Database problems of maritime transport industry on high load platform

Sergey Sokolov^{1,}, Nikolay Glebov¹, Roman Novoselov¹, and Sergei Chernyi¹*

¹Admiral Makarov State University of Maritime and Inland Shipping, 198035, Dvinskaya st., 5/7, Saint-Petersburg, Russia

Abstract. The article says about distributed and high load systems, common problems and common instructions for solving such problems as denormalization, replication and sharding of transport and maritime aspects. The analysis of strengths and weaknesses of technology for the transport field of activity was carried out. A fragment of the code for sampling optimization is given. Conclusions about the global use of industry technology are made. When your current infrastructure starts to show the first signs that it is no longer cope with the load and if you have a 128 MB VPS - for you, it can be 10 requests per second. For corporations, it can be 10,000 requests. There is a need for scaling and optimizing the infrastructure. Maritime office needs a good monitoring system. It will help determine the very moment when it's time to scale. Stable systems for monitoring and tracking server performance trends: Munin, Zabbix, Nagios. What usually happens when the high load point is approached: slow or infinite loading of pages; random errors; broken connections from the Web server; partial loading of content (for example, there is no part of the image), and a decrease in audience activity. This direction is very modern and promising for the transport industry.

1 Introduction

At present, the development of any transport enterprise is impossible without providing its information infrastructure. The process of functioning of a transport enterprise requires not only the movement of material values but also the constant movement of information flows. Intra-production, national and international transportation of goods requires continuous information coverage and documentary support. At all stages of the transportation and transshipment of cargo, there is a constant exchange of data between the participants in the transport process. This places high demands on the accuracy and speed of information transfer, which often depends not only on the clarity and continuity of the process but also on the performance of the contract. Compliance with the requirements can only be achieved through the introduction of automated information management systems that implement orderly storage and rapid tracking of information about cargo and transport components, coordinated planning and management of the fleet and cargo flows. Common

* Corresponding author: sergiiblack@mail.ru

systems unite all agents in all countries, provide instant access to the necessary data, and significantly simplify the procedures for issuing and verifying the necessary documents. The possibility of electronic data exchange with customers and partners reduces the likelihood of errors and delays related to the human factor. The totality of the technical means used in such systems and the methods of using them are referred to the information technologies of the transport complex.

In modern world, the amount of information is increasing every minute. There is a problem named “information explosion” - the rapid increase in the amount of published information or data and the effects of this abundance. It's obvious that problem of working with information is very urgent, and we can't use easy technical solutions. We have to find some ways to solve the problem. In this article, we will define the high load information systems and consider some methods for load optimization.

2 Definition method and formulation of the problem

To discuss the problem, we must define a high load information system, but initially, distributed system. Andrew Tanenbaum in his book «Distributed Systems: Principles and Paradigms» [1] suggested the following definition:

«A distributed system is a collection of independent computers that appears to its users as a single coherent system».

- opportunity to work with different devices: different device providers, different operation systems, different hardware;
- opportunity for development and scaling;
- persistent access to resources (even if some element of system out of touch);
- hiding features of communication from users.

A set of information systems that differs in syntactical or logical aspects, such as hardware platforms, data models or semantics, is called a heterogeneous system.

For distributed system, Tanenbaum uses some types of transparency:

- access - hide differences in data representation and access to a resource;
- location - hide where a resource is located;
- replication - hide that a resource is replicated;
- concurrency - hide that a resource may be shared by several competitive users;
- failure - hide the failure and recovery of a resource.

As we can see, some of them we are using in context of high load systems. But we have to figure out what class or subclass is and define a high load information system.

High load Information System - is a distributed system, while developing one, they are using some other technologies which are not connected with distributed systems. Example: design patterns, algorithms and others. Also, high load systems are connected with big data concept. Big data - is a term for data sets that are so large or complex that traditional data processing application software is inadequate to deal with them. Big data challenges include capturing data, data storage, data analysis, search, sharing, transfer, visualization, querying, updating, and information privacy.

It is ironic, but big data is not a volume, just methods.

Examples of tasks which uses big data:

- Logging users' behaviour in the Internet.
- GPS signals from drivers to office of transport company.
- Data which captures from Large Hadron Collider.
- Digitized books from the National Library of Russia.
- Information about transactions of bank clients.

- Information about purchases in big commercial network etc.

This data is contained in databases or rarely in simple text files (ex. logs of users' behaviour). This article will consider a web application for "highload information system".

In this article, we will solve a problem of loading an information system, firstly, into a database.

3 Ways to solve transport problem

In previous works [2-7], we proved that speed of work directly depends on amount of data. That's why the first thing which will slow down the system is database. First, we have to move database to separate server to increase its performance and reduce negative influence on other system components (resources, web-server etc.). Then, we have to move web-server to another node. We will release more resources for calculating. Later, we can split the calculations to several servers, set cache, queues, and balancers.

Information technologies for the transport industry, which are the result of a combination of technical capabilities of computer technology, telecommunications, computer science, are aimed at collecting, accumulating, processing, analyzing, and delivering information to consumers. The latter are scientists, engineers, managers, production managers, doctors, diplomats, economists and other specialists of the most diverse profiles. Information technologies in transport are designed to provide automation of routine operations of management personnel, preparation of analytical information for decision-making and to supply to its consumers any kinds of data regardless of distances and volumes.

The above-mentioned properties of information technologies are especially needed in transport with its rapidly changing factors: the disposition of vehicles, the movement of goods, the changing demand and supply of transport, market conditions and many other indicators. But usually the heaviest element of a system is database, so we should give much more attention to it. Scaling of database is one of the hardest tasks of system growth. There are many different practices - denormalization, replication, sharding etc. Let's consider some of them.

3.1 Denormalization data

There is a so-called normal form of data storage, which involves avoiding duplication of data. There are two key rules:

- Atomicity means that all entities are stored in an indivisible form. For example, if we store the address, it is likely to be divided into the name of the city, country and street. All of them should be presented in separate tables. The name of the city will be atomic, as it will not be further divided.
- Uniqueness requires that each entity is defined only once. For example, the city name with ID 1 must be present only in the cities table.

Denormalization is a gradual process of getting rid of normalization rules where it's necessary. Typically, these are cases when requests to logically related data are repeated frequently [3].

A simple example is storing the city name together with the user's data. If you denormalize this data, you can get the city name with one SQL query, not two.

3.2 Data replication

Replication is one of the techniques for scaling databases. This technique consists in the fact that data from one database server is constantly copied (replicated) to one or several others (called replicas). You can use more than one server to process all requests. Thus, it becomes possible to distribute the load from one server to several. There are two main approaches when working with data replication:

- Master-Slave replication;
- Master-Master replication. [4, 7-9]

3.3 Sharding

Sharding is the principle of scaling a database when data is shared across different servers. We have at our disposal two approaches:

- Vertical sharding is the simple allocation of tables to servers. For example, you can put the users table on one server and the orders table on the other. In this case, the groups of tables for which JOIN is executed must be on the same server.
- Horizontal sharding - the separation of very large tables which cease to be on the same server on different servers. This greatly complicates the logic of the application, but at the moment, there are no better scaling mechanisms [5, 7-9].

The best option, according to the article's author opinion, is the combination of all these methods. Denormalization of data will help to get rid of complex JOIN queries - connecting several tables is a very resource-intensive operation. Replicas, as mentioned above, will help distribute the load from one server to several. As the most effective method, sharding will allow receiving really optimized and fast highly loaded system.

4 Result and example

For example, imagine some maritime office network. The main idea of development of the system is achieving the company's main goals and improving the efficiency of the freight forwarding company.

The effectiveness and efficiency of the optimized system (consisting of separate subsystems (mechanisms)) is proposed to be considered within the single concept of «efficiency» as the degree of achievement of the goals of the organization to which the system is oriented. This takes into account both the results that characterize the achievement of the goals (qualitative, quantitative and performance indicators) and the resources used (the development of corporate culture). The next important aspect of the system to be considered is the quantitative and qualitative indicators. Quantitative indicators include: total volume of customers base; the number of customers who re-applied to a company; number of received applications; number of completed applications; number of requests for replacement of applications due to its nonconformity; revenues from transportation; profit; total transportation costs; number of complaints on the implementation of the service; number of entities involved in the transaction [10-12]. Qualitative indicators include: sustainability of relations with suppliers; ability to adjust flexibly to rapidly changing market conditions; range of services; execution time of the application; choice of the optimal solution for the proposed application; tracking the progress of an application; the company's reputation in the market of freight forwarding services; the profitability of transport; registration of necessary documents of title and transport documents; using its own container fleet and/or vehicles where it's possible.

Performance indicators are derived from the nature of the company. Depending on the specific conditions and requirements, the list of these indicators can be expanded and

supplemented. This conditional division into groups, in our opinion, allows us to determine the indicators of the optimized system more adequately [12].

We filled tables with test data - 10000 users and 5 cities. Examples of table are presented below. Time of query execution was determined via build-in MySQL profiler.

Table 1. Sample of table 'users'

ID	firstname	lastname	city_id
1	John	Smith	2
2	Anthony	Park	3
3	Richard	Edwards	1
4	Reynard	McKinney	2
5	Charles	Turner	1

Table 2. Sample of table 'cities'

ID	name
1	London
2	New-York
3	Paris

Imagine we need user with ID 6, including the city. According to current scheme, we have to use JOIN. The query will look like this:

```
SELECT *, 'name' AS 'city_name' FROM 'users' LEFT JOIN 'cities' ON 'users'. 'city_id'='cities'. 'id' WHERE 'users'. 'id'=6
```

This query takes 0.000502 seconds. Now, we'll try to denormalize data: we are adding column 'city_name' to 'users' table, where we'll fill it with the city name.

Adding column:

```
ALTER TABLE 'users' ADD COLUMN 'city_name' VARCHAR(100);
```

Now, to get the user including the city, we have to run this simple query:

```
SELECT * FROM 'users' WHERE 'id'=6;
```

This query takes 0.000398 seconds. Accordingly, the performance gain is more than 20%. And in the case of a large amount of data [3, 9-12], the result will be much better.

5 Conclusion

The study shows that the most simple and least powerful way to optimize the load is the denormalization of data.

However, such a solution is finite - it is impossible to denormalize the data indefinitely. Replication and sharding are more popular ways of optimization, besides, they can be used together. In the case of developing the architecture of a truly loaded information system, it is necessary to use replicas and shards. According to the author of the article, the ideal option is to combine all the considered methods. When using these methods in combination, you can create a system that will meet the criteria for a distributed system.

References

1. A. Tanenbaum, *Distributed Systems: Principles and Paradigms*, (Piter, 2003)
2. URL: <https://ruhighload.com/post/>

3. URL: <https://ruhighload.com/post/replication>
4. URL: <https://ruhighload.com/post/MySQL>
5. V. Budnik, S. Chernyi, *Procedia Engineering*, **150**, 2150-2156 (2016)
6. S. Sokolov, A. Zhilenkov, S. Chernyi and A. Nyrkov, *Procedia Computer Science* **125**, 420-426 (2018)
7. D. Malioutov, A. Corum, *IEEE Journal of Selected Topics in Signal Processing* **10(6)** 1006-1014 (2016)
8. A. Nyrkov, A. Zhilenkov, S. Sokolov and S. Chernyi, *Automation and Remote Control*, **79(1)** 195-202 (2018)
9. A. Zhilenkov, S. Chernyi, *Procedia Engineering* **100** 1247-1252 (2015)
10. G.D. Alekseev, V.A. Karpovich, *Power installations of fishing vessels*, (Shipbuilding, 1972)
11. S. Sokolov, A. Zhilenkov, A. Nyrkov, and S. Chernyi, *Advances in Intelligent Systems and Computing*, 421-427 (2017)
12. S. Chernyi, A. Zhilenkov, S. Sokolov and A. Nyrkov (2017) "Algorithmic approach of destabilizing factors of improving the technical systems efficiency" *Vibroengineering PROCEDIA* 13: 261-265.