TechMetrix research

# Performance White Paper
# Zend Cache

A Technical White Paper by
TechMetrix Research
Audience: Project Managers, Technical Architects, and Developers

www.techmetrix.com

# Table of contents



# Table of figures

# 1. Introduction: Zend Cache

The Zend Cache integrates with the Zend Engine, which is the "kernel" of PHP 4.0.

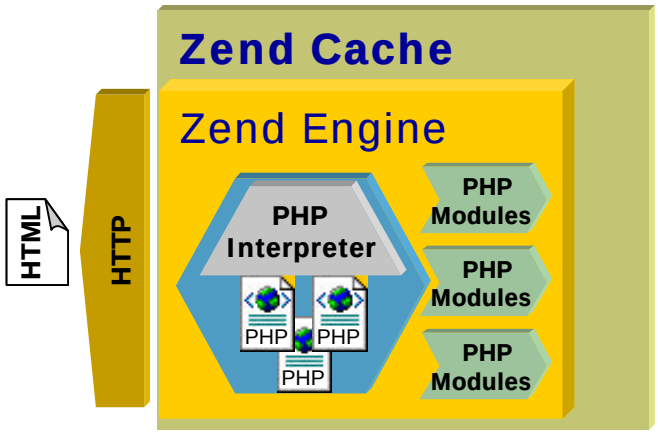


*Figure 1: Zend Engine and Cache*

The Zend Engine is a Scripting engine that interprets the PHP code. Each time a user requests a page (e.g.: http://www.mysite.com/home.php), the PHP page is loaded from the file system, interpreted and compiled then executed to deliver the final HTML page.

The Zend Cache performs optimizations and various types of caching, using advanced techniques that yield much better performance. PHP scripts are compiled once and stored in the Zend Cache's allocated memory. Storing compiled scripts using the Cache saves from having to recompile a script every time it is called. It also makes it possible to bypass time-consuming disk accesses.

Zend Cache is available as a separate product, and integrates with the Zend Engine, which is part of PHP 4.0.

# 2. Objectives and Methodology

## ➲ Objective

This Performance White Paper is an unbiased analysis of the Zend Cache product. It provides practical performance figures and detailed comments about the product's behavior in high-stress situations.

The purpose of this benchmark is to validate and measure Zend's claims about the performance optimizations provided by Zend cache.

## ➲ Methodology

Zend provided TechMetrix Research with the Zend Cache product and access to technical support (we can note that we did not need to use this technical support).

TechMetrix Research chose the application to be used for Zend Cache performance measurements.

The PHP – Zend Cache platform was tested using an existing application developed by Groupe SQLI. This application, called CapCity, and based on the OSS framework Interligo (www.interligo.org) is a Knowledge Management Portal.

No specific optimization was carried out on the application before or during the tests. Basically, TechMetrix Research performed a series of measurements, classed as "Before" (without Zend Cache) and "After" (with Zend Cache).

This application features various functionalities:

- Consultation of documents.
- Creation of documents and topics.
- Search engine.
- Access statistics.

In terms of design, the application is based on a page of PHP script which, depending on the parameters passed in the URL, will include a number of files or libraries.

# 3. Performance Summary

There is no question as to the advantages brought by Zend Cache:

- On average, it enabled the performances of the application under test to be doubled.
- It improved reliability of the application when subjected to heavy workloads.

Without Zend Cache, the server was incapable of serving to more than 150 concurrent users. By activating Zend Cache, the server became able to serve pages up to a level of 200 robots (maximum for tests), with a constant transactional rate and 100% reliability.

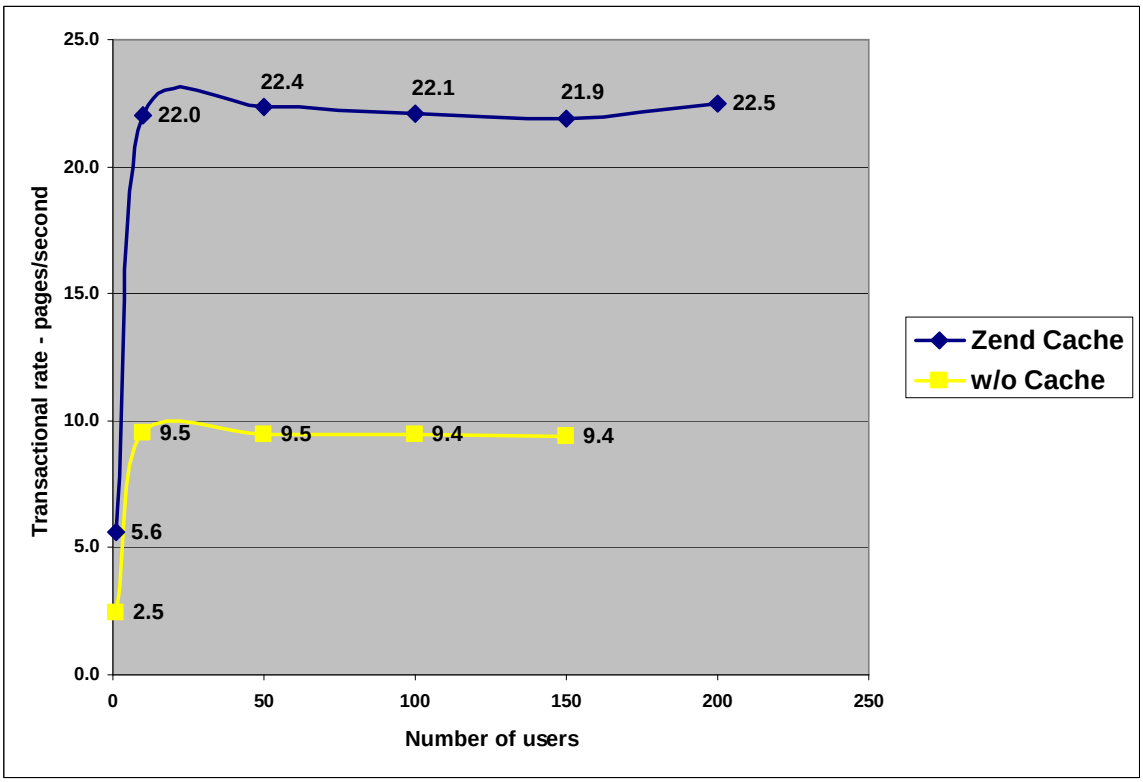


*Figure 2: Transactional Rate: Pages/Second with and without Zend Cache*

# 4. Performance measurements

## 4.1 Summary of results

The application was first tested without using the Cache, up to a level of 150 robots. Beyond this point, the application experienced reliability problems, characterized by saturation of the server memory and therefore extensive use of swapping.

Once Zend Cache was activated, the tests could be increased to 200 robots, and we observed efficient consumption of the server's system resources, as well as reasonable response times.

The configuration of Zend Cache in the php.ini file, was carried out as follows in order to obtain the best possible performances:

```
zend_cache.use_cwd=0
zend_cache.memory_consumption=64 ; 64MB
zend_cache.validate_timestamps=0
zend_extension=/usr/local/lib/ZendCache.so
```

An additional test, in which we activated zend_cache.use_cwd, had no signficant effect on results. This option enables the actual path of the PHP scripts to be recognized, in order to avoid confusion between two scripts with the same name but stored in different directories.

The zend_cache.validate_timestamps option was not used. This is an option which enables Zend Cache to tell whether a PHP script has been updated. If it has, Zend Cache will immediately load the new version.

In addition, the Zend Optimizer was activated for all tests, as the two modules can operate alongside each other.

## 4.2 Reliability

| | Average | 1 robot | 10 robots | 50 robots | 100 robots | 150 robots | 200 robots |
|---|---|---|---|---|---|---|---|
| Zend Cache | HTTP reliability | 100% | 100% | 100% | 100% | 100% | 100% |
| | Database transactions | 100% | 100% | 100% | 100% | 100% | 100% |
| W/o Cache | HTTP reliability | 100% | 100% | 100% | 100% | 100% | 50% |
| | Database transactions | 100% | 100% | 100% | 100% | 100% | 50% |

The tests carried out without the Cache revealed reliability problems during testing, above the level of 150 robots. This was due to the saturation of the server memory caused by the application under test.

When the Cache was activated, no "Web" errors thwarted the measurements. To make sure that the correct HTML page was returned, we verified the presence of a keyword on each page. We checked the MySQL database after workload sessions; it did not show even the slightest sign of weakness: all of the database transactions were carried out successfully.

## 4.3 Response Time Results

### ➲ Summary

| Response Time/Page (in Seconds) With 150 Users | w/o Cache | w Cache | Performance Optimization |
|---|---|---|---|
| **Home Page** | **14.2** | **5.1** | **281%** |
| **Document Page** | **15.7** | **7.3** | **214%** |
| **News Page** | **17.9** | **8.1** | **220%** |

When we compare the response times obtained for 150 concurrent users, we can clearly see that performance is improved. On average, response times showed an improvement of 230% across all the pages of the application.

### ➲ Details

Without Zend Cache

| Response Time/Page (in Seconds) | 1 User | 10 Users | 50 Users | 100 Users | 150 Users | 200 Users |
|---|---|---|---|---|---|---|
| **Home Page** | **0.369** | **0.946** | **4.699** | **9.542** | **14.24** | **N/A** |
| **Document Page** | **0.409** | **0.969** | **5.232** | **10.43** | **15.66** | **N/A** |
| **News Page** | **0.442** | **1.24** | **5.913** | **11.83** | **17.92** | **N/A** |

With Zend Cache

| Response Time/Page (in Seconds) | 1 User | 10 Users | 50 Users | 100 Users | 150 Users | 200 Users |
|---|---|---|---|---|---|---|
| **Home Page** | **0.146** | **0.368** | **1.71** | **3.402** | **5.072** | **6.344** |
| **Document Page** | **0.197** | **0.425** | **2.462** | **5.051** | **7.311** | **10.05** |
| **News Page** | **0.192** | **0.568** | **2.531** | **5.123** | **8.152** | **10.32** |

## 4.4 Transactional Rate results

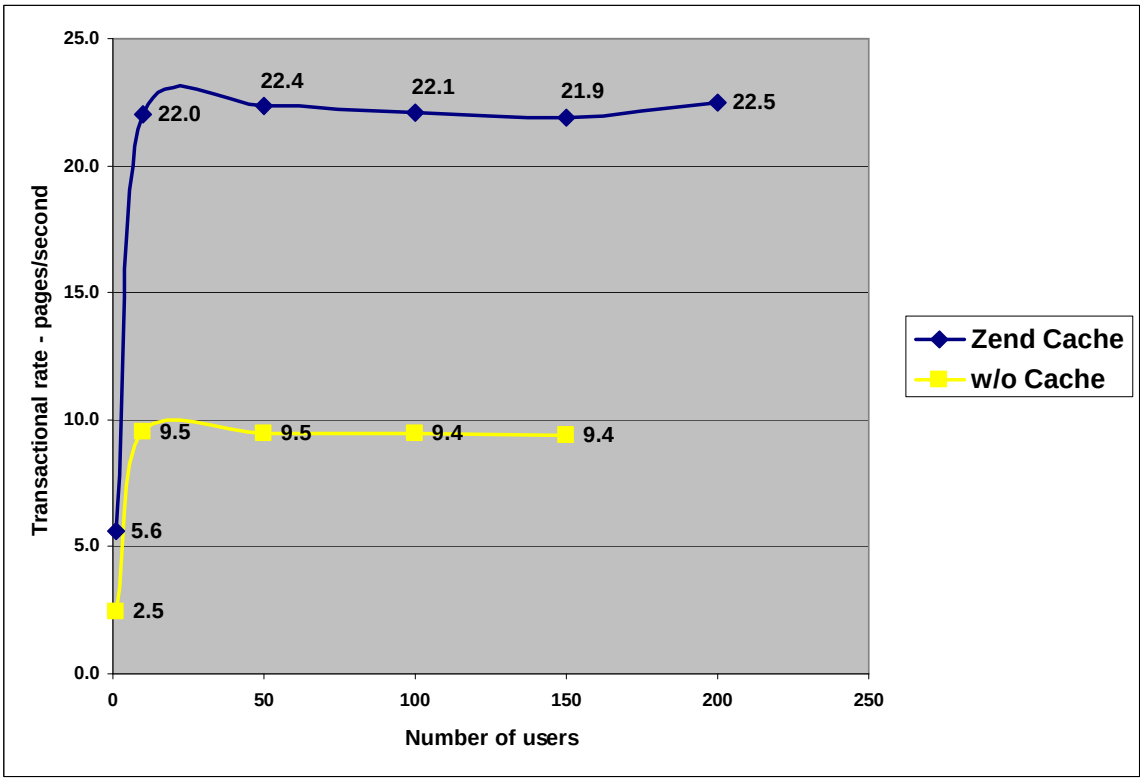


*Figure 3: Transactional Rate: Pages/Second with and without Zend Cache*

The transactional rate remains perfectly stable for both configurations. The improved performance brought about by the activation of the Cache represents approximately double the performance level of a configuration without Cache.

Response times are completely linear. Without the Cache, response times are not particularly good; with the Cache, however, the response times become acceptable.

In terms of system consumption, we observed memory saturation problems during the "before" tests. With 150 robots, the server was already consuming nearly 95% of its total memory, while with the cache, it only consumed 75% of its total memory.

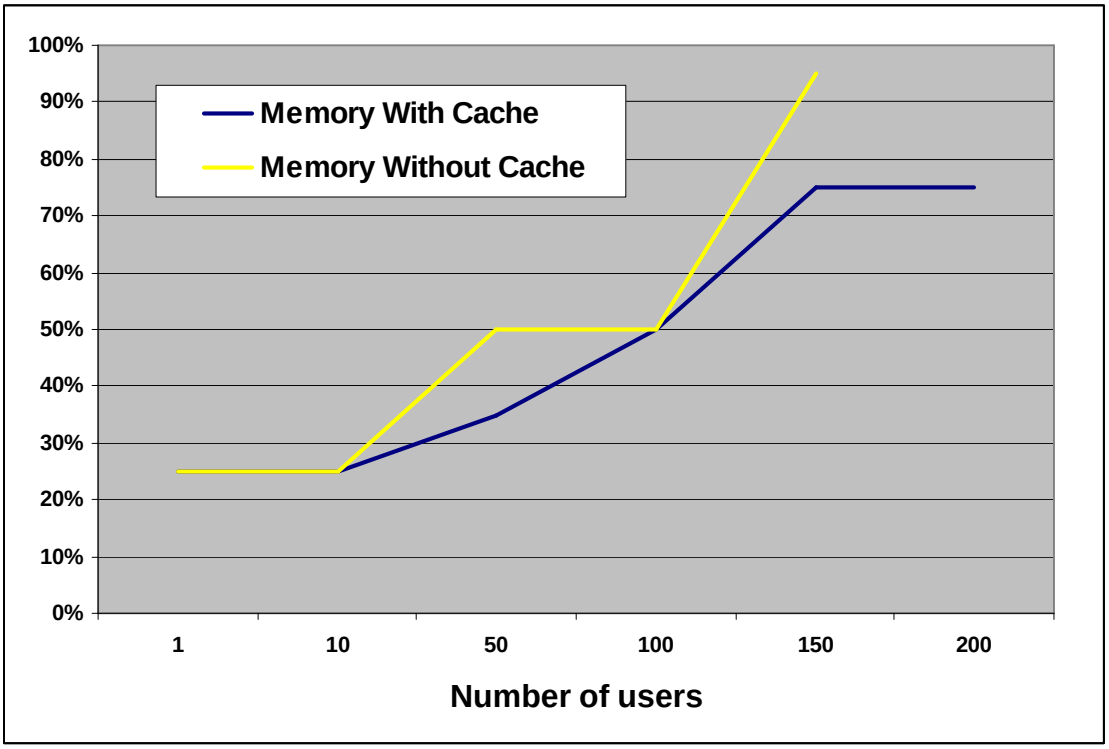


*Figure 4: Memory used on Zend Engine Server.*

# 5. Conclusion

We can therefore confirm that by using Zend Cache on a PHP platform, it is possible to preserve the server's system resources, thereby improving the performances of this type of application.

To make the most of the assets of Zend Cache, we recommend incorporating, from the design phase of a PHP application, a certain number of rules which will increase the beneficial effects of Zend Cache, for example:
Factorizing the business functions into libraries included in the various scripts of the application.

In the event of a "poorly" developed application, using the cache can nonetheless make it possible to avoid the cost of further development, if the application shows deficiencies in terms of performance.

# 6. Application and Platform

## 6.1 Application

The application was installed on the TechMetrix platform and subjected to sustained workload increases in order to assess and analyze the application server's operation in extreme conditions. TechMetrix consultants carry out all measurements.

The PHP – Zend Cache platform was tested using an existing application developed by Groupe SQLI. This application, called CapCity and based on the OSS framework Interligo (www.interligo.org), is a Knowledge Management Portal.

No specific optimization was carried out on the application before or during the tests. Basically, TechMetrix Research performed a series of measurements, classed as "Before" (without Zend Cache) and "After" (with Zend Cache).

The scenario selected for the application involved browsing through three pages:

- Home: site homepage.
- Document: document consultation page.
- News: site news page.

Insertion in the MySQL database is carried out within each script. Selection of the first 30 lines of this access statistics database is carried out within the Document and News scripts.

## 6.2 Platform

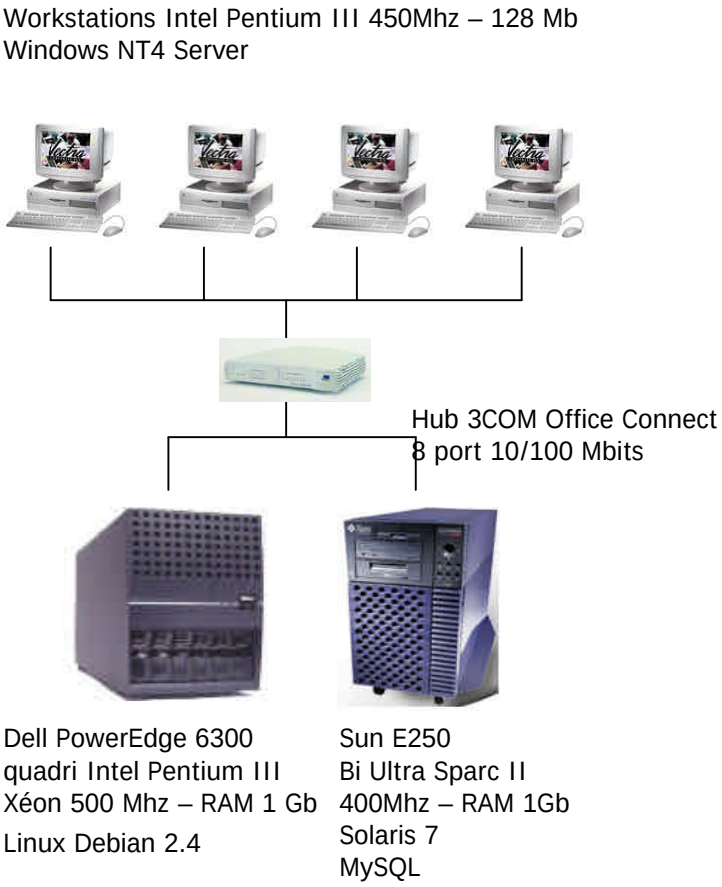


*Figure 5: Platform used in performance measurements*

A white paper written by

# TechMetrix Research

**Author**
Matthieu Belge

**Contributor**
Jean-Christophe Cimetiere

Publication date: April 2001
RC01

**USA**

TechMetrix Research
440 Totten Pond Road, 2nd fl
Waltham, MA 02451

Tel.: +1 781-890-3900
Fax: +1 781-890-9143

http://www.techmetrix.com
info@techmetrix.com

**EUROPEAN HEADQUARTERS**

TechMetrix Research/SQLI
55/57 Rue Saint Roch
75001 PARIS
FRANCE

Tel.: + 33 1 44 55 40 00
Fax: + 33 1 44 55 40 01

http://www.techmetrix.net
info@techmetrix.net

**SWITZERLAND**

TechMetrix Research/SQLI
Chemin de la Rueyre
116-118
CH-1020 RENENS

Tel.: + 41 (0) 21 637 72 30
Fax : + 41 (0) 21 637 72 31

http://www.techmetrix.net
info@techmetrix.net