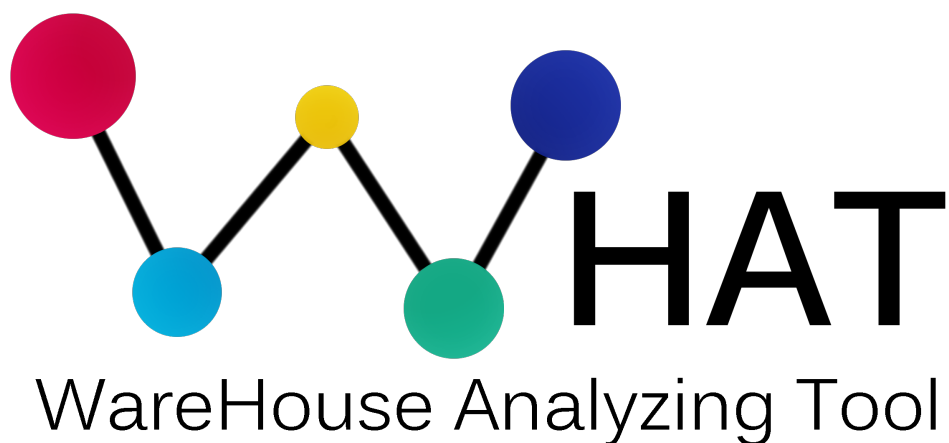


## QA & Testing



February 21, 2013

Functional Specification	<b>Alexander Noe</b>
Design	<b>Jonathan Klawitter</b>
Implementation	<b>Anas Saber</b>
QA / Testing	<b>Nikolaos Alexandros Kurt Moraitakis</b>
Final	<b>Lukas Ehnle</b>

E-Mail: [pse10-group14-ws12@ira.uni-karlsruhe.de](mailto:pse10-group14-ws12@ira.uni-karlsruhe.de)

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Implementation mistakes</b>	<b>4</b>
2.1	Parsing . . . . .	4
2.1.1	Task determination . . . . .	4
2.1.2	Negative or too high poolsize . . . . .	4
2.1.3	NullPointerExceptions when correct file doesn't exist . . . . .	4
2.1.4	.trim() . . . . .	4
2.1.5	Anonymous Proxy . . . . .	4
<b>3</b>	<b>Configuration file</b>	<b>5</b>
<b>4</b>	<b>Tests</b>	<b>6</b>
4.1	Automatic tests . . . . .	6
4.1.1	Parsing . . . . .	6
4.2	Test scenario . . . . .	6
<b>5</b>	<b>Performance</b>	<b>7</b>
5.1	Starting . . . . .	7
5.2	Loading . . . . .	7
5.3	Extracting . . . . .	7
<b>6</b>	<b>Stability</b>	<b>8</b>
6.1	Nonfunctional requirements . . . . .	8
6.2	Stress testing . . . . .	8

## 1 Introduction

This document

## 2 Implementation mistakes

### 2.1 Parsing

---

#### 2.1.1 Task determination

---

When having too many threads (10+), sometimes a thread stopped working. This caused the counter for finished tasks to be stuck under the number of threads and the program not to get finished. So we implemented a watchdogtimer which replaces a thread by a new one, if said thread doesn't finish a line for a predefined amount of time. (standard = 2 seconds)

---

#### 2.1.2 Negative or too high poolsize

---

The parser works with a pool of threads, which are created when parsing. In the Parser-Mediator there is a variable poolsize, which is the used poolsize for parsing. When using a negative or a really high number (tried Integer.MAX\_VALUE), the program crashes. So we implemented a check for the poolsize, which looks if it is between 1 and 50 and stops parsing if it isn't.

---

#### 2.1.3 NullPointerExceptions when correct file doesn't exist

---

*sp\_parser.Logfilehadacheck, if the file which is given to it is actually existent. It used the file-class from java and got a NullPointerException, which was caught and used as an indicator, that there*

This worked, but to improve our program, we used a class called FileHelper, which creates the file-object one time and returns it when needed, because other parts of the program need the file too. It didn't return a NullPointerException when the file doesn't exist, so this check got overrun and the parser "thought" that there is a correct file which wasn't. This caused the program to crash. Fixed by checking for null instead of catching a NullPointerException

---

#### 2.1.4 .trim()

---

In some cases the logfile contains unneeded whitespaces, which produced false data. This was fixed by adding the .trim()-command in StringRow.split() and StringMapRow.split()

---

#### 2.1.5 Anonymous Proxy

---

Sometimes the city- and country-name of an IP returned Anonymous Proxy, which is now replaced by "other" to fit in with the other undefined countries and cities.

...



### 3 Configuration file

This section is about the part of the system reading the configuration file

O\_o

## 4 Tests

### 4.1 Automic tests

---

#### 4.1.1 Parsing

---

- smallParseTestStandardSize - parse 10 lines, standard (5) poolsize
- smallParseTestSize10 - parse 10 lines, poolsize 10
- smallParseTestTooBig - parse 10 lines, poolsize 100 -> Error #2: ParserMediator.poolsizeParsing is bigger than 50.
- negativePoolsize - parse 10 lines, poolsize -2 -> Error #1: ParserMediator.poolsizeParsing is smaller than 1.
- nonexistentFile - try to parse nonexistent file -> Error #5: The path is wrong.
- wrongFile - try to parse from an empty .txt file -> Error #5: The path is wrong.
- flawedFile - try to parse from a file with a typo in "type" -> Error #11: The configuration file got a different format.
- mediumParseTestStandardSize - parse 1000 lines, standard (5) poolsize
- mediumParseTestSize50 - parse 1000 lines, poolsize 50
- bigParseTestStandardSize - parse 30k lines, standard (5) poolsize
- veryBigParseTestStandardSize (IGNORED) - parse a whole month, standard (5) poolsize, ignored because it may take hours to finish.

### 4.2 Test scenario

## 5 Performance

We should test the performance and then check how we could get it better.

### 5.1 Starting

maybe?

### 5.2 Loading

Parsing of 1k lines with old upload schema:

1. 19048ms
2. 22033ms
3. 11020ms

Parsing of 1k lines with new upload schema:

1. 11016ms
2. 10017ms
3. 11017ms

### 5.3 Extracting

## **6 Stability**

### **6.1 Nonfunctional requirements**

### **6.2 Stress testing**