

PROJEKTARBEIT

Titel der Arbeit

Max Mustermann

Entwurf vom 20. Oktober 2016



PROJEKTARBEIT

Titel der Arbeit

Max Mustermann

Aufgabensteller: Prof. Dr. Claudia Linnhoff-Popien
Prof. Dr. Max Mustermann
Prof. Dr. Max2 Mustermann2

Betreuer: Betreuer Name1
Betreuer Name2
Betreuer Name3

Abgabetermin: 1. Januar 2099



Hiermit versichere ich, dass ich die vorliegende Projektarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 1. Januar 2099

.....
(*Unterschrift des Kandidaten*)

Abstract

[illegible]

Inhaltsverzeichnis

1 Einleitung

Dies ist der L^AT_EX Rahmen zur Bearbeitung von Bachelor-, Master-, Projekt- und Diplomarbeiten. Alle relevanten Dateien befinden sich im Verzeichnis `text`.

1.1 Unterverzeichnisse und Dateien

Das Verzeichnis `text` beinhaltet weitere Unterverzeichnisse und Dateien, die den Rahmen charakterisieren.

1.1.1 `main.tex`

Diese Datei stellt die zentrale Konfigurationsdatei für den Rahmen dar. Unter anderem müssen hier Informationen über die Aufgabensteller, Betreuer, die Art der Arbeit sowie deren Title eingestellt werden. Hier können auch weitere Pakete eingebunden werden. Die Datei ist dokumentiert und sollte selbsterklärend sein.

1.1.2 `hyphenation.tex`

Manche Wörter werden von L^AT_EX nicht (ordentlich) getrennt. Diese können in dieser Datei mit deren Trennungsstellen hinzugefügt werden.

1.1.3 `Makefile`

Um das Dokument zu erstellen muss man den Aufruf `make all` tätigen. Dabei werden einige temporäre Dateien erstellt sowie die Datei `main.pdf` die das entsprechende Dokument enthält. Mit dem Aufruf `make clean` werden alle temporären Dateien sowie die Datei `main.pdf` gelöscht. Sie können die Datei `Makefile` ihren Anforderungen entsprechend erweitern.

1.1.4 `text`

Es bietet sich an für verschiedene Kapitel eigene Quelldateien zu pflegen. Diese sollten sie alle im Ordner `text` ablegen. Wie ein Kapitel eingebunden wird, kann man aus dem Beispiel in der Datei `main.tex` ablesen. Das Verzeichnis `text` beinhaltet zudem die Datei `abstract.tex`. In diese Datei soll eine kurze Zusammenfassung (ca. eine halbe Seite) der Arbeit eingetragen werden. Die Datei `appendix.tex` kann verwendet werden um einen Anhang zu generieren.

1.1.5 `pictures`

Hier müssen sie alle Grafiken ablegen, die sie in ihrem Dokument einbinden wollen. Es sind nur die Formate PDF, PNG und JPEG erlaubt (GIF ist möglich, wird aber nicht empfohlen).

1.1.6 bibliography.bib

In diese Datei müssen alle Referenzen eingetragen werden, die innerhalb ihrer Arbeit zitiert werden. Verwenden sie zur Verwaltung ihrer Referenzen einen geeigneten Editor z.B. *JabRef* (<http://jabref.sourceforge.net/>).

1.1.7 mdsg.sty

Hierbei handelt es sich um das Stylefile, das das Erscheinungsbild des Dokuments lenkt. In dieser Datei sollten in der Regel keine Veränderungen notwendig sein.

1.2 Beispiele

Es gibt eine Unmenge an \LaTeX Tutorials und Dokumentationen, die guten Einstieg in das Arbeiten mit \LaTeX ermöglichen. Im Folgenden werden aber ein paar undokumentierte Minimalbeispiele gegeben, die den direkten Einstieg ermöglichen. Betrachten sie den Quelltext, um die Beispiele nachzuvollziehen.

1.2.1 Zitate

Wir zitieren hier eine Quelle von James Aspnes et al [?], die in der Datei `bibliography.bib` steht.

1.2.2 Listen

Es gibt verschiedene Möglichkeiten Listen zu erstellen, z.B. ohne Nummerierung...

- Das ist der erste Punkt,
 - das der erste Unterpunkt,
 - das der zweite Unterpunkt,
- das der zweite, und
- das der dritte Punkt.

...oder mit Nummerierung...

1. Das ist der erste Punkt,
 - a) das der erste Unterpunkt,
 - b) das der zweite Unterpunkt,
2. das der zweite, und
3. das der dritte Punkt.

1.2.3 Referenz auf anderen Text

Es ist auch möglich auf andere Stellen im Text z.B. Kapitel ?? zu verweisen.

1.2.4 Hoch- und tiefgestellter Text

Man kann Text tiefstellen indem man `\textsubscript` verwendet, z.B. ergibt

`text\textsubscript{tiefgestellt}`

den Text `texttiefgestellt`. Das selbe funktioniert mit `\textsuperscript` verwendet, z.B. ergibt

`texthochgestellt`

`texthochgestellt`

1.2.5 Tabellen

Es gibt schöne Möglichkeiten Tabellen einzubinden wie z.B. Tabelle ??.

Parameter	Value	(Unit)	Available for Chord
Query timeout	10	seconds	✓
Republish timeout	300	seconds	✓
Stabilize timeout	5	seconds	✓
Fix fingers timeout	30	seconds	✓
Message timeout	1	second	✓
Connect timeout	10	seconds	✓
Ping superpeer timeout	5	seconds	×
Cost-Optimality estimation timeout	20	seconds	×
Significance for change in number of superpeers	10	percent	×
Significance for change in estimations	10	percent	×
Number of permanent superpeers	32	nodes	×
Mean number of peers	1000	nodes	✓
Mean number of lookups per hour	60	queries	✓
Mean number of shared InfoProfiles per node	20		✓
Identifier space	16	bits	✓
Direct insertion acknowledgment	true	bool	×
Direct query responses	true	bool	×
Force query resolution	true	bool	✓

Tabelle 1.1: Common simulation parameter settings.

1.2.6 Bilder

Man kann sehr einfach Bilder einbinden so wie z.B. in Abbildung ??. Es lassen sich auch mehrere Bilder nebeneinander platzieren wie z.B. in Abbildung ?? zu sehen ist.

1.2.7 Programm Code

Eine elegante Möglichkeit, Programmtext einzubinden, lässt sich mit dem listings-Paket erreichen. Das HelloWorld Programm aus Listing ?? hat in Zeile ?? übrigens einen Programmierfehler.

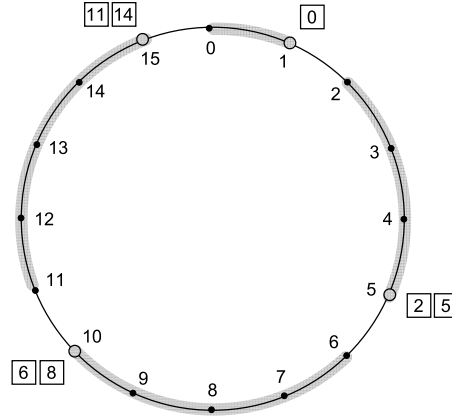


Abbildung 1.1: Example of a 4-bit Chord identifier circle. The responsibility ranges for each peer are accentuated in light gray

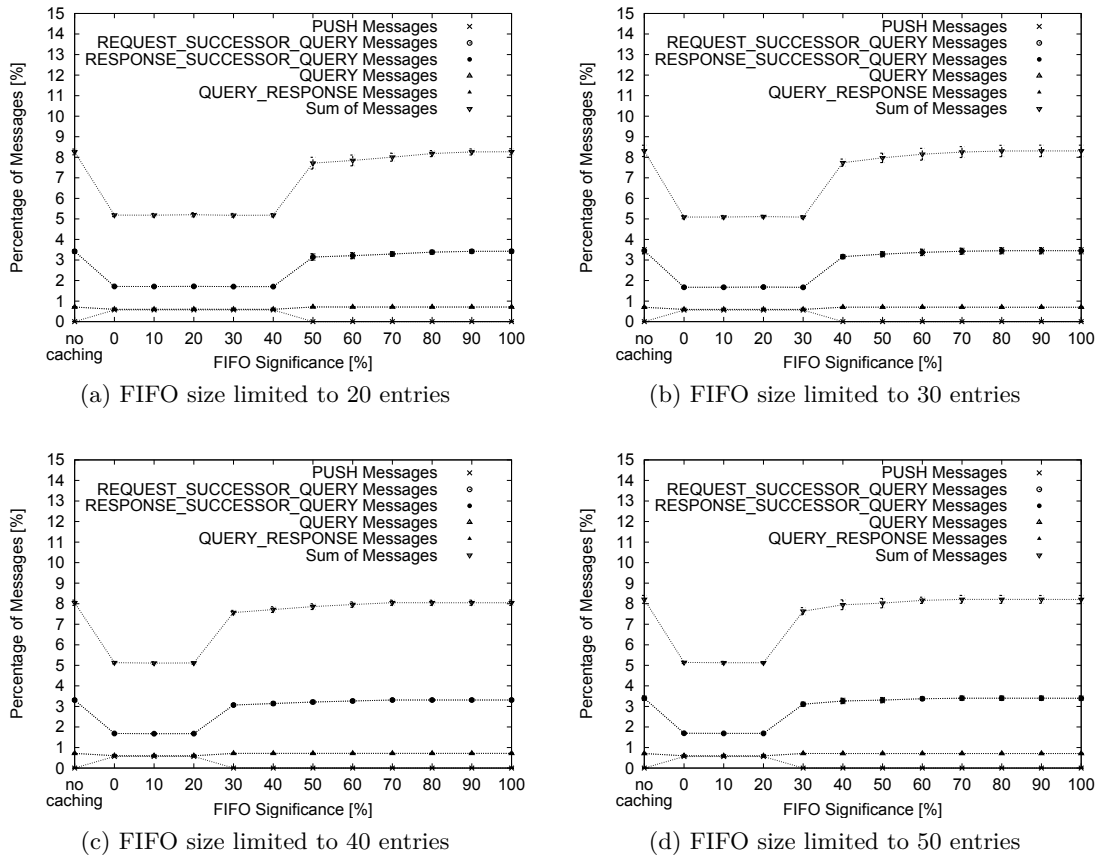


Abbildung 1.2: Observed message fractions and 95% confidence intervals for Chord without the influence of churn. The FIFO capacity varies from 20 (??) – 50 (??) entries (decadic steps).

Listing 1.1: Hello World

```

1 public class HelloWorld {
    public static void main(String [] args) {
3         System.out.println("Hello , World");
    }
5 }

```

1.2.8 Fußnoten

Wenn man auf Google ¹ verweisen will, bietet sich statt einer gesonderten Referenz auch einfach eine Fußnote an.

1.2.9 Formeln

Man kann mit L^AT_EX sehr schön Formeln erzeugen:

$$L_P(k) = R_P^{orig}(k) + \sum_{i=0}^n 2 * R_P^i(k)$$

¹<http://www.google.com>

A Beispiel Anhang

```
/*
 * This code serves the initialization of an auxiliary probability array.
 * The array holds at each position a pre-calculated probability for the index
 * of that position. The probability reflects the Zipf-distribution for the
 * corresponding indexes
 */
Set zipfExponent to 1.4
Set sum to 0
Set maxInteger to 65535
FOR i = 0 to maxInteger
  probArray[i] = 1/pow(i + 1, zipfExponent)
  Set sum = sum + probArray[i]
END FOR
FOR i = 0 to maxInteger
  Set probArray[i] = probArray[i]/sum
END FOR
FOR i = 1 to maxInteger
  Set probArray[i] = probArray[i] + probArray[i-1]
END FOR

/*
 * This code gets called in case a Zipf-distributed number is required. It
 * iterates over the probability array until the chosen random number v
 * is less than the value stored at the current array position i. The value of
 * the array position will be returned as the calculated Zipf-distributed
 * number
 */
Set v to a random number between 0 and 1
FOR i = 0 to maxInteger
  IF v < probArray[i] THEN
    RETURN i
  END IF
END FOR
RETURN 0
```


Abbildungsverzeichnis

Tabellenverzeichnis

Listings