

---

# DEZSYS Protokoll

## Load Balancing

---

DEZSYS  
5BHITT 2015/16

Klaus Ableitinger  
Manuel Reiländer

Note:  
Betreuer: Th.Micheler

Version 0.2  
Begonnen am 12. Feb. 2016  
Beendet am 12. Februar 2016

# Inhaltsverzeichnis

<b>1</b>	<b>Aufgabenstellung</b>	<b>1</b>
1.1	Auslastung . . . . .	1
1.2	Tests . . . . .	1
1.3	Modalitäten . . . . .	2
1.4	Quellen . . . . .	2
<b>2</b>	<b>Ergebnisse</b>	<b>3</b>
2.1	Tabelle . . . . .	3
2.2	Aufzählung . . . . .	3
2.3	Code . . . . .	3

# 1 Aufgabenstellung

Es soll ein Load Balancer mit mindestens 2 unterschiedlichen Load-Balancing Methoden (jeweils 6 Punkte) implementiert werden (ähnlich dem PI Beispiel [1]; Lösung zum Teil veraltet [2]). Eine Kombination von mehreren Methoden ist möglich. Die Berechnung bzw. das Service ist frei wählbar!

Folgende Load Balancing Methoden stehen zur Auswahl:

- Weighted Distribution
- Least Connection
- Response Time
- Server Probes

Um die Komplexität zu steigern, soll zusätzlich eine "Session Persistence" (2 Punkte) implementiert werden.

Vertiefend soll eine Open-Source Applikation aus folgender Liste ausgewählt und installiert werden. (2 Punkte)

<https://www.inlab.de/articles/free-and-open-source-load-balancing-software-and-projects.html>

## 1.1 Auslastung

Es sollen die einzelnen Server-Instanzen in folgenden Punkten belastet (Memory, CPU Cycles) werden können. Bedenken Sie dabei, dass die einzelnen Load Balancing Methoden unterschiedlich auf diese Auslastung reagieren werden. Dokumentieren Sie dabei auftretenden Probleme ausführlich.

## 1.2 Tests

Die Tests sollen so aufgebaut sein, dass in der Gruppe jedes Mitglied mehrere Server fahren und ein Gruppenmitglied mehrere Anfragen an den Load Balancer stellen. Für die Abnahme wird empfohlen, dass jeder Server eine Ausgabe mit entsprechenden Informationen ausgibt, damit die Verteilung der Anfragen demonstriert werden kann.

### 1.3 Modalitäten

Gruppenarbeit: 2 Personen

Abgabe: Protokoll mit Designüberlegungen / Umsetzung / Testszenarien, Sourcecode (mit allen notwendigen Bibliotheken), Java-Doc, Build-Management-Tool (ant oder maven), Gepackt als ausführbares JAR

Bewertung: 16 Punkte

- 2 Load Balancing Methoden (jeweils 6 Punkte)
- Session Persistenz (2 Punkte)
- Einsatz Load Balancing Software (2 Punkte)

Viel Erfolg!

### 1.4 Quellen

- [1] "Praktische Arbeit 2 zur Vorlesung 'Verteilte Systeme' ETH Zürich, SS 2002", Prof.Dr.B.Plattner, übernommen von Prof.Dr.F.Mattern (<http://www.tik.ee.ethz.ch/tik/education/lectures/VS/SS02/Praktikum>)
- [2] <http://www.tik.ee.ethz.ch/education/lectures/VS/SS02/Praktikum/loesung2.zip>

## 2 Ergebnisse

Hier sollen die Schritte der Laborübung erläutert werden. Alle Fragestellungen der Lehrkraft müssen hier beantwortet werden. Etwaige Probleme bzw. Schwierigkeiten sollten ebenfalls hier angeführt werden.

Es kann gut möglich sein, dass Lehrkräfte hier auch noch andere Eckpunkte explizit verlangen. Diese können dann in der selben Hierarchiestufe wie die *Ergebnisse* eingeordnet werden. Viel Spass nun mit einer kleinen Übersicht von L<sup>A</sup>T<sub>E</sub>X-Elementen.

### 2.1 Tabelle

Header	Kopf
<b>Lorem</b>	Ipsum dolor sit amet, consetetur sadipscing elitr
<b>Ipsum</b>	At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus
<b>Dolor</b>	Consetetur sadipscing elitr, sed diam nonumy

Tabelle 1: Lorem ipsum dolor sit amet [1]

### 2.2 Aufzählung

- **Lorem ipsum:** dolor sit amet, consetetur sadipscing elitr
- sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat
- ut labore et dolore magna aliquyam erat, sed diam voluptua

### 2.3 Code

At vero eos et accusam et justo duo dolores et ea rebum.

```

1  try{
    gTransCur.begin();
    //Perform the operation inside the transaction
    not_registered =
        gRegistrarObjRef.register_for_courses(student_id,selected_course_numbers);
6
    if (not_registered != null)
        //If operation executes with no errors, commit the transaction
11    boolean report_heuristics = true;
        gTransCur.commit(report_heuristics);
    } else gTransCur.rollback();
16

```

```
    } catch(org.omg.CosTransactions.NoTransaction nte) {  
        System.err.println("NoTransaction: " + nte);  
        System.exit(1);  
    } catch(org.omg.CosTransactions.SubtransactionsUnavailable e) {  
21    System.err.println("Subtransactions Unavailable: " + e);  
        System.exit(1);  
    } catch(org.omg.CosTransactions.HeuristicHazard e) {  
        System.err.println("HeuristicHazard: " + e);  
        System.exit(1);  
26 } catch(org.omg.CosTransactions.HeuristicMixed e) {  
        System.err.println("HeuristicMixed: " + e);  
        System.exit(1);  
    }  
}
```

Listing 1: Implizite Transaktion [1]

## Literatur

- [1] A.S. Tanenbaum and M. Van Steen. *Verteilte Systeme: Prinzipien und Paradigmen*. Pearson Studium. Addison Wesley Verlag, 2007.

## Tabellenverzeichnis

1	Lorem ipsum dolor sit amet [1] . . . . .	3
---	--	---

## Listings

1	Implizite Transaktion [1] . . . . .	3
---	-------------------------------------	---

## Abbildungsverzeichnis