

# OXID eSales OXID eShop Quality Report

June 2010

## Copyright

Copyright © 2010 OXID eSales AG, Germany  
All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the prior written permission of OXID eSales AG. Decompilation of source code, piracy as well as transfer to a third party is not allowed. Use of this content for any other purpose is expressly prohibited and may result in severe civil and criminal penalties. Violators will be prosecuted to the maximum extent possible. The information contained within this document was created accordingly to the state of the art technology. OXID eSales AG assumes no liability or warranty for the accuracy, the completeness or reliability of any content. In spite of all efforts, the occurrence of mistakes cannot be eliminated completely. Thus we appreciate any feedback.

## Conventions

The following conventions are used:

`proportional font with grey background`

For user input, code and URLs

*Grey italic*

For filenames, file paths and other italic highlighting

**Bold**

For input fields and navigation paths

**Bold red**

For warnings and important hints

## Contact

OXID eSales AG  
Bertoldstrasse 48  
79098 Freiburg  
Germany

Fon: +49 (761) 36889 0  
Fax: +49 (761) 36889 29

Represented by the Supervisory Board: Roland Fesenmayr (CEO), Andrea Seeger  
Registered Office Freiburg i. Br.  
Commercial Register Freiburg  
No. HRB 701648

**Table of contents**

Copyright .....2

Conventions.....2

Contact .....2

Table of contents.....3

1 WHAT'S NEW .....4

    1.1 More precise file type filtering for the statistics .....4

    1.2 Publishing the unit tests .....4

    1.3 Different tools.....4

    1.4 Source code base for the metrics has changed .....4

2 STATISTICS.....5

    2.1 Source code progress .....5

    2.2 Code coverage .....6

    2.3 Cyclomatic complexity and code sniffer .....7

    2.4 Bug tracking/fixing process .....8

3 OXID eShop Performance .....9

    3.1 Test method .....9

    3.2 Results.....10

    3.3 Conclusion.....12

## 1 WHAT'S NEW

For various reasons we changed some aspects of how to measure our metrics. In this quality report, you will see different statistic values about some previous OXID eShop versions in various charts and tables, compared with earlier quality reports. Thus, we can compare current metrics as well as possible with older versions. This will lead you to a better understanding about our development process and quality assurance results.

However, not everything is directly comparable with older quality reports. If we added these changes incremental into each quality report we would never be able to compare the results with the previous versions. Therefore we decided to integrate all changes in a single step. At least for the near future we will have comparable quality reports again. The reasons to change our metrics measurement in detail are:

### 1.1 More precise file type filtering for the statistics

Up till now in some metrics non-PHP files were also counted into the total amount of files. This affected the proportion in certain statistics. This is now cleaned up, making it more precise. Beginning with this release, we only include the PHP source code files for the metrics directly related to source code quality

### 1.2 Publishing the unit tests

We now publish our unit tests source code. This actually costs us some effort to achieve. In the development we have a "unified" version of our product and its unit tests. The editions are created by parsing the unified version during deployment, from which then the release package files are generated.. The continuous integration process uses our deployment system and generates edition packages on the fly on which it will then run the automated tests. The unit tests itself however were still unified and was told by the continuous integration server which edition it's testing. To publish the unit tests, we needed to create similar packages of our unit test source code according to their corresponding editions – Enterprise, Professional and Community. Thus, we had to rework our deployment and packaging system to also incorporate the unit tests. A side effect of this is that we now have different statistical information for each edition unit tests. Thus to make comparison test info with metrics, we will apply the previous values (collected on combined unit test source code) for all separate editions (for previous versions). So in the statistical reports the test data for EE, PE and CE up until version 4.3.0 have the same values. Take this into account when comparing.

### 1.3 Different tools

We now use *JMeter* as our performance testing tool instead of *siege*. Older information can't be compared, because of differences from these tools.

### 1.4 Source code base for the metrics has changed

Traditionally we consider the shop as logically divided into 2 parts. This is mostly used in the requirements (decision making) process. It's not a technical perspective. This division is based

on primary target audience interacting with the shop. There is what we call the core or frontend, everything intended to be used by shop visitors and is reachable from the "outside world". The other part is the admin area, which is only available for the shop owner. During the development of very first 4th generation OXID eShop version (version 4.0.0.0) we refactored the core of the shop, which is, like stated before mostly frontend. More specific, this is the source code in the *core* dir. When we started to gather statistical information about changes and quality of development we did it only for the core. After the initial release we worked on the admin area as well as in the *views* source code. So we started to measure metrics about this source code, too. From this quality report on, we will publish our metrics based on all php files inside the OXID eShop directory structure, with the exception of the third party libraries which we exclude. One other exception is the unit testing (think code coverage) which is done on *admin/*, *core/* and *views/* directories only.

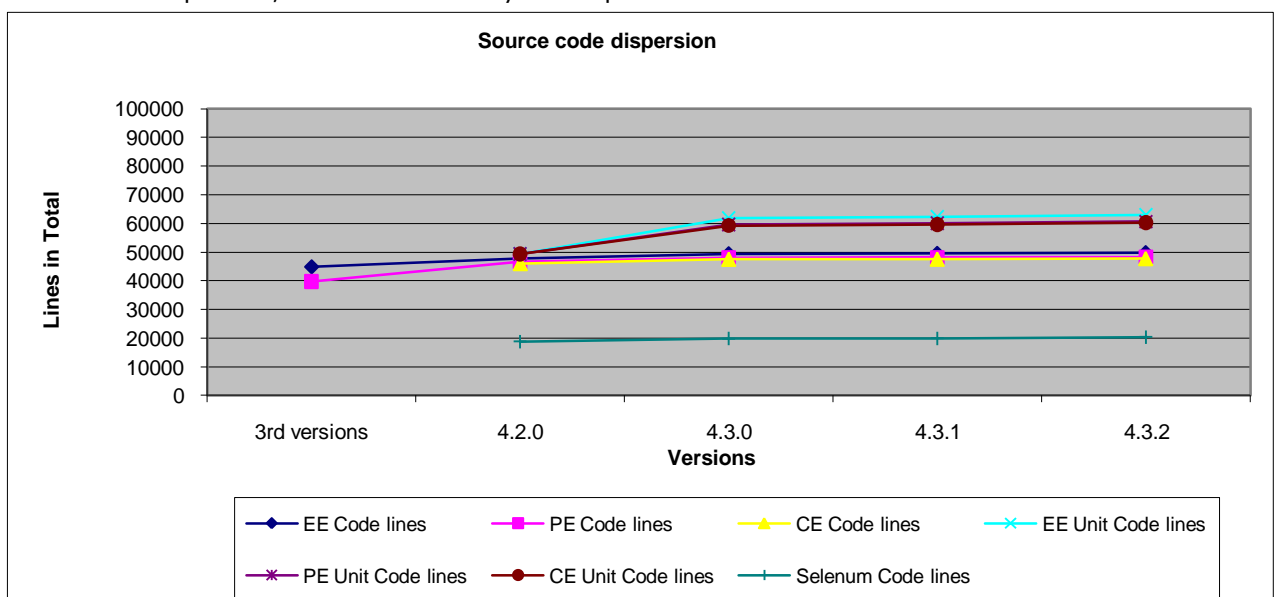
## 2 STATISTICS

### 2.1 Source code progress

Below is a table with amounts of total lines in source code (only php-files), which is managed by the OXID eShop development team:

Section	Total # of Lines	Growth since version 4.3.0, %	Source code	Comments
OXID eShop EE	130374	0.75	49814	49229
OXID eShop PE	126265	0.58	48314	47698
OXID eShop CE	128933	0.58	47771	51273
Unit tests EE	126870	1.57	62958	29038
Unit tests PE	122096	1.46	60627	27898
Unit tests CE	124485	1.44	60203	30979
Selenium tests	25426	6.12	20373	3029

Source code dispersion, what is handled by development team:

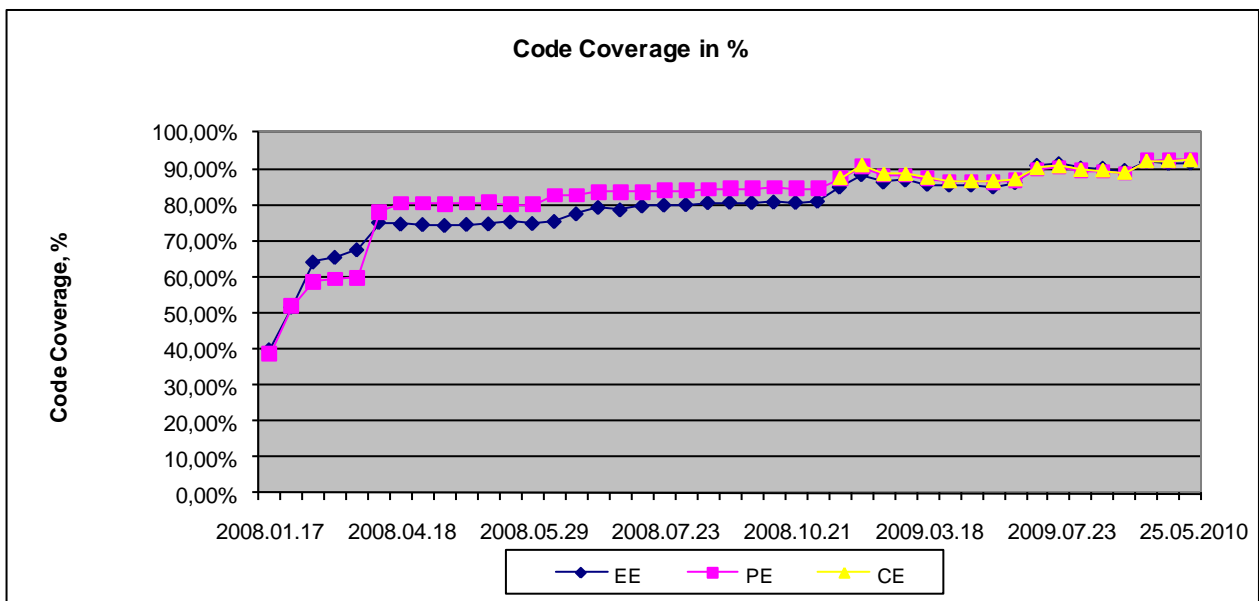
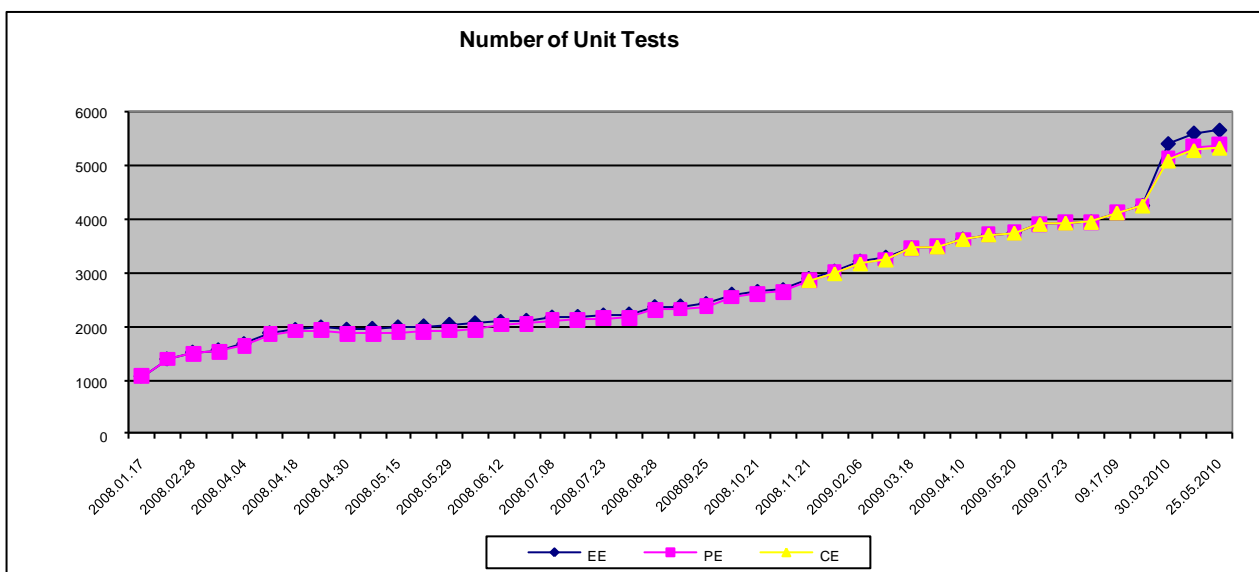


## 2.2 Code coverage

Comparing to version 4.3.0, the total number of source code lines in OXID eShop and unit tests keeps increasing. The already high code coverage continues to improve, although only by a very small percentage:

	Amount of tests	Code Coverage %
EE	5661	91.49
PE	5390	92.47
CE	5344	92.37

Our effort in this place you can see in these charts:



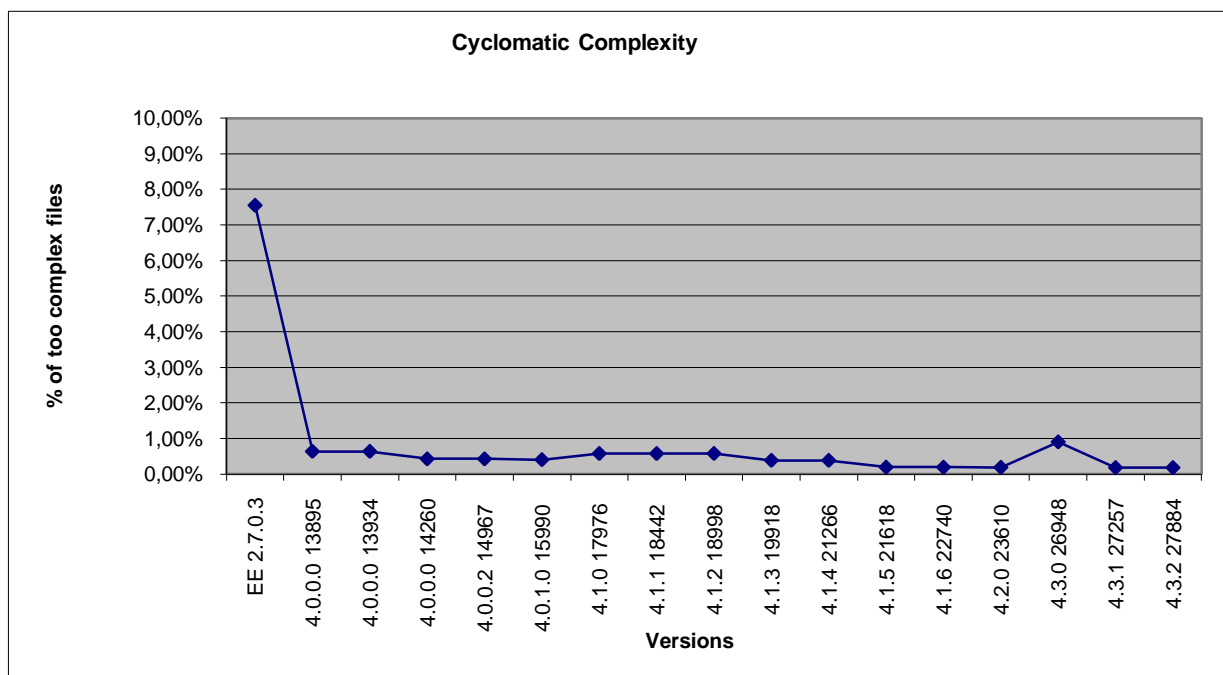
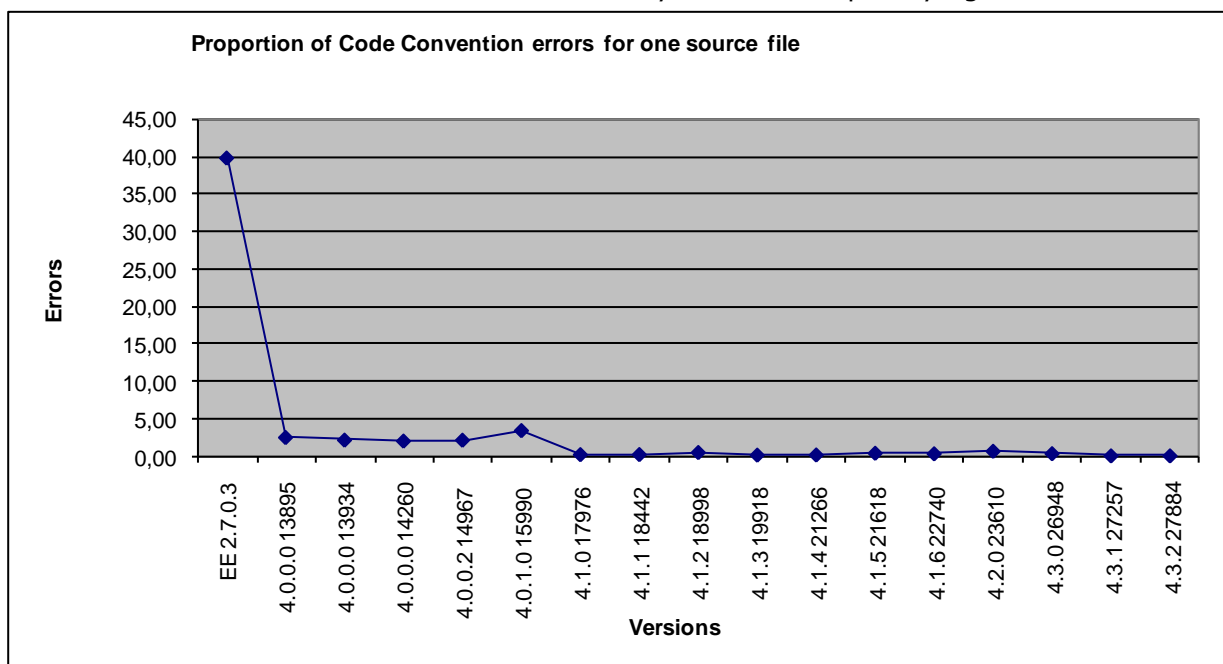
.. and, since we have published the unit tests since 4.3.0, you can check this for yourself.

### 2.3 Cyclomatic complexity and code sniffer

Though source code in OXID eShop grows continuously, we always trying to keep it clean and simple as possible. With each new release we go further from previous versions and source code gets more and more complicated. Partly because workarounds are needed for backward compatibility and we still keep deprecate source code. But even with this we still manage to keep cyclomatic complexity at low level: only 1 source code file exceeds the allowed limit (20).

But this will be rectified when we clean the source code from deprecated code.

See here the charts about codesniffer errors and cyclomatic complexity figures:

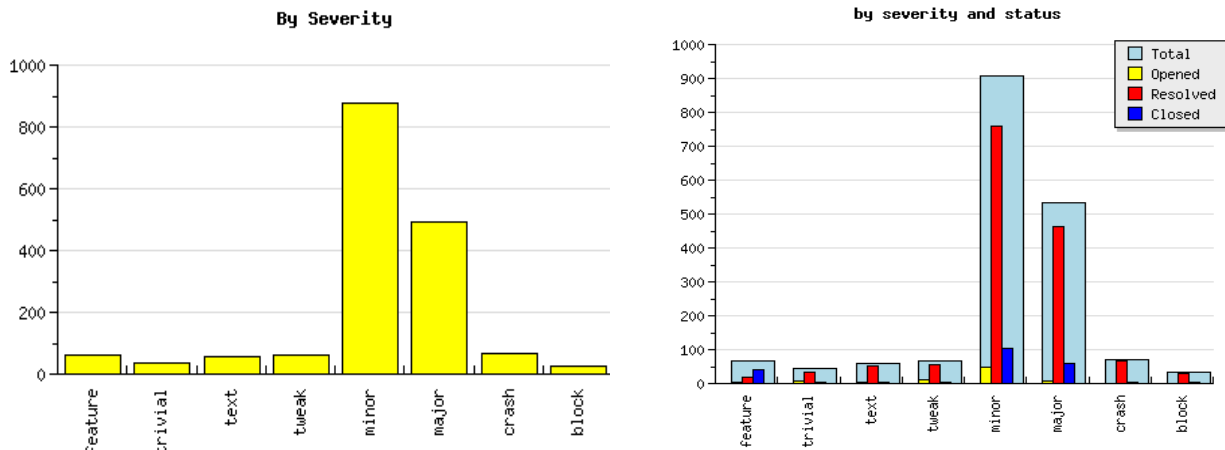


### 2.4 Bug tracking/fixing process

OXID eShop 4.3.0 version release took quite a long time as it included lot of new features and changes. Thus, keeping an eye on very important bugs, we had focused on new features development. Now during versions 4.3.1 and 4.3.2 release we are working on fixing issues, though still a number of these are left open:

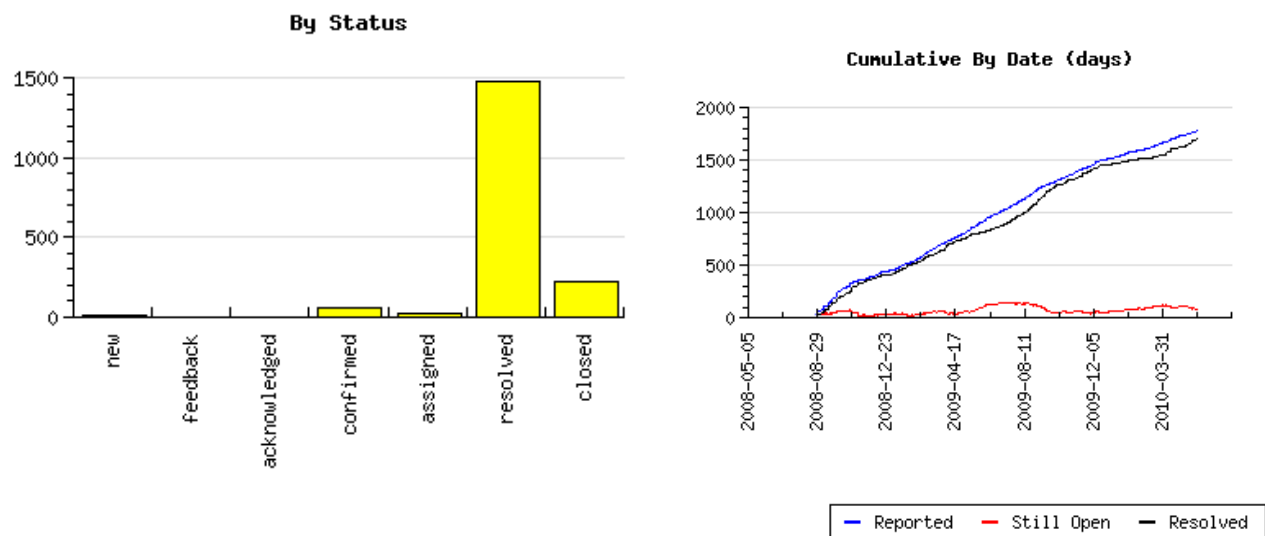
By Status	Open	Resolved	Closed	Total:
Total	80	1390	216	<b>1686</b>

Here is the distribution of bugs by severity and priority:



We kept the proportions of bug chart: peak in the charts is more to left side (severity=minor and priority=normal). As a rule, we release only when all high priority bugs are fixed.

The next two charts show continuous bug fixing process and how development reacts to opened bugs:



### 3 OXID eShop Performance

#### 3.1 Test method

We also made a major change in measuring OXID eShop performance and its behavior during load tests: we moved from *siege* to *JMeter* as our preferred performance test tool. We wanted to make the tests more realistic and have more realistic testing scenarios. We missed the possibility to define the delay between requests. Such delay makes tests more realistic and similar to real user behavior and *JMeter* has such possibility. In tests with *JMeter* we added 3 sec delay between each request to shop. See the results below.

Remember that these results can't be compared with earlier quality reports!

The test cases remained the same as during previous tests:

- Performance and load testing is done by calling the 4 most important pages of OXID eShop with a particular amount of concurrent users (generated by *JMeter* tool). These pages are:
  - Main page,
  - Details view,
  - Category view, and
  - Search
- Each concurrent user is started with 3 sec delay.
- Each user makes 50 requests (reduced from 100, as it gives the same result, but reduces test time to half) with 3 sec delay to OXID eShop by prepared URLs.

The performance (or benchmark) test is when OXID eShop is besieged by one virtual user. Load testing is when OXID eShop is besieged by more than one virtual user (10, 20, etc.). As the result the average response time of the entire test is taken.

Performance and load testing also were performed on Fatboy DB (big data base with realistic data). The amount of data in Fatboy DB is:

- articles:       ~45,750
- attributes:    12,537
- categories:    3,242

### 3.2 Results

Additionally, we measured the EE with caching ON and OFF, to check the caching benefit.

We changed the PHP version on testing server from 5.2.6 to 5.2.13, and apparently on this version the shop runs much faster. Of course all tests here were run on the same PHP version to enable a fair comparison

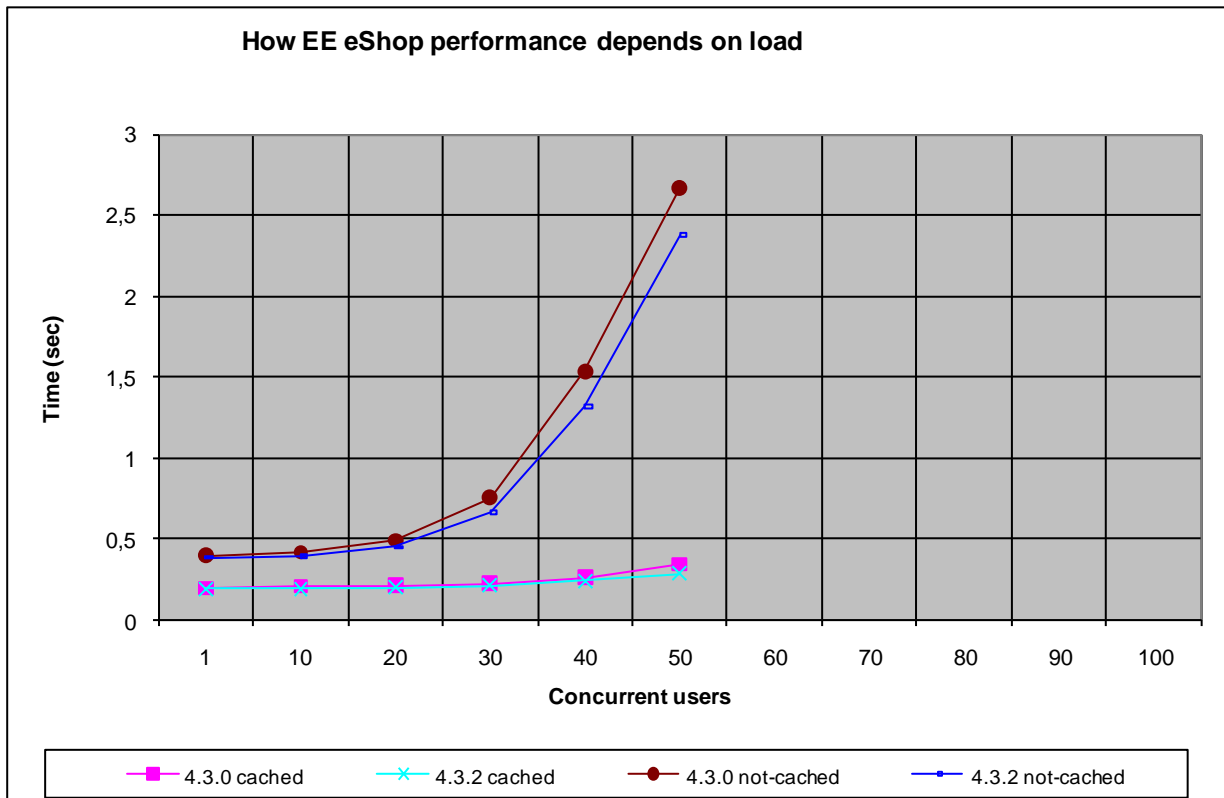
Currently, the performance status of OXID eShop 4.3.2 is:

- EE without Caching:* EE 4.3.2 is about **9%** faster than EE 4.3.0.
- EE with Caching:* EE 4.3.2 is about **7%** faster than EE 4.3.0.
- EE with Fatboy DB:* EE 4.3.2 is about **10%** faster than EE 4.3.0.
- PE:* PE 4.3.2 is about **11%** faster than PE 4.3.0.

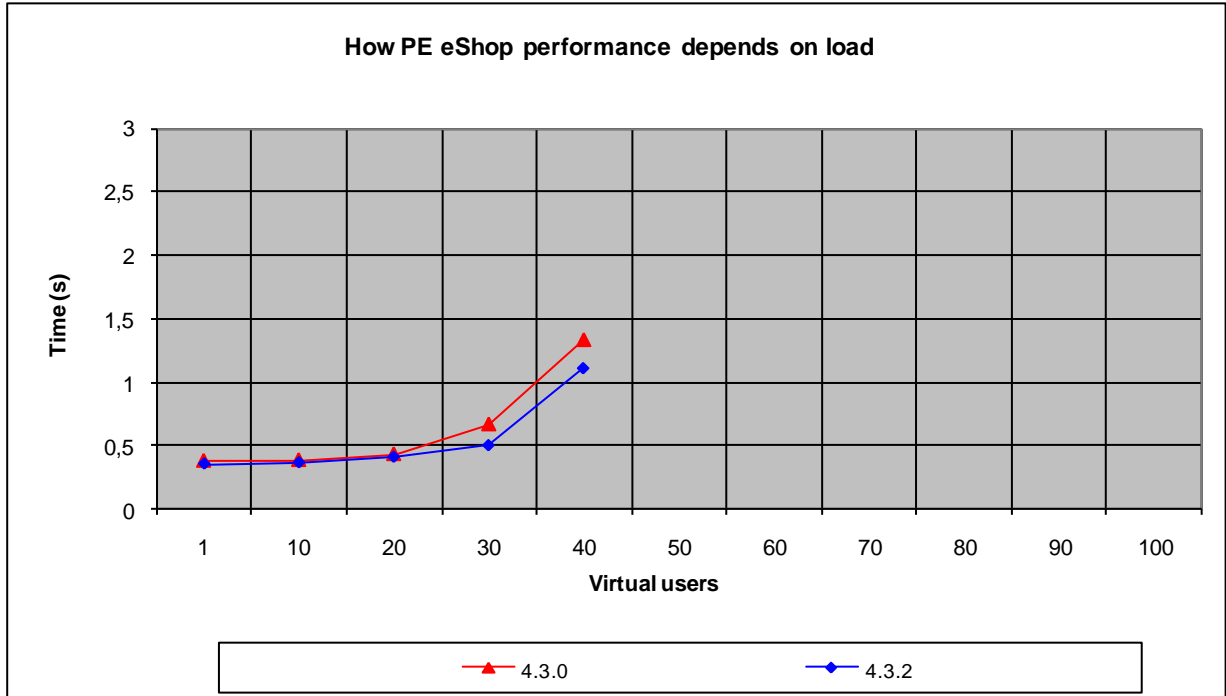
The average absolute response time for one concurrent request is:

- *EE without caching:* **0.381 sec** (in EE 4.3.0 was 0.396 sec)
- *EE with caching:* **0.188 sec** (in EE 4.3.0 was 0.193 sec)
- *EE with Fatboy DB:* **0.343 sec** (in EE 4.3.0 was 0.353 sec)
- *PE:* **0.356 sec** (in PE 4.3.0 was 0.379 sec)

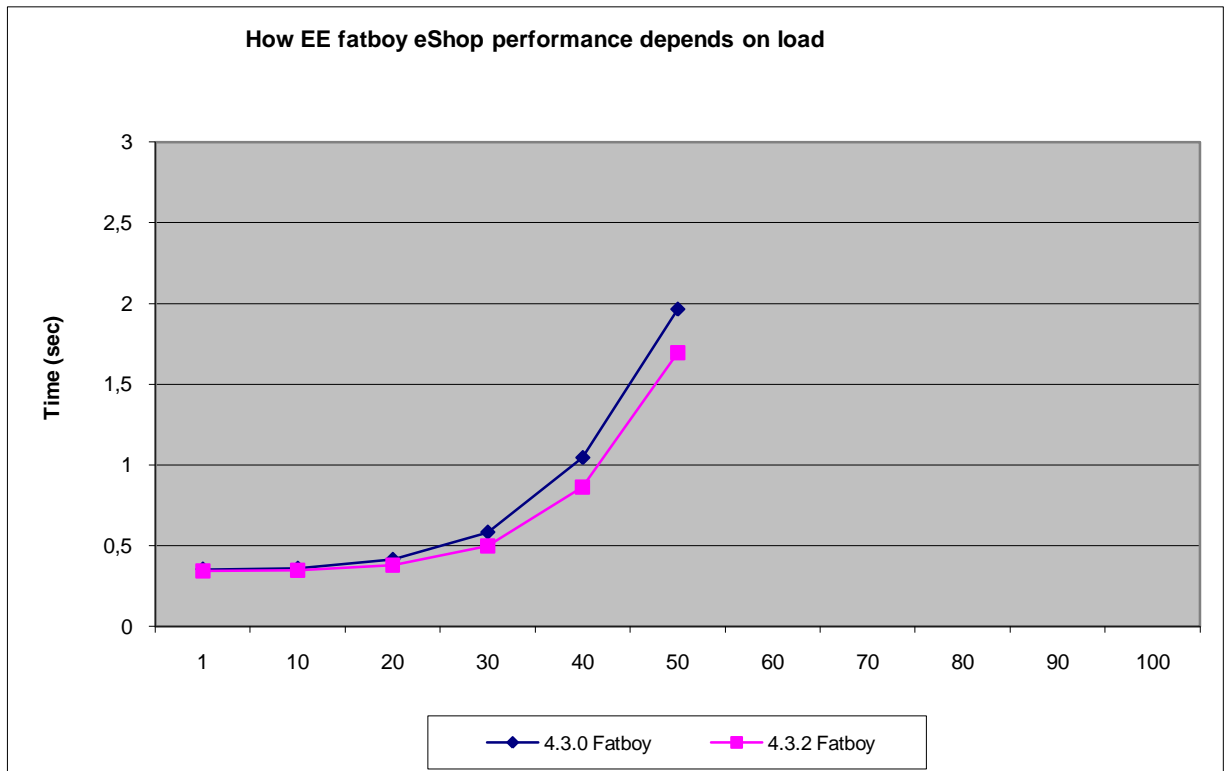
Load testing results when OXID eShop is receiving concurrent requests:  
For EE:



For PE:



For EE with Fatboy DB:



### 3.3 Conclusion

We can conclude that adding 3 sec delay between requests makes test cases more realistic and also shows that the OXID eShop is more resistant to a high load of concurrent users, than testing with the former tool (*siege*) showed us.

From these results we see that:

- The response times significantly reduced, comparing to previous tests. I.e. response time of EE 2.7.0.3 under 30 concurrent users load was 8.25 sec, now it's only 0.662 sec, for PE 3.0.4.1 – it was 7.57 sec, now it's – 0.502 sec. This is affected also by changed PHP version.
- The threshold, when tests start failing (the requested pages cannot be displayed because there is an overload on server) has shifted much further. I.e. with previous tests this boundary for older versions (EE 2.7.0.3, PE 3.0.4.1) and OXID eShops with Fatboy DB was at 40-50 concurrent users, now it is at 60-70. For 4th versions OXID eShops this boundary was shifted from 80-90 to 110-120 concurrent users. So it tells that OXID eShop is about 30% more resistant to high load then we thought before.