

Breaking nine million users achieve scalable architecture

<http://www.infoq.com/cn/presentations/9-months-achieve-large-number-users-scalable-architecture>

<http://segmentfault.com/a/1190000002924334>

在不同阶段选择最适合自己的方案

First, start with the architecture, the beauty shot look through several stages:

1. Minimalist design (quick release on-line)
2. Keep it simple line design (product rapid iteration)
3. Scalability and high availability guarantee (number of users to a certain order of magnitude)
4. High scalability and high availability guarantee

Then we look at the United States beat the problems encountered along the way:

1. MySQL slow queries
2. MySQL write bottleneck
3. redis timeout
4. memcached hit rate is extremely low
5. Services interdependence
6. Monitoring alarm instability
7. CDN service a variety of failures
8. Add Field high cost
9. After middleweight up, MySQL continues to slow

Then we press service dimension to each service apart, look at each service in the United States beat the iterative process architecture.

A, MySQL

MySQL is the most important service in the United States to shoot architecture has undergone several iterations. In the first edition is a direct instance, in order to keep the code simple, can do business logic in the database are made into the database, such as Feed, direct access with the MySQL join.

But then there have been some cases of slow queries, this time to do a master-slave, do separate read and write, **more than from the library** used to make queries. Then later the situation appears too slow to write, this time has been done to optimize the architecture, but the hardware upgrade, because now is the stage of rapid development of business, you need very simplified design, at this stage to focus more on business development on (also estimated that wood has to hire the right people :)).

After a period of time and the situation began to write slowly, and this time did not start points table. But wait until the focus on scalability and availability, they encountered a new problem, a big problem or a write slow, the other is with the increasing amount of data, add special field of high cost. For these two issues does the following two options:

1. **Asynchronous writes , so that the front end can always write back into the queue inside the complex things to do asynchronous**
2. **Indexing and data separation , the need to split out a separate index fields in a table, other data storage kv, value pb binary data is all the attributes and values, home field to solve difficult problems**

This time for architecture optimization of MySQL before it was a paragraph :)

Second, the cache

Cache mainly uses memcache and redis (redis should be mainly used for counting and queue service). In the relatively small amount of time is the use of simple and crude, but soon encountered redis overtime problem, this time to **redis expansion, using a multi-slave architecture**. Then when the user requests the impact **rdb dump the problem, the solution first, before going to dump in the morning traffic is low, the second is to use the machines do not foreign service dump**.

Then memcached experiencing very low hit rate problem, a big problem is capacity bottlenecks, this nothing to say, expansion (small army mentioned, we should always do the expansion ready). Another is the **slab calcification** problem (also known as slab calcification problem, this is a **memcached memory allocation mechanism** caused, is simply **memcached memory will be divided into N slab, when adding a new memory object according to the size of the object to choose a different slab, without the right will create a slab, and that if this is not enough memory allocated time remaining a slab? this time there have been problems of calcification**), the United States was shot solution is the **core business of isolation deployment**, to avoid this problem.

To guarantee the availability of the stage, the more cache availability is very important, the cache may be linked to the entire system hung up, difficult to end, it is necessary to ensure the availability of the cache. This is the time to do a master-slave optimization, master also bear read queries to ensure cache heat, slave penetrate into master, master penetrate into the slave, to prevent a single point of failure.

Third, operation and maintenance

In the early simply monitor alarms, and sometimes problems with the availability may not receive a warning or you do not see is what something is wrong, then gradually improve monitoring alarms and alarm monitoring is relatively high server configuration, alarm monitoring to ensure . If more then monitor the log dimensions and more convenient positioning. Dependent on third-party services and resources to do the switch, the problem can be ensured by switching service **core paths available** .

Fourth, third-party services

CDN services are mainly mentioned. One big problem is that DNS is attacked, abducted, and in addition to the operator to maintain communication, but also to do with a policy of multi-service providers, **such as the same data in multiple cloud services there to do redundancy , client access If a problem occurs when you switch to another access address and the client do a server availability detection** . It is also a very valuable experience.

Fifth, technology stack

The United States will find the whole look down pat architecture do very steady, small army also mentioned, to overcome early in the project when the high-speed development stage architecture desire to do the perfect architecture to overcome the desire for new technology, let the system up and running .

But throughout the iterative process, the United States has been shot in the introduction of new technologies, such as the use of MongoDB in the first part of the business when the team is not familiar with the focus on scalability and availability stage, the introduction of java do business logic underlying the introduction of c do service.

<http://segmentfault.com/a/1190000000383231>

