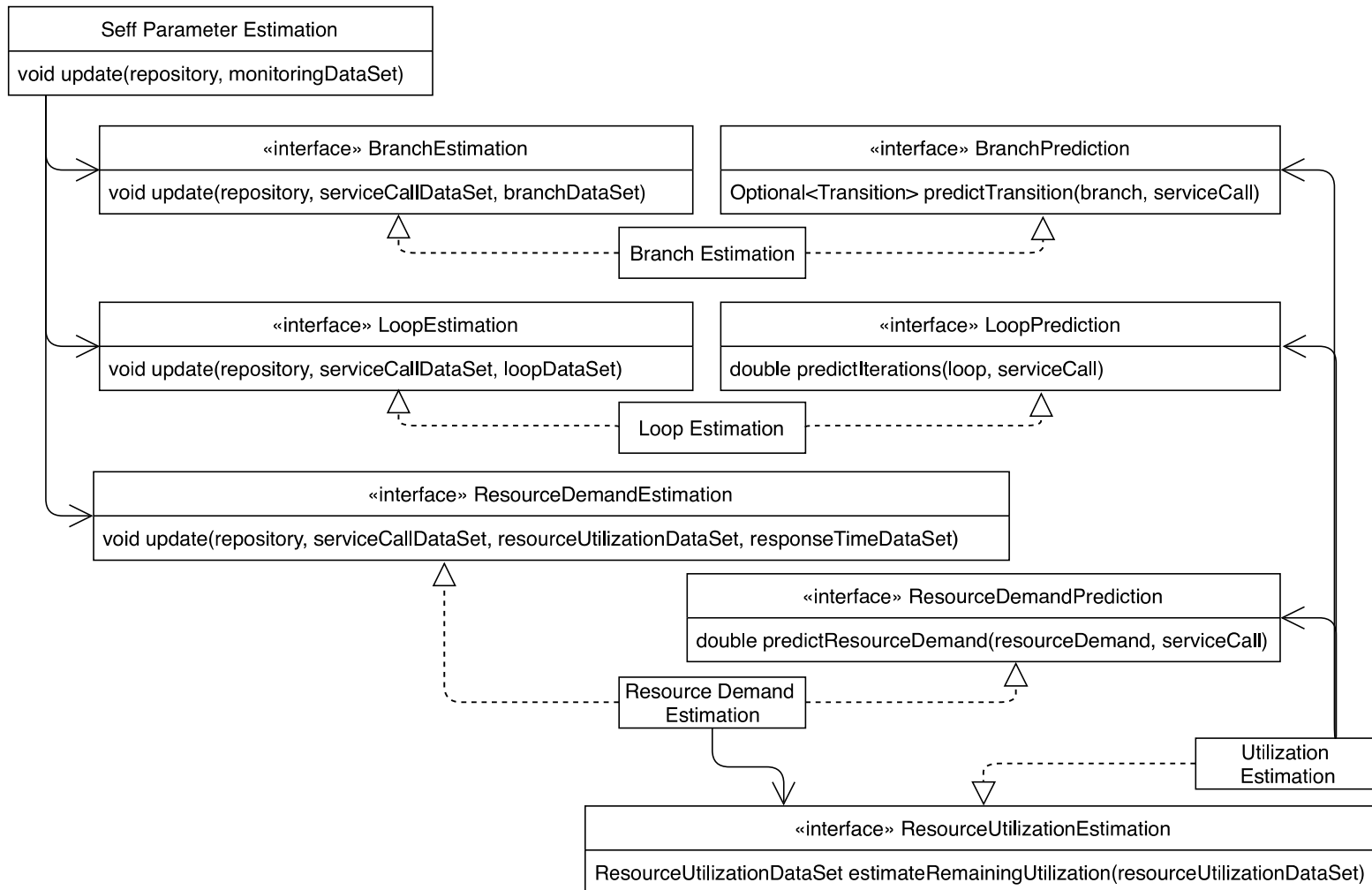


Main Components





Schnittstellen



Motivation

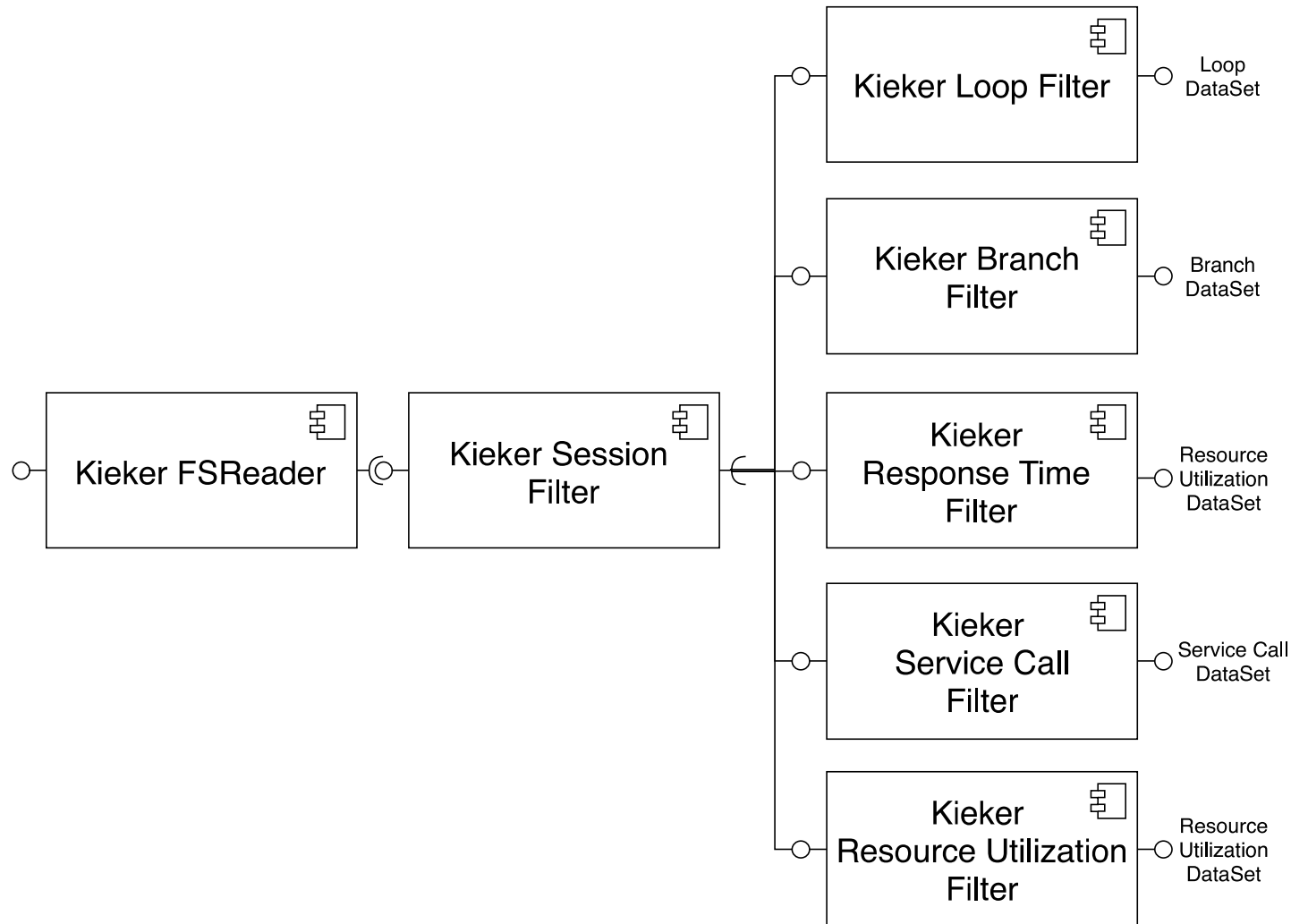


Architektur



Review

Lesen der Monitoring Daten



Motivation

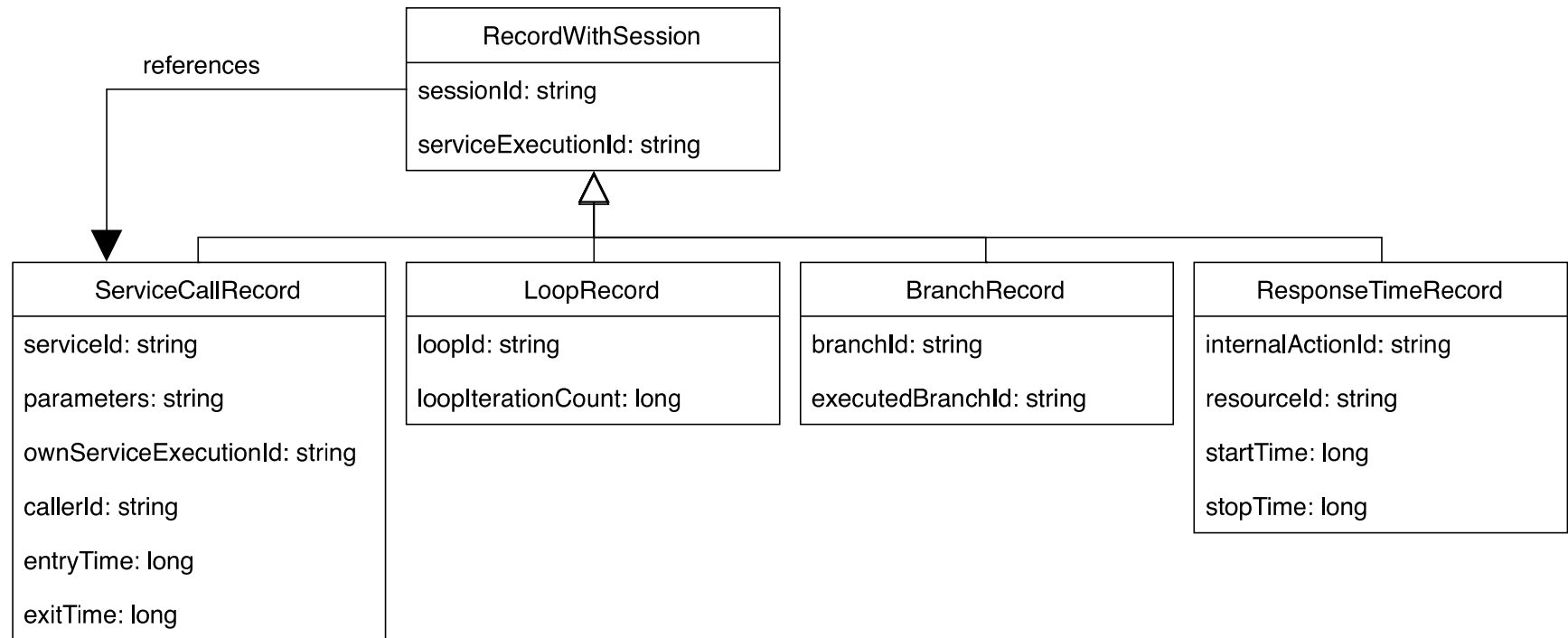


Architektur



Review

Monitoring Daten



Review

