Comparativas de tecnologias de disponibilidad

Jesus Delgado Castillo, Luis Caxi Calani, Gerardo Concha llaca

September 24, 2022

Resumen

En la actualidad, debido a la gran demanda de servicios de internet y a la transferencia de la información de todo tipo, es incuestionable que los sistemas informáticos deben funcionar de forma ininterrumpida y sin errores los 365 días del año. El intervalo de tiempo entre caídas del sistema y la reparación de la falla que provocó esa caída son variables que determinan la disponibilidad de un sistema, para ello es necesario definir mas claramente un métrico indispensable para determinar el nivel de disponibilidad de un sistema.

Para cualquier sistema de base de datos es importante lograr un nivel de alta disponibilidad, es decir estar disponible el 99.98esto se toma ventaja de mecanismos y características de los diversos manejadores de bases de datos, tales como: base de datos en espera, replicación y base de datos en paralelo.

Abstract

Currently, due to the great demand for internet services and the transfer of information of all kinds, it is unquestionable that computer systems must work uninterruptedly and without errors 365 days a year. The time interval between system crashes and the repair of the failure that caused that fall are variables that determine the availability of a system, for this It is necessary to define more clearly an essential metric to determine the level of availability of a system.

For any database system it is important to achieve a high level of availability, that is, to be available 99.98This takes advantage of mechanisms and features of the various data handlers. databases, such as: standby database, replication, and database on parallel.

1 INTRODUCTION

Databases have evolved from tens of megabytes to terabytes and still up to petabytes. Adding complexity to the management of these huge databases data, the restriction of long periods of operation that include requirements of 24 hours a day, 7 days a week. It is quite difficult to provide an administration appropriate for this type of databases, maintenance tasks, tuning and backup they become impossible; If the database cannot be taken out of service for a time, even some of these tasks can take several days. A possible solution for this conundrum, is the adoption of a high-availability technology for data systems. database.

A database (BD) can be defined as a collection of interrelated data that contain important information to fulfill the objective for which it was built. A database management system (DBMS) incorporates a set of programs to create and maintain a database. A DBMS provides protection of stored information against hardware or software malfunction, protection against unauthorized access to the system, concurrent access to stored data, among the most important functions. There are different types of databases. The relational data model is based on a set of files called tables, made up of rows and columns. Each row represents a record in the file and is called a tuple, and each column represents an attribute of the record with a domain of specific values. This type of database representation is currently the most widely used, given its potential and simplicity compared to other data models. Another data model is object-oriented (OO). This model is an extension of the OO programming paradigm.

2 AVAILABILITY

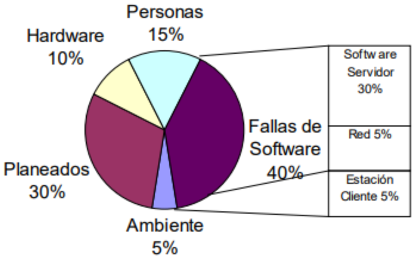
2.1 Availability Measurement

When discussing availability requirements with a user or head of projects, invariably request 100 percent availability: “Our project is so important that we can’t have any downtime on the system”. But this requirement changes when the project manager discovers what is the cost of that 100 percent availability. Then it becomes a matter of money, and as part of a negotiation process

A trading point is the time it takes for 100 percent of operation. If this is only necessary for a few hours a day, then the goal it is completely achievable. For example, it may require 100 percent time of operation during production hours, but not during the rest of the day. For another On the other hand, if 100 percent operation is required 7 days a week, 24 hours a day, day 365 days a year, the costs become prohibitive. These costs only the most profitable applications and large companies or corporations they can consider.

2.2 Causes of downtime

Planned fall times are scheduled events, usually in the afternoon or night, when system administrators add hardware to their systems, update operating systems or other critical software such as the database manager data, or perform an administrative maintenance task on the system. Some Sometimes these planned downtimes consist solely of performing a reboot preventive, to clean logs, temporary directories, and memory.



Fuente: IEEE Computer Abril 2004

Various causes of drop times are examined in the chart. One of the Larger regions of the graph are for planned fall times. This is also one of the segments that can be reduced more easily.

The human factor is another major cause of downtime. The People cause downtime for two closely related reasons. What

The first reason is that they sometimes make careless or clumsy mistakes.

The second reason is that people cause down times because they don’t they always fully understand the way the system operates. The best way to combat downtime caused by people is through a combination of education and simple system design. Sending staff to training that keeps them updated on technologies, and having a solid documentation up to date, you can reduce the number of system crashes from this cause.

Possibly the region of the graph that causes the most surprise is the hardware. The Hardware causes only 10 percent of system crashes. This means than the best RAID disk array in the world and the best redundant networks, prevent only this 10 percent of downtimes.

In fact, in addition to disk and network failures, hardware problems are also include central processing unit and memory failures, font loss power, and internal cooling systems.

The most common cause of system downtime is probably system failure. of software. In total, the software is responsible for 40 percent of the times of crash of a system. Software bugs are probably the hardest bugs to fix. correct. As hardware becomes more reliable, and methods exist to reduce planned fall times, their percentages decrease, while the percentage of problems attributed to software causes increase. This is because the software it becomes more complex, and its problems become more frequent. By Of course, the development of new purification techniques has made progress, so software problems are less prevalent.

2.3 Availability Technologies

Sharding

Sharding is simply the process of dividing large volumes of data into smaller pieces, which are usually stored in different physical or virtual partitions. Any database can be partitioned like this, but it's a complicated process, requiring a whole refactoring and reconfiguring according to the new distribution.

Splitting up tabular models so that we can deploy part of the database in one region of the cloud, for example, and part in another region, requires some really complex infrastructure and system design. Particularly because of how the relational model is designed, to decouple data that is accessed frequently, but must remain relative to another (or other) groups of data to form an entity.

Four common sharding strategies::

- Horizontal or Range Based Sharding
- Vertical Sharding
- Key or hash based sharding
- Directory based sharding

Partitioning

Partitioning is the process where very large tables are divided into multiple smaller parts. By breaking up a large table into smaller individual tables, queries that access only a fraction of the data can run faster because there is less data to scan. The main purpose of partitioning is to aid in the maintenance of large tables and to reduce the overall response time for reading and loading data for particular SQL operations.

types of partitioning:

- Vertical partitioning
- Horizontal partitioning
- Functional partitioning

Replication

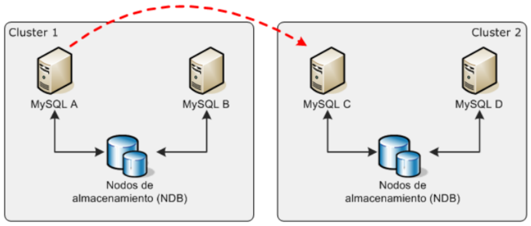
Data replication is when the same data is intentionally stored on more than one site or server. There are several reasons why companies replicate data. It allows data to be seamlessly available in the event of server downtime or heavy traffic to the server. Data becomes accessible to users on a consistent basis without interfering with or slowing down other users' access. For cloud applications, data replication allows you to access a copy of the data in a local database with much higher performance than accessing the data through the cloud application's API, which is especially useful for analytics and data science. Data replication can also allow you to avoid API transaction limits and throttling that some cloud applications have.

types of replication:

- Instant replication
- Transactional replication
- Merge replication

Clustering

clustering refers to a technique that allows multiple systems to be combined to work in parallel and behave as a unified computing resource to: serve a group of tasks, provide fault tolerance, and have continuous availability. For example, in the case of Internet users, clustering provides database, email, file, or other system services without interruption. If a fault were to occur within a network of servers in a cluster, it would be corrected immediately without users noticing..



3 Comparison

3.1 Sharding vs Partitioning

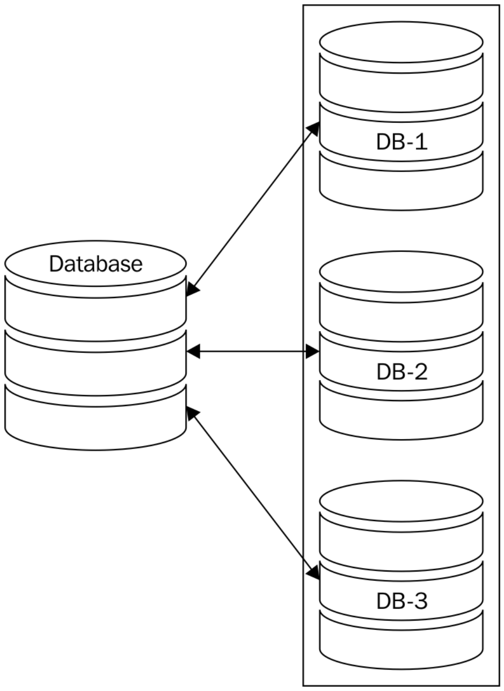
Partitioning is a generic term that just means dividing your logical entities into different physical entities for performance, availability, or some other purpose. "Horizontal partitioning", or sharding, is replicating the schema, and then dividing the data based on a shard key.

On a final note, you can combine both partitioning and sharding techniques on your database. In fact, sometimes using both strategies is required for data-intensive applications.

3.2 Clustering vs Replication

Replication - Copying an entire table or database onto multiple servers. Used for improving speed of access to reference records such as master data.

Clustering - Using multiple application servers to access the same database. Used for computation intensive, parallelized, analytical applications that work on non volatile data.



3.3 Replication vs Sharding

Sharding exists to increase the total storage capacity of a system by splitting a large set of data across multiple data nodes. Sharding handles horizontal scaling across servers using a shard key. This means that rather than copying data holistically, sharding copies of pieces of the data across different servers. Each server that stores a piece of data is called a shard and it can be a replica set to take the advantage of replication. These shards work together to utilize all of the data.

Think of it like a pizza. With replication, you are making a copy of a complete pizza pie on every server. With sharding, you're sending pizza slices to several different replica sets. Combined together, you have access to the entire pizza pie.

Replication and sharding can work together to form something called a sharded cluster, where each shard is replicated in turn to preserve the same high availability.

4 Conclusions

Fault protection has a monetary cost, as do power outages. system, so it is important to find the balance between investing in equipment and redundant systems, and the financial loss caused by downtime of the system, so measuring the availability of the system is mandatory and key, being the ratio of the average time between failures and the sum of this time and the average failure repair time known as availability.

A highly available database system is nothing more than the implementation of features of a DBMS, to achieve a level of high availability.

5 Recommendations

To implement high availability in a system, it is necessary to understand the level of availability that users require in applications and its impact financial, so a service level agreement must be defined where it must be conduct a review of critical applications and hours of operation, as well as the cost of an eventual system failure, both from the business perspective as at the level of technology.

Review of a company's current technological resources is mandatory for a technology director, faced with the challenge of raising the availability levels of the systems.

Review of a company's current technological resources is mandatory for a technology director, faced with the challenge of raising the availability levels of the systems.

References

[1] [Kleppmann, Martin.] Designing Data Intensive Applications

[2] [Rob, Peter (2004)] Sistemas de bases de datos: diseño, implementación y administración

[3] [Bell, David (1992)] Distributed database systems.

[4] [Iglesias, Eva .L] Bases de datos distribuidas

[5] [González Martín, Osca] Arquitecturas de sistemas de bases de datos.

[6] [Chinchilla Arley, Ricardo] Fragmentación de datos en bases de datos distribuidas.