

**Team:** Falco Winkler (FW), Daniel Schruhl (DS)

**Aufgabenteilung:**

**Quellenangaben:**

**Bearbeitungszeitraum:**

- 26.03.2017 (FW,DS)

**Aktueller Stand:**

**Änderung des Entwurfs:**

-

# Inhaltsverzeichnis

<b>1 Gesamtsystem</b>	<b>3</b>
1.1 Architekturüberblick . . . . .	3
<b>2 Subsysteme und Komponenten</b>	<b>4</b>
2.1 Client . . . . .	4
2.1.1 Aufgabe und Verantwortung des Clients . . . . .	4
2.1.2 Entwurfsentscheidungen . . . . .	5
2.1.3 Außensicht . . . . .	5
2.1.4 Innensicht . . . . .	5
2.1.5 Konfigurationsparameter . . . . .	5
2.1.6 Komponente Leserclient . . . . .	5
2.1.7 Komponente Redakteurclient . . . . .	5
2.2 Server . . . . .	5
2.2.1 Aufgabe und Verantwortung des Servers . . . . .	5
2.2.2 Entwurfsentscheidungen . . . . .	6
2.2.3 Außensicht . . . . .	6
2.2.4 Innensicht . . . . .	6
2.2.5 Konfigurationsparameter . . . . .	6
2.2.6 Benutzungsschnittstelle . . . . .	6
2.2.7 Komponente CMEM . . . . .	6
2.2.8 Komponente HBQ . . . . .	6
2.2.9 Komponente DLQ . . . . .	7

# 1 Gesamtsystem

## 1.1 Architekturüberblick

Komponentendiagramm

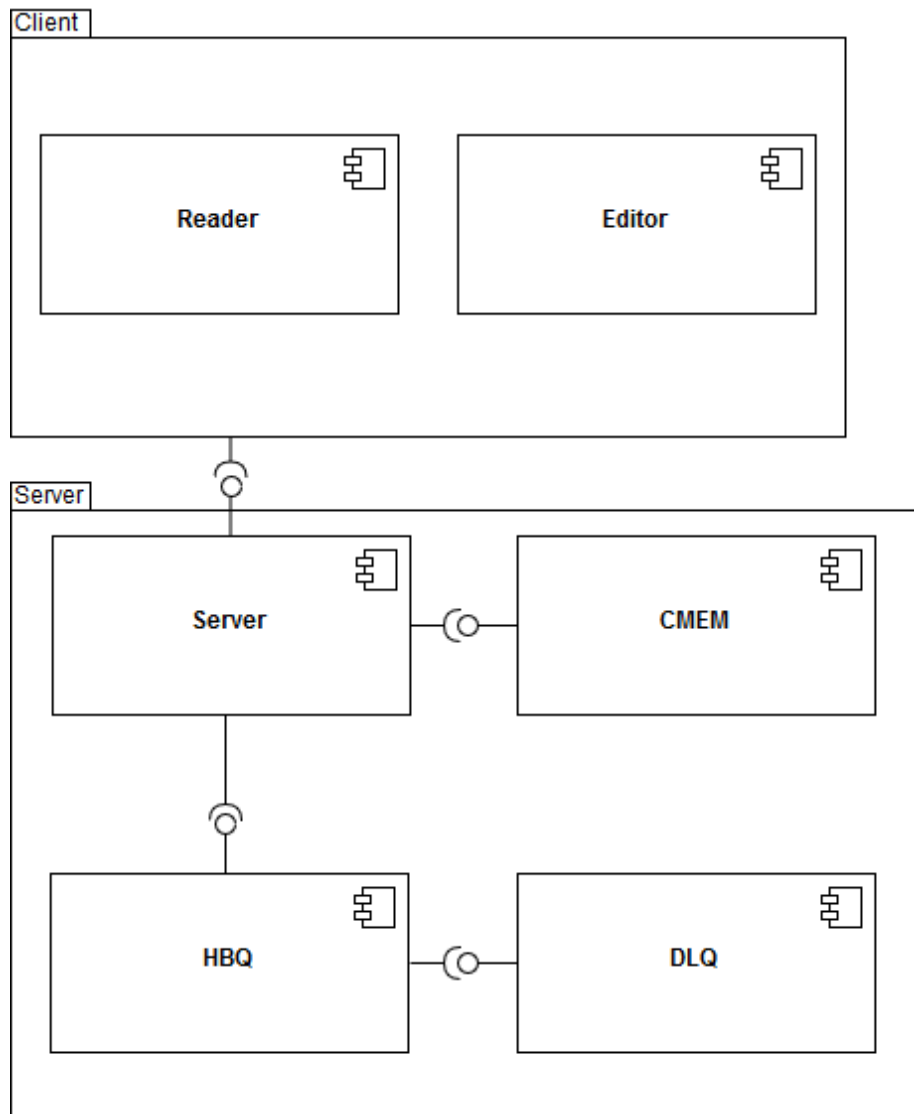


Abbildung 1: Komponentendiagramm der Message Of The Day App

Das Softwareprodukt besteht aus mehreren Modulen und Paketen. Das Client-Paket und das Server-Paket.

Im Server Paket befinden sich das Server-Modul und alle vom Server-Modul verwendeten Datenstrukturen in Modulen. Dazu gehören das HBQ-Modul, das DLQ-Modul und das CMEM-Modul.

Die Schnittstellen des DLQ-Moduls und CMEM-Moduls sind als RPC implementiert. Die Schnittstellen des HBQ-Moduls und des Server-Moduls sind als entfernte ADT realisiert.

Die DLQ Schnittstelle wird nur von der HBQ konsumiert und die CMEM Schnittstelle wird nur vom Server konsumiert. Die Schnittstelle der HBQ wird nur vom Server verwendet und die Schnittstelle vom Server ist nur an den Client angebunden.

Das Client-Paket beinhaltet die Client-Module. Diese sind zum einen der Lese Client (Reader-Modul) und der Redakteur Client (Editor-Modul). Beide Clients sind als ein Prozess implementiert.

### Entwurf:

#### Sequenzdiagramm

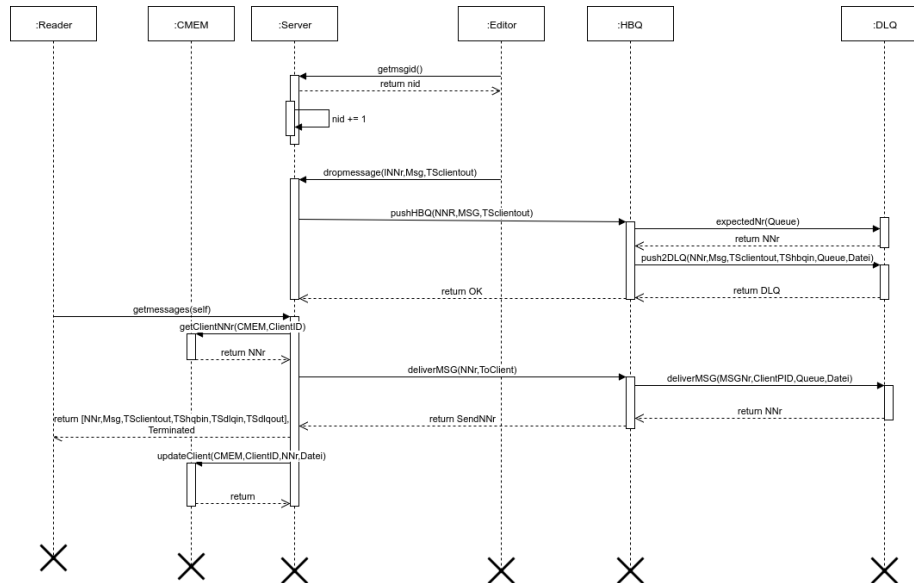


Abbildung 2: Sequenzdiagramm bei fehlerfreiem Nachrichtenaustausch

## 2 Subsysteme und Komponenten

### 2.1 Client

#### 2.1.1 Aufgabe und Verantwortung des Clients

Der Client ist ein einzelner Prozess, der sich jedoch in zwei Rollen aufteilt. Im Endlosdurchlauf einer Schleife wechselt er zwischen Leser und Redakteur - Rolle.

Der Client wird in drei Module aufgespalten: Client, Reader und Writer. Client ruft

in einer Endlosschleife wechselweise die soeben beschriebene Funktionalität in den Modulen Reader und Writer auf.

### **2.1.2 Entwurfsentscheidungen**

#### **2.1.3 Außensicht**

#### **2.1.4 Innensicht**

#### **2.1.5 Konfigurationsparameter**

#### **2.1.6 Komponente Leserclient**

**Aufgabe und Verantwortung des Leseclients** Der Leseclient liest solange Nachrichten vom Server, bis keine weiteren Nachrichten mehr vorhanden sind.

### **Entwurfsentscheidungen**

#### **Außensicht**

#### **Innensicht**

#### **Konfigurationsparameter**

#### **2.1.7 Komponente Redakteurclient**

**Aufgabe und Verantwortung des Redakteurclients** Der Redakteurclient sendet fünf Nachrichten an den Server, und fragt danach einmal die nächste NNr ab, um eine Lücke in den fortlaufend nummerierten Nachrichten zu produzieren.

### **Entwurfsentscheidungen**

#### **Außensicht**

#### **Innensicht**

#### **Konfigurationsparameter**

## **2.2 Server**

### **2.2.1 Aufgabe und Verantwortung des Servers**

Der Server hat die Aufgaben

1. Sich anfragende Clients mit ProzessID und Zeitstempel zu merken
2. Clients die ihnen zugehörigen Nachrichten zurückzugeben

Für die erste Aufgabe wird das Modul CMEM verwendet. Zum Senden an die Clients leitet der Server die Nachrichten zunächst an die HBQ weiter. Erscheint eine Anfrage vom Client, erfolgt über die HBQ der Aufruf an die DLQ, die Nachricht zu übermitteln.

#### **2.2.2 Entwurfsentscheidungen**

#### **2.2.3 Außensicht**

#### **2.2.4 Innensicht**

#### **2.2.5 Konfigurationsparameter**

#### **2.2.6 Benutzungsschnittstelle**

#### **2.2.7 Komponente CMEM**

**Aufgabe und Verantwortung der CMEM** Dieses Modul hat die Aufgabe sich anfragende Clients mittels einer ProzessID und eines Zeitstempels in einer Map-Datenstruktur zu merken, und diese für den Server bereitzustellen.

#### **Entwurfsentscheidungen**

#### **Außensicht**

#### **Innensicht**

#### **Konfigurationsparameter**

#### **2.2.8 Komponente HBQ**

**Aufgabe und Verantwortung der HBQ** Von Redakteur-Clients gesendete Nachrichten werden hier zwischengespeichert. Gibt es keine Lücke in der fortlaufenden Nummerierung der Nachrichten, werden Sie an die DLQ weitergeleitet. Bei einer Überfüllung der HBQ (2/3 der Maximalen Kapazität der DLQ) wird die älteste Lücke in den Nachrichten mit einer künstlichen Nachricht geschlossen, und es werden erneut alle Lückenlosen Nachrichten an die DLQ weitergeleitet. Die HBQ stellt nur ein Interface für den Server bereit.

#### **Entwurfsentscheidungen**

#### **Außensicht**

#### **Innensicht**

#### **Konfigurationsparameter**

### **2.2.9 Komponente DLQ**

**Aufgabe und Verantwortung der DLQ** Die Delivery - Queue hat die Aufgabe Nachrichten an Clients zuzustellen. Die Nachrichten in der DLQ liegen immer sortiert vor, und es gibt keine Lücken in den NNr's. Bei dem Sendevorgang erhält die DLQ ein Signal von der HBQ mit der PID des Ziel - Clients und der Nachrichtennummer. Daraufhin wird die Nachricht mit dieser NNr an den Client verschickt. Wenn diese Nachricht nicht vorhanden ist wird die Nachricht mit der nächst höheren NNr verschickt. Es wird jeweils die Nummer der verschickten Nachricht zurückgegeben.

#### **Entwurfsentscheidungen**

##### **Außensicht**

##### **Innensicht**

##### **Konfigurationsparameter**