# Performance Report(S2)

### Definition

Performance report is a documented assessment of performance and capacity of an application or system that is complex, time consuming and expensive to build. Performance test helps to reduce the risk of down time on multi-user interface by conducting tests that use “load” to reveal errors and limitations in the application.

The steps in a performance test are:

* Discovery
* Modelling
* Developing Scripts
* Execution of tests

### Goal

1. To clarify metrics and factors that the pages operate on
2. State of assumptions
3. Process description
4. Improvement opportunities

### Assumptions

These assumptions should be revised by the team closely related to the business and specific part of the application.

Software and Hardware

* CPU
* Network Connection
* Hard Drive
* Memory
* Version of Operating System
* Version of Software
  + Web Server
  + Database
  + Applications Server

### Process Description

All the steps in performance testing matter in making good decisions to make a project successful. These steps include, but aren’t limited to:

* Discovery
* Modelling
* Developing Scripts
* Executing Test

### Optimizer Tools

Possible tools for code optimisation/performance improvement would include:

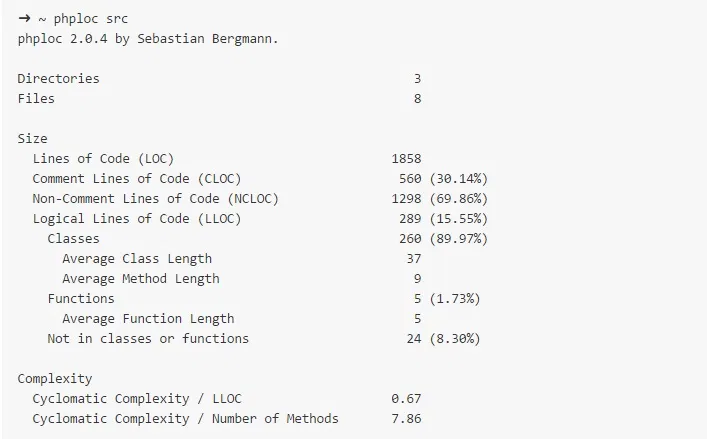
#### PINBA

A MySQL storage engine that monitors the performance of PHP code and detects bottlenecks in real time. Statistics are displayed in user-friendly, human language. Interface is read only.



### PHPLOC

A tool to measure how many lines of code in a PHP file as well as number of classes, files, etc. Ability to generate retroactive reports in CSV format.



## Data Need Analysis

Load testing tools have graphic capability. Graphs are just tools but not an actual report, however graphical data aids visualization to guide the stakeholders in consuming actionable information. Learning pattern recognition can take years to acquire where the need exists to recognize the system performance changes after certain load is surpassed, understanding of the limiting resources etc, is an ongoing and changing process.

A performance tester does the following duties:

* Form hypothesis
* Draw tentative conclusions
* Collect information to determine the information needed for the above
* Prepare key visualizations that provide insight into the performance and bottlenecks
* Support report narratives

To perform the above duties a thorough understanding of the following is very important:

* Architecture
* Hard and Soft Resources
* Garbage Collection Algorithms
* Database Performance
* Message Bus Characteristics
* Auxiliary Components of Systems – in complex systems

The full value of the performance test is unlocked when there is collective information from Developers, Operations, database analysts, help desk techs, business stakeholders and all teammates. Few effective steps to successfully achieve this are:

### Collecting

To weigh up the validity of the performance results its essential to gather-

* + Errors and type of errors
  + Pattern of errors
  + Obtaining error logs from the application

Measurements from every few seconds helps to understand the granularity of the application and help us to spot the trends and transient conditions in application development and testing.

### Aggregating

Measurements with statistics like scatter plots, graphs, data ranges, variance studies to study the data distribution aids in making the report more accurate. Using various levels of granularity provide isolated to collective views of the performance with compared with consistent granularities. This can be an improvement strategy standard.

### Visualizing

Comparison studies are done by using key graphical indicators to help us understand what is happening during the testing of the application.

List of the comparative visualizations are

* Check for validity of results by studying **Load data vs Error**
* Recognizing bottlenecks by checking **Load vs bandwidth throughput**
* Study scaling and scaling behavior by studying **Load Vs Response time**
* **Capacity of Infrastructure** i.e.system resources adequacy is determined by
  + Load vs Server CPU
  + JVM heap Memory
  + Input/output latency
  + Database lock contention

### Interpreting

Evaluating the data and drawing conclusions from hypothesis can be done by

* Quantitative observations – What can be mainly observed in the data?
* Comparing the observations – Where are the consistencies and inconsistencies?
* Developing hypothesis based on observations
* Testing the hypothesis
* Conclude from the hypothesis by validating

### Analyzing

Deciding on actions to be taken by checking if the objectives are met, then determining remediation options at business level, applications level, system requirements and network level, then retest.

Transparency in costs, benefits, and risk is essential, them must be specific and actionable at technical level or at the management level.

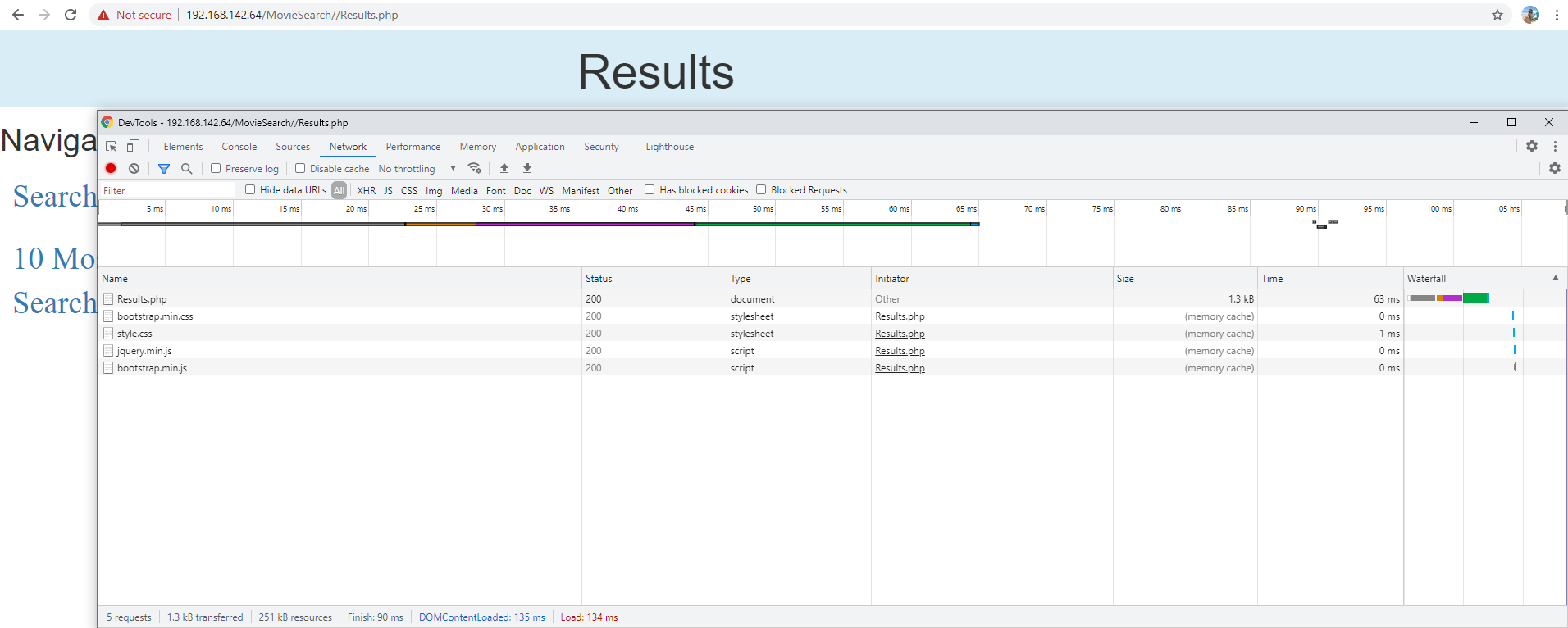
### Reporting

This is done by aggregating and presenting the risks, costs, limitations and recommendations to the stakeholders’ terms in a short elevator summary or a brief narrative. The report has the following sections:

* Executive summary
* Supporting detail
* Documents associated with the test
* Presentation

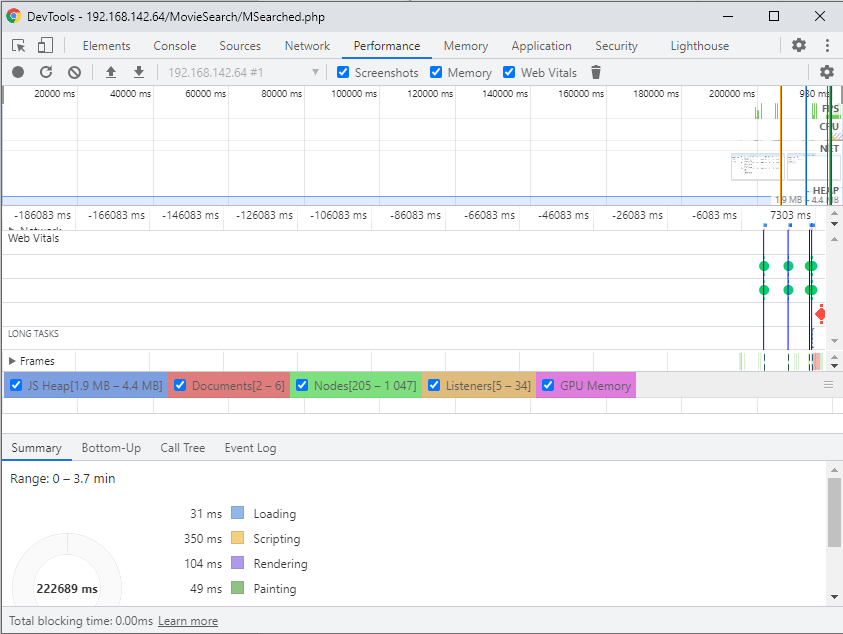
## Performance of Results

Response that we get within 65ms.



## Performance of Top 10 Movies

The completion of top 10 movies takes up to 980ms which includes the loading of a graphical chart.



## Performance of Home Page

The Home(Index) page loads less than 50ms.

