



Firebird und IBExpert White Paper

The Cloud and Firebird: The physical fundamentals of failure or Why it works really well when you do it right!

Holger Klemt, June 2022

A few days ago, a customer complained about the performance of his software with the Firebird server in an Amazon EC2 cloud instance. As already described in various white papers of ours, we have pointed out that we do not recommend such installations. Only for those who have a lot of time and do not attach great importance to the software being truly quick to use will be quite satisfied.

But who can afford to accept considerable losses in terms of speed?

If the advertised advantages of the fail-safe nature of an Amazon EC2 cloud instance - having everything secured by simple mouse clicks here and there - are more important to you than performance, you should not complain about the lack of performance.

A simple mouse click can certainly remedy this situation: More power = simply upgrade everything ...

Well then, if you think this is all so important and great, why do you complain about the speed of this solution?

It's always the same and anyone working in engineering ought to be familiar with the problem.

Since not everyone reading this understands all the typical IT management shortcuts, here is an easy-to-understand variation on why IT projects fail.

You purchase new office printers for your company.

Almost all printers have an incredibly economical energy-saving mode, as the manufacturer describes it. Often, however, this energy-saving mode ensures that employees who want to print something out do not just have to wait for the printout time until they hold the document on paper in their hands. Depending on the laser printer and the technology, a few minutes may pass before the printer reaches its working temperature.

So what do you do? You switch off the energy-saving mode!

The printout is promptly available again, but electricity costs increase, but are the lesser evil in view of the waiting times that are no longer necessary and the associated loss of working time and productivity.

So office printers have caught on and if they respond quickly, employees and supervisors waiting for reports are happy.

The deactivated energy-saving mode, multiplied by the number of workplace printers, often causes the concept of many workplace printers to be questioned in view of the current sharp rise in energy prices and steadily increasing prices for materials.



You now receive an offer from a printer manufacturer. The xxx model as a departmental printer would be much more expensive to purchase, but offers significant savings in consumption and energy costs. No sooner said than bought, the workplace printers are abolished and instead of printing costs of, say, 8ct per page and a maximum of 10 pages per minute on the workplace printer, the departmental printer creates direct consumption costs of 2ct per page and 40 pages per minute.

This sounds great and convinces management of the need to take cost-cutting measures. However, as is so often the case, it is only half the truth, because now when a single employee prints out his 500-page report, everyone else has to wait to continue with their work. Coffee-drinking queues at the departmental printer may promote communication among employees, but rarely productivity.

What to do? Printer manufacturers are now recommending switching to the hyper press, which can print up to 100 pages per minute. And to avoid bottlenecks, therefore, simply purchase 2 per department right away, or preferably even 3!

However, employees who do not want to print 500 pages at once point out that in order to print a single page, they first have to get from their workstation to the departmental printer and then find out, for example, that the document contains a spelling mistake that was not recognised on the screen. So into the bin, back to their workstation, reprint, back to the printer, etc.

Many high-end printers require a relatively long time after the printer makes the first noise, indicating that it wants to fulfil the print job, until the finished printout is in the output tray. Not to mention use of a duplex unit. If 10 employees want to print out one sheet each, then a high level of unproductive time spent by the employees can be measured directly with a stopwatch and the advantages can be clearly evaluated against the disadvantages.

What does all this have to do with Firebird running on cloud systems?

While the above example shows directly measurable advantages or disadvantages compared to the tried and tested solution with workplace printers, a comparison with cloud systems and virtual servers is not so easy.

Many server manufacturers and virtualisation software producers promise far more advantages than is actually the case in practice. Certainly, server operation benefits from the replacement of e.g. formerly 10 dedicated servers by 2 more power-saving high-end servers.

Many software systems can even measure how quickly the software used displays the current order summary on screen, for example.

But if the order summary takes 15 seconds to appear on the screen and the staff are of the opinion that it has become much slower since the use of the new cloud system hardware or virtualisation and previously only took 5 seconds, then the decision-makers still do not question the project. Most of the time, the software manufacturer is blamed because the newly acquired machine is the best of the best, and it is the software that cannot really implement it.

Often, this very software with a Firebird server as backend is the main tool used by staff, so that the working speed thus influences the productivity of the entire company.



Often, IT department heads are the ones who have just spent at least twice the annual salary of the administrative staff on the virtualisation project for the hardware and software installation. Because heaven and earth was promised, that everything would be faster, better and safer, different variants are questioned, but rarely the new expensive hardware. Complaining to the software manufacturer shifts the responsibility.

It is then that IBExpert is often the contact for Firebird-based software, often on behalf of software manufacturers.

Our benchmark provides a reproducible speed value for the operation of the Firebird database server for each Firebird database and we are ready to explain to IT department managers where the causes of the performance problems are to be found.

However, because it is often physical basics that are responsible for the slowness when operating a Firebird server on unsuitable hardware/software, it is usually evident after performing our IBExpert benchmark that it is simply not sufficient to just adjust a few configuration parameters here and there.

As already described in other white papers, we recommend the use of dedicated Firebird servers, which are proven to be suitable for the operation of Firebird databases and are really very fast, such as our IFS servers.

By the way, we use virtual servers intensively, especially for web servers and complex application servers, but not a single important Firebird server is run virtually.

The enormous costs that the IT department heads have now invested in the local hardware solution and virtualisation, often ensure that the project, due to the very poor overall performance, is now transferred and continued elsewhere.

Failure can also be an opportunity ...

Imagine further that a company with 800 employees in a call centre, should be able to work both physically in the company but also in their home office. (Incidentally, we serve customers of this size with Firebird-based software with our IFS server hardware in their data centres). If, because of the hardware used, each call centre conversation requires 15 seconds for the order screen with the caller's data to load and not, for example, 5 seconds as was previously the case, then significantly more employees are needed for the same number of calls, just because IT managers are of the opinion that this is the best and most secure solution available.

Unfortunately, it is neither the best and certainly not the safest solution. We also have customer reports on this, where a cloud solution unfortunately could not deliver a single one of the promised fail-safe features in the real worst case scenario. The snapshot was unusable and caused a complete breakdown for several days with considerable data loss.

Next step: the Cloud

Local virtualisation has failed, so management is now convinced of the next step into the cloud - after all, everything is much better, much safer, much faster and, above all, much cheaper in the cloud! If you click together a virtual instance, then it is easy to start it, copied as often as you like, and thus have all the performance on demand that you need.



Welcome to the next dead end!

Since customers like to question whether our negative statements are not in fact a disguised attempt to encourage more customers to switch to our IFS servers, we have taken a close look at our customer's viewpoint and made a direct comparison.

Therefore here is a simple, if somewhat limited, scenario.

Everyone is likely to be familiar with the Amazon AWS Cloud from hearsay, and the customer on whose enquiry this text is based is certainly familiar with it. Technically, in our experience, this is similar for Firebird with almost all cloud providers, as Microsoft's Azure Cloud etc. also work similarly and are often well suited for other software, but unfortunately not for Firebird.

In the absence of practical experience with Amazon AWS, because they do not offer Firebird support directly in the cloud, but only for MySQL or PostgreSQL, users of Firebird-based software still have the solution of installing their software there completely, including Firebird as a server, on an instance.

A preliminary test on an Amazon AWS instance

Since we initially only wanted to use this for our benchmark test, we set the basic data on the smallest instance for this test purpose.

We were given these values as a suggestion:

512 MB RAM, 1 vCPU, 30 GB SSD
Windows Server 2019
Frankfurt, Zone A (eu-central-1a)

The RAM was relatively low, but perfectly adequate for our standard benchmark and 30 GB SSD space sounds quite good.

The fact that Amazon allows a great deal of customisation of the configuration with AWS for an additional charge, can be gleaned from the 1,050-page PDF here:

<https://docs.aws.amazon.com/general/latest/gr/aws-general.pdf>

But before you print it out on the high-end department printer and select duplex mode to save paper, it's better to wait until your colleagues have already gone home...

The benchmark was of modest speed and even a virtual server instance that can be rented for EUR 10.00 or EUR 20.00 monthly from Hosteurope with Windows OS and 100 GB SSD manages over 40% in all categories.



Hostname: EC2AMAZ-74RC55S
 Instance ID: i-0a096b748cdaa2e2d
 Public IPv4 Address: 3.120.134.83
 Private IPv4 Address: 172.26.11.230
 IPv6 Address: 2a05:d014:e8a:a100:cf6e:e900:bfe4:a2
 Instance Size: t2.nano
 Availability Zone: eu-central-1a
 Architecture: AMD64
 Total Memory: 512 MB
 Network Performance: Low to Moderate

Test	Drive Test	CPU Test	Net Test
Create DB	22.56s	6.42s	173 iops
TMP	16.23s		
Threads	75.70s	58.47s	Thread Test
Result	34%	13%	4%

benchmark executed in 253 seconds

As you can see from the image, the Amazon instance, which will cost significantly more than EUR 10.00 or EUR 20.00 at the end of each month, achieves just 34% in the Drive Index, but is scarily slow in the CPU Index, i.e. with sufficient filesystem cache at Firebird level.

I won't even bother to mention the over 500% that our IFS servers achieve.

We were however surprised to see that they are working on a CPU that was released by Intel 6 years ago:

Hostname: EC2AMAZ-74RC55S
 Instance ID: i-0a096b748cdaa2e2d
 Public IPv4 Address: 3.120.134.83
 Private IPv4 Address: 172.26.11.230
 IPv6 Address: 2a05:d014:e8a:a100:cf6e:e900:bfe4:a2
 Instance Size: t2.nano
 Availability Zone: eu-central-1a
 Architecture: AMD64
 Total Memory: 512 MB
 Network Performance: Low to Moderate

Test	Drive Test	CPU Test	Net Test
Create DB	21.53s		
TMP	16.59s		
Threads	81.38s		Thread Test
Result	32%		

Load and execute benchmark script file... runtime: 191 s

Task Manager: CPU 100% 2.30 GHz (Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz)

The fact that the Intel Xeon E5-2686 v4 CPU supports an impressive 18 CPU cores is more than reasonable from Amazon's point of view as a provider.

But the relatively low frequency of just 2.3 GHz (which is also reached as the maximum in the task manager, although the CPU could theoretically reach a turbo mode of 2.7 GHz with all 18 cores, but this is probably



only provided by Amazon at an additional cost) also ensures that the performance that one might expect as the maximum nowadays is not achieved.

In comparison, Intel's flagship, the i9 series, has about three times the single-thread and also about three times the multi-thread CPU performance according to PassMark. The fact that it is more expensive is a moot point, but the surcharge is more than economically justifiable considering the waiting times of employees who need to be paid.

But: due to the cloud solution, only one CPU core of the 18 physical cores is available to AWS customers in the above-mentioned package, and is limited by the cloud solution that Amazon uses for this. This does not at all mean that on this server 18 available CPU cores are provided exclusively for a maximum of 18 paid cores, which, for example, 18 different customers have ordered. We don't know Amazon's figures, but I would expect a minimum of 50-100 Core Channels sold in parallel on this 18 Core CPU.

For the sake of simplicity, let's assume for the calculation that we are talking about 100 core channels and then return to the physics.

The above CPU has the socket type 2011 and this number has a relatively simple meaning: Simply turn the CPU over and count the contacts. There are exactly 2011 on this CPU.

So if all 100 virtual cores were to demand I/O power at the same time, then there would be just 20 lines available on average for each core.

Whether 10 cores are allowed to work serialised on 200 circuits one after the other makes little difference, because in addition, many of the CPU contacts are not responsible for memory I/O access.

And don't forget, a Firebird database is relatively undemanding in terms of CPU load, but an extremely demanding byte-pusher. The performance determines the maximum clock rate and thus the maximum speed of the connection of the CPU registers to the first/second/third level cache.

From here, the power then goes through the above-mentioned 2011 contacts to the mainboard memory controller, which then transfers what is to be read or written in so-called pages from the main memory to the CPU again through the 2011 contacts, or writes back to the main memory on the return journey.

If something that the CPU would like to see is not in the working memory, then the content of the pages is loaded from the data carrier into the working memory via memory controller chips on the mainboard and from there into the CPU if required.

Conversely, the memory pages in the working memory that are deemed worthy of storage by the software on the CPU are sent back to the data carrier and then hopefully reliably stored there.

That is also just simple physics.

In cloud systems, performance problems are often not only caused by the above-mentioned 2011 contacts, regardless of the MHz or GHz number with which they can transmit data to the storage per contact. Often cloud systems, as in our example above, are not even supported by fast multi-channel data carrier systems with high tact, such as NVMe PCIe 4 onboard drives, which can be found with 3-4 GB and ≥ 400000 IOPS per second write and read performance at reasonable prices.

However, if 100 processes working in parallel on 18 physical cores are primarily concerned with I/O read and write operations, then the data carrier connection must implement this in such a way that 100 small packages



are cached in a short time frame and the cache is released again as quickly as possible after writing, so that the cache is then available again for further I/O operations.

So, in the hope that not all of the virtual cores that the customer has paid for will cause full I/O load at the same time, the vendor will not provide a disk on every motherboard with maximum connectivity speed for large files, where GB per second is important, or many small files, where IOPS is important.

A look at the Amazon AWS documents (see PDF link above) does not give a clear statement on guaranteed IOPS, but defines so-called IOPS quotas.

Or in other words, no relevant lower limit is defined, but only an upper limit that the customer can expect, but which, for whatever reason, is not always attained.

Without having analysed this exactly, the highest IOPS quota value found in the PDF is 30000. Dedicated Firebird hardware, such as the IFS server, on which only Firebird-relevant I/O load is generated, can read and write 3-4 GB per second and also read or write back 400000 IOPS, i.e. small data packets per second directly from the database file. These are values that the IFS actually achieves.

However, cloud providers usually rely on their own dedicated storage systems that receive the data to be written via network cable somewhere in the network or read the data requested by the CPU and send it via the network to the mainboard of the CPU. If we assume a 10 G network connection, then you get a maximum of 10 Gbit of data packets in or out over the wire. Bit and not byte. In bytes, that's about 1.1 GByte per second, if nothing else is being transmitted on the cable. Sounds quite OK, actually.

Back to physics: Most readers of this document have certainly already dealt with the installation of a SATA hard disk or SSD. The SATA protocol is actually quite powerfully defined and has only reached its upper limits with SSDs. Whilst SATA HDDs can rarely read or write more than 150 MB per second, the maximum 600 MB per second achievable with the SATA 3 standard is quickly reached as the upper limit for fast SSDs. With the SATA 3 protocol with up to 6 GBit/s however, maximum cable lengths of 60 cm are permissible. With SATA 2, this was still 90 cm for the maximum possible 3 GB/s at that time.

If you try to connect a SATA 3 HDD or SSD to a SATA 3 port with a 90 cm cable, it will drop back to SATA 2 mode or even SATA 1 mode, but depending on the cable it may simply not work at all, even though the cable has identical connectors. At least, that is my experience and based on the physics of the latencies or frequencies of the signals on the cable.

Extremely high data volumes cannot be transmitted via cable at any clock speed. You can recall Mr. Einstein again in the back of your mind with regard to fundamentals and side effects, because he knew that the speed of light and thus also the speed of voltage signals in cables cannot exceed an upper limit. Otherwise it would be easy to connect NVMe SSDs directly to the mainboard somewhere in a closet with a 50 m cable and they would then have the same speed as onboard. IT managers may ignore this constant when selecting their systems, but this still does not invalidate it.

But that's right, there are network connections, such as the 10 GBit Ethernet connection mentioned above, which can use much longer cables. Without discussing all the physical basics of network technology: Assume that these amounts of data can only be achieved unidirectionally per wire pair in the cable with large data packets, where the CPU requests something from the data carrier, but does not have to wait for its response to be able to request the subsequent packet.



Why does it take longer?

In the network, most connections are realised with the TCP protocol. When a data packet is sent, the sender receives a receipt, which gives him the certainty that the packet has actually arrived. There is also the UDP protocol, which works unidirectionally, and where packets that have not arrived are of no importance. Live streaming on TV, for example, is seldom bothered by the fact that individual packets have not arrived at all, because with a sufficiently good connection a few faulty pictures here and there are frequently not even noticed.

However, whether the transaction that was just triggered by your application concerning a stored record is written back from the CPU's memory to the database file on the disk, in a way that is considered as correctly stored in the event of a server failure, is a completely different factor in this type of application.

High-end NAS/SAN storage systems therefore usually offer connections that deviate from the TCP protocol, however here the problem of cable length remains.

Whereas with onboard NVMe this is done by the chip for the memory manager in interaction with the directly connected RAM modules, in the network several chips on both sides of the network cards are involved and, depending on the length of the cable, these must then adjust to the transmission time and wait when something comes back. And have you ever wondered why NVMe SSDs need significantly more contact connections than just a 2-wire cable? Not to mention the more than 200 contacts on memory modules.

For single packages, it is advisable to simply enter the following commands in a DOS window and ask yourself why it takes so much longer if the distance is longer (Mr. Einstein could help there).

```
ping yahoo.com.au  
ping yahoo.com  
ping yahoo.de
```

The network hubs through which data packets pass can be displayed with the command *tracert*. Often they do not take the most direct path and at each junction local CPUs in their routers make sure that your packet is transmitted in the right direction to the next cable.

Depending on the network connection and your own location, a geostationary satellite may also be involved. The distance from the transmitter to the satellite and then back from the satellite to the receiver is at least an additional 2 x 36,000 km. At a maximum of 300,000 km/s, this can be a few hundred milliseconds, but please do not forget that the TCP protocol also needs the receipt packet, i.e. the same time back again.

On a small scale, this is nothing different than what is happening in your network. The only difference is that in normal operation Firebird does not request just one data packet for reading or writing, but sometimes several thousand per second, if the hardware is capable of handling them. And the particular thing about Firebird is that the data packets have to be read and written in certain sequences.

It is precisely this latency between the data carrier and the CPU that is actually handled very well in a 10 GB network protocol connection. Strange constellations caused by application programmes, which are based more on IOPS than on MB/s, then ensure that the software is slower than you actually think, although you have optimally selected all the modules involved.

The IT managers who, when selecting a virtualisation solution as in the first example, have ignored the physical basics are actually facing the same dilemma that customers face with cloud computing. It should



be faster, but it isn't. And it won't become faster by adjusting a few configuration parameters on your own hardware or purchase various add-ons from the cloud provider.

In the case of your own hardware solution, at least you know which CPU is doing the job and by adjusting the configuration, presumably in the BIOS, you could for example, increase performance by activating turbo mode. With the cloud solution however, you have to reckon with the fact that the virtual instance that has been running on the 2.3 GHz CPU today will be moved to some other server tomorrow because your load profile on the CPU, together with the other 99 processes, has caused too much load. Therefore, you now have to work on a different CPU that may only allow 2 GHz or even less, unless you pay extra for add-ons ...

A vicious cycle of dependence!

Conclusion, or how we can implement a solution for you

There are data centres, such as Hetzner.com, which offer very powerful virtual systems for comparatively little budget, and you can choose the location yourself. Unlike other providers, Hetzner also offers a so-called colocation, in which we operate our powerful dedicated IFS servers for Firebird databases for you and also maintain them there for you. We can operate servers in your own colocation, which is nothing more than a cabinet in the data centre, which can be freely equipped with 14U (EUR 99 per month plus energy) or 42U (EUR 199 per month plus energy).

Due to high demand, there are temporarily no free colocation racks of certain types available, but we currently still offer sufficient free slots in our colocation in Nuremberg if you would like to commission us to operate and maintain one of our dedicated IFS servers there. We charge an additional fee of EUR 600 per server annually for this service.

Should you wish to realise special applications directly in the cloud, e.g. as a terminal server with the Firebird server as the backend, or for completely separate applications without Firebird, as virtual instances as a terminal session, then it is possible to book and operate these directly in the data centre in Nuremberg. Your application is launched on servers in the data centre, access to the Firebird server on the extremely fast IFS hardware is performed with latencies in the range of 0 ms, i.e. as if you were running the servers in your local network. With a sufficiently fast internet connection from your head office, or when field staff dial in using a smartphone/mobile phone, or employees access from their home office, everyone has extremely fast access to their own terminal server instance and the application software running there benefits considerably from the IFS server directly available in the same data centre network.

You and your employees can work with simple terminal client systems or with your own PC or laptop, even from home or on the road. The costs for your own data centre are significantly reduced because measures such as fire protection, fail-safety, etc. are covered by the data centre as part of the basic price. The hardware and software for the terminal servers are taken care of by the Hetzner data centre and you do not even need to buy any Microsoft licence packages.

IBExpert then takes care of the set-up and operation of the fast Firebird IFS server(s) as part of the RemoteDBA maintenance contract with maximum security, up to and including real-time replication. You can concentrate on your actual business and add as many virtual workstations as you like, or cancel them if you no longer need them. For your workplace, a simple all-in-one computer or small NUC computer with a screen connected to the RDP session at Hetzner is usually sufficient.



IBEXPERT WHITE PAPER



And yes, a workstation printer might also be useful ...

Less is sometimes more.

Of course, we can't guarantee whether your hardware-focused IT management will agree, but we can help you to share opinions.