

# White paper: Firebird performance recommendations

PDF Download

Holger Klemt, June 2013

This White Paper was prepared following an on-site performance workshop at a client's, the aim being to analyze an application running on a Firebird 2.5 database. The company was suffering tremendous performance problems, although the database was only 10 GB large with 50 clients working on it at any one time.

The following details the proposals made by IBExpert, specific to this client's issues. We have decided to publish this paper, as we are sure that many other companies could benefit by some or all of the following proposals.

Should you be interested in having your Firebird-based application analyzed, please contact [sales@ibexpert.biz](mailto:sales@ibexpert.biz). The analysis can be performed on site or via remote connection.

## Software-related issues

There are several non-optimized SQL statements executed by the software that create a much heavier workload on the server than necessary. The IBExpert [Monitoring](#) feature enabled us to see most statements, and the [Trace and Audit](#) function in IBExpert revealed certain additional issues.

There is no general rule that allows, for example, only 1,000 non-indexed reads or 1,000 page reads per statement, but when a statement is executed with such poor syntax optimization, based on the deployed O/R mapping architecture chosen by the software manufacturer, the resulting workload for the database server of a single user will have the same impact as dozens of users working on the database server with optimized syntax.

From my perspective, there should be a quality control level inside the application development process that evaluates the impact of all statements on the database, especially on a database that represents large amounts of real world data, not just a small developer database.

Any SQL that requires more than 100ms of execution time should be reviewed; every query that has a [SORT](#) or [NATURAL](#) statement in the plan should be reviewed. Results of the review could lead to changes in the database model, for example defining new tables, altering existing tables, adding indices and occasionally perhaps also removing existing indices, but also removing and replacing parts of SQL statements.

The Firebird Database Server platform is capable of handling the workload of thousands of users. I reported from enterprise customer projects with this number of users. But when each user creates a workload of one hundred users, this cannot work (and will also not work on any other database platform).

When the O/R mapping architecture used by the software manufacturer automatically creates non

optimized statements, the resulting software product is of no use for enterprise applications, whatever the advantages for the developers might be.

The focus should be on the comfort and usability for the end user, combined with the technical features of the software. If all autogenerated SQL statements suffer from poor syntax, they should be replaced by optimized versions to ensure end user comfort and usability.

## Conclusion

The software manufacturer should optimize every single statement, as we did with some statements during the workshop. If external help is required, we are able to provide our services based on hourly rates with on-site workshops.

Using non-optimized SQL statements for whatever reasons cannot be accepted. "The O/R Mapper creates it automatically" is not an excuse for wasting employees working time and productivity! As we have seen, small changes often improve the speed dramatically and decrease the server workload significantly.

## Hardware related issues

Using a Firebird server with the database on an external SAN/NAS Drive is a bad idea, since the speed of a Firebird database server depends on the speed of access to individual 4k blocks on the device. As I mentioned, external SAN/NAS Drives are great for transferring hundreds of MB of data from one file location to another. However the Firebird server transfers hundreds of thousands of small pages between memory and physical hard disk found on different file offsets.

On a classic mechanical hard drive you have to consider the access time of the hard drive to be the limiting factor, because a commit writes data in very different file offsets. A RAID does not improve this; the head position must change in the same way. Caching RAIDS improve it but, with a high workload, they can sometimes increase the problems.

Classic hard drives have at least 5ms average seek time, which results in a maximum of 200 IOPS, the new standard for measuring faster devices.

An SSD (especially enterprise SSD) allows much more IOPS (up to 500,000 on PCIe-based hardware). An SSD built in the dedicated database server, which is not virtualized, is the optimal solution for maximum performance.

## Example configuration

- **CPU:** Intel Xeon E3 or E5 (non-threaded versions are OK)
- **RAM:** 8GB or more (ECC is OK, but not required)
- **DRIVES:** All SSDs should be enterprise level certified by the manufacturer: 1 SSD for the OS, 1 SSD for the database server, 1 Hot Plug SSD for an active shadow, optional 1 hot plug SSD for a part-time shadow as a backup replacement. When the database server and shadow SSDs reach an age of 2 years, they should be replaced to avoid unplanned failure (in server systems, we recommend the same for classic hard drives). The size of the SSDs should be 400% of your

current database size or higher.

- **OS:** Win7 64 Pro is OK, no file shares or any other MS services except RDP should be active, no antivir, no online backup or imaging software etc. Shutdown all unused services, especially Windows updates. File system access should be managed by an FTP server, for example FileZilla, to transfer backup files to the external company backup solution. Linux is also OK, but it does have a much wider variety of potential problems. We prefer stripped Win7x64 Systems. Win2008R2x64 is also OK.
- **RAMDISK:** For LCK and TMP Directory, a RAM Disk is very useful. For RAM Disk use, the physical memory should be increased to 16GB or 32GB. More is not required. A free RAM Disk software can be recommended on request.
- **REPLICATION:** Optional: A replication system implemented by IBExpert can ensure transaction-safe clustering. A backup or shadow solution requires admin operations to restart the server on a second server in case of a severe hardware failure.

A replicated cluster enables work on the backup server with no loss of data. This requires some changes in the database and perhaps also in the system architecture. IBExpert can support the software manufacturer in the implementation process. A replication solution based on our technology is a separate project.

## Important

A Firebird database server should be a dedicated server and should only be responsible for the Firebird database service in order to ensure maximum performance. Virtual servers lose between 20% and 80 % of their speed under a high load on a VM. Any advantage of a VM-based Firebird Server will be paid for dearly by all employees waiting additional time to carry out and complete their jobs.

From:  
<http://ibexpert.com/docu/> - IBExpert

Permanent link:  
<http://ibexpert.com/docu/doku.php?id=01-documentation:01-06-white-papers:firebird-performance-recommendations>

Last update: **2023/06/20 16:37**

