Novelty in storing ERP data

A. Selmeci*, T. Orosz**
Óbudai Egyetem AMK, Székesfehérvár, Hungary
E-mails: *selmeci.attila@amk.uni-obuda.hu , **orosz.tamas@amk.uni-obuda.hu

***Abstract*— ERP systems are at each company, nowadays as standard and basic applications. The data managed within an ERP system should be accessed differently according to the daily habits. To store and access data has been changed during the last years, and according to the requirements it should be changed further.**

Among others the ERP software architecture, the disk access, the bandwidth, the latency, the disk number and type are main factors in speed and reliability. In our paper we compare the different approaches from ERP point of view and show a new direction, which can be a manageable innovation for companies to better store ERP data.

Keywords: **Storage, SSD, FusionIO, Local storage, Architecture, ERP, Innovation, Sustainability**

I. INTRODUCTION

To install an earlier ERP system we had to design the disk layout accurately keeping in mind the behavior, functionality, and mechanism of many different hardware and software components. Some of these components should be mentioned, like the ERP solution itself, the underlying database management system solution and the operating system. The business data is stored generally always in database tables with primary keys. In some cases the relation between the tables are defined in the database level, but there are ERP solutions, where the table correlations are defined on application level having own dictionary for that. In both cases the data is read from (and stored in) tables, which are not always accessed by unique key attribute values, but several times by not key attributes. To have an acceptable response time from the database layer indexes should be created for the frequently searched attribute sets. The first database system based ERP systems required special attention to isolate table data and index parts to separate disk areas. The main reason was the speed, because the database can read the index and data parts simultaneously in that case. The data store of ERP systems as technique and method can be divided into three main areas:

- Business data representation in the ERP system (logical view)
- Database storage of the internal representation (technical view)
- Data or disk storage technology (physical view)

These different approaches cannot be handled as separated, unequally replaceable methods, but they are rather layers, which not in any case are really distinguishable. They have to be managed together as close as it could be. But different groups, companies develop the different layers, so it is not really possible to handle them together.

We show how the different layers envelope and how to follow them to achieve better performance in ERP environments.

II. GENERAL OVERVIEW

In the hardware world the physical layer was improved in the last two decades. The two main technical goals were generally the reliability (e.g. redundancy, high availability) and the response time, speed (e.g. load distribution, bandwidth). The early data stores used different RAID (redundant array of independent disks) technologies to approximate both goals together. The capacity and the speed of the disks increased and it was available to build and use larger data stores. The first RAID controllers were hardware components only, but afterwards some software configuration tools were offered to manage, design the disk distribution in a server. This direction founded the early storage system and solutions having specially connected disks with special-purpose hardware and an operating system offering many different features and services.

According to our study the storage vendors did not really follow the new hardware developments, options, but only the palette of services grew in a wide range.

Nowadays the storage systems offer several redundancy, high availability, data handling (like de-duplication), backup, fast copy, read cache and other options. These features are very useful and required in some cases, but the costs are much more higher and the value that they provide. On the other hand the speed as a goal is not in the main focus of the vendors, because they advice only faster disks or even flash drives (SSD-s), and of course more disks to distribute the load among the spindles. These directions are correct but with having many features, special connectivity layers, like Fiber channels and switches, the disks, the real data storing surfaces are far away from the database server or even application server CPU-s. The newer and faster hardware elements made easier to use bigger, faster and reliable 'local' disks in a server. The local disks or even PCIe storage cards (like FusionIO) provide data in a much higher speed limit, which are sometimes necessary. Our measurements proved that less CPU power is enough if the data storage layer can faster provide data for the ERP system (in case of commercial RDBMS it could decries license amount as well).

The top layer of the above mentioned three data handling approaches determine the mechanism of data management and data store in lower layers as well. Each ERP system contains several business modules and functions, which handles business objects, like accounting documents, employees, sales or purchase orders, etc. The business objects can be defined as object oriented entities

with status (defined via properties), object modification functions (declared through class methods), and of course published status modifications (managed by events) to inform other objects about status change (like employees were hired or even fired). The business objects on technical level in an ERP system are not always defined as real object oriented entities, but only logical correspondence and relationships are defined between data or even tables. Many ERP environments can manage them as objects in higher level, but each time these objects are stored in database tables separating the current status to several tables connected in the database (or as described above in ERP system) level via foreign key relationships. The object as a unit or even entity cannot be recognized on database level.

According to our study for ERP and other kind of OLTP (Online Transaction Processing) systems the object based data handling on logical and technical level would be more effective that the currently widely used table based representation. This direction goes forward to the database layer, where the object oriented database solution would handle better the ERP or OLTP requirements than the relational database management systems.

Our paper collects the availability of the offered technologies, which could be used for ERP systems and architectures as future environment and technical solution. We try to refer to data-warehouse systems as well pointing out the differences in architecture and requirements positioning the possible directions. We figure out the advantages and disadvantages of the different technologies, approaches and try to sketch a way, which leads to now opportunities of data store for ERP systems.

III. PROGRESSION OF PHYSICAL LAYER

The physical layer was developed a lot with solid-state solutions. Figure 1 shows some of the different possible storage usage. As we see close to the CPU we can only use smaller capacity, but by increasing the distance the capacity will increase as well.

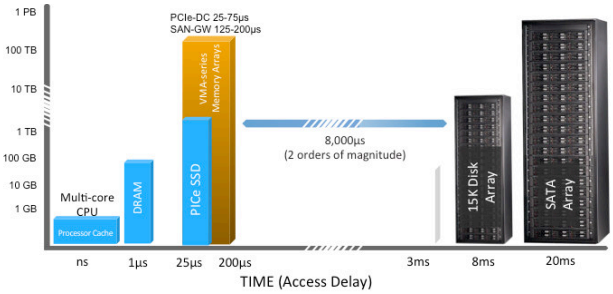

Figure 1 - Solid state storages (HPE ©)

If we consider a machine, there the I/O bus, where the data goes thru in case of local disks or DAS (Direct Attached Storage, see later as well) solution as the Figure 2 shows. The latency is almost the same, but the bandwidth and IOPS (IO per second) depends on the speed of the used disk or DAS containing the disks. If we use SSD instead of HDD (e.g. SAS, Serial attached SCSI) we recognize a magnitude in the speed. Technically the latency, IOPS and distance to the CPU can be used as KPIs for storage systems. [1]

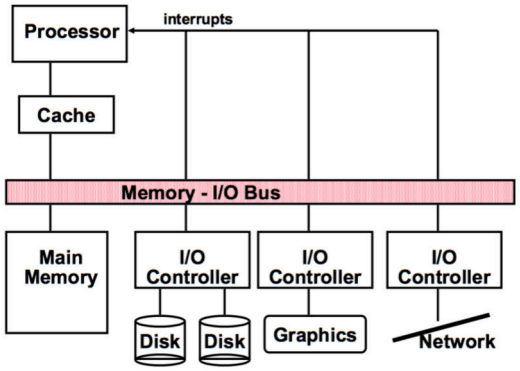

Figure 2 - I/O bus in general PC architecture

Certainly the main memory and the CPU caches are much more faster. (The currently used DDR3 and DDR4 RAMs have 192 and 204.8 Gbit/s bandwidth.) The memory speeds (10-100 times slower than CPU L1 cache) and bandwidths values are much higher than the rates of disks (HDD access time is 4-5 order of magnitude). If we think about using faster storage, than we should go closer to the CPU. One possible way is using the PCI express connections as shown on Figure 3 below (source: https://www.design-reuse.com). (As we can see on the Figure 3 the graphical card is on the PCI express bus as well. We can use the GPUs (Graphical Processing Unit) not only for gaming, but for calculations as well. See later in this paper.)

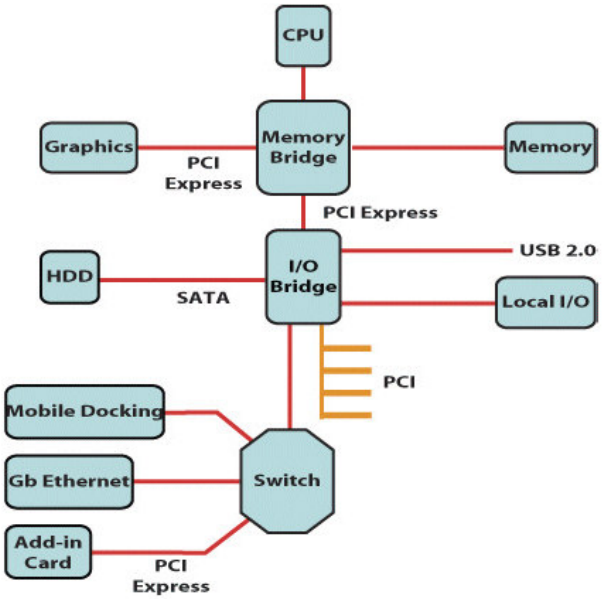

Figure 3 - PC architecture with PCI express

The PCI express solution is using switched topology. It helps to exploit the bandwidth and enables multiple extension opportunities. As the Figure shows the SATA and PCI connections are much further to the CPU than the PCI express bus. There are some products like the first fastest solution: FusionIO (SanDisk acquired in 2014 and in 2016 Western Digital acquired SanDisk), M.2 or similar. So with such memory (SSD) card we are closer to

the CPU, so the latency can be omitted and the bandwidth is what the PCIe can provide.

There were some measurements made by HW companies as well to see which solution give better performance. We should keep in mind that the speed and reliability are working in opposite directions. So if we use a faster solution, like PCI express card for storing data, the reliability and available may be suffered (see later).

As we write the disk and memory speeds are not comparable, but using SSD and PCI express we can narrow them to each other. These cards are also called NVM or NVMe (Non-Volatile Memory (express)) cards. In the future we suppose the use really NVMe cards instead of memory if the speed is just as good as the RAM has [3]. It is a very good opportunity to use such memory cards, because the used memory should not be recreated and rearranged if a system (like ERP) should be restarted. The processes should only reconnect to their memory areas. Some applications before stopping generate special memory dumps from the used important caches, buffers to be able to provide a faster start time. These content can help in it by loading the data directly into the place where they were before stopping the system. The old UNIX systems (like HP-UX; Solaris, AIX, etc.) are not really prepared for this usage. Therefor the several Linux distributions and MS Windows server solutions are more beneficial. (As we have tested with an Oracle database, the Mac OS can be used as well with such solutions, but technically these machines are not really designed for data center usage.)

The Figure 4 shows a similar approach where the internal storage areas and the storage system were measured and the calculation proofed that the storage systems are very slow compared to the internally used solutions.

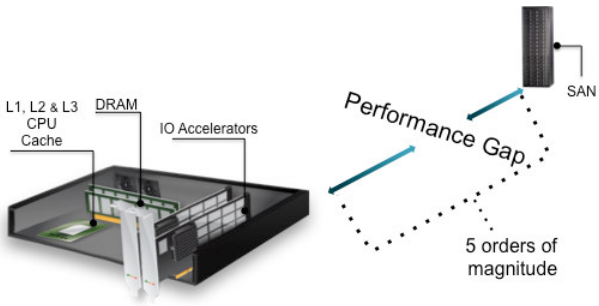

Figure 4 - Data distance from CPU (HPE ©)

IV. STORAGES AND ACCESSING THEM

As the Figure 1 and 4 show the further the data from CPU is, the higher the latency is. Not only the latency is a problem, but if we have a storage system and we put several SSD instead of HDD, though the SSD can be faster e.g. 30 times than the HDD, the whole speed will not be so grate. It is easy to think about. The storage is a computer with CPU, RAM and operating systems. It runs the program, which manages the data through the disks. Some components are built from hardware elements, but others are still software parts. The disks are attached to this internal CPU using IDE, SATA, PCI, and SCSI buses. The nodes of the storage if available have wide internal connection. The internal program should manage the disks and the defined layers of the storage. Some features like snapshots (creating logical copy from a volume) or de-duplication (store the same data only once and refer to the original) can also be available, but these will decries the overall performance as well. The faster elements, like SSD are connected on the same way, so the bus speed, internal calculation will derogate a part of the disk performance in the direction of outside world.

That is why the storage vendors offer PCI express accelerator cards into these storages. These cards solve as a cache in these machines offering read and write cache as well. Other usage can be in hierarchical storage management to provide different speeds for data storage and retrieval. In this case the storage CPU manages the data movements permanently between the different layers according to the measured usage statistics.

To access the data there are many options. Some of the database management systems under ERP used so-called raw devices (many of them can use row devices nowadays as well). It means that they write directly to the disks, they do not use file systems on them and they bypass the operating system as well. The file systems can degrade the disk access IO performance, which is very important in case of databases. The raw devices can be bound to block devices as well. Here is the point what is also interesting for us. We can use different file systems and access methods on disks. As the Figure 5 shows, there are generally three used storage environments: DAS, SAN and NAS.

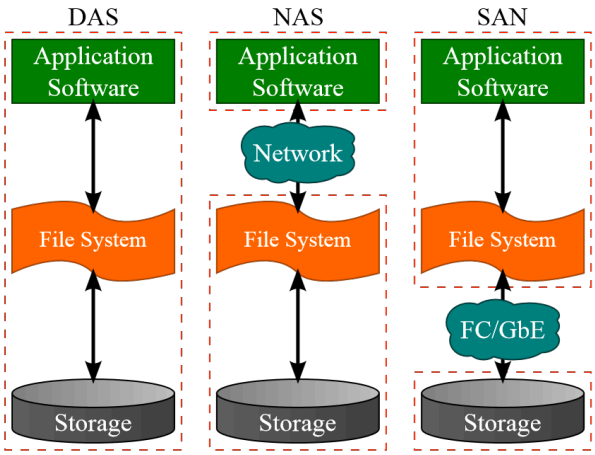

Figure 5 - Storage networks (source: abdullrhmanfarram.wordpress.com)

The difference is not only on the access, but also on the real storage on the disk layer.

- **DAS** (Direct-Attached Storage): disks are connected to server locally (internal DAS) or via SCSI or FC (Fiber Channel) building an external DAS. External DAS: bigger data amount, slower access (distance and protocols). Cannot be share between compute nodes.
- **SAN** (Storage Area Network): multiple hosts can use multiple storages (traditionally) via FC-switches (a kind of virtualization). Special SAN network should be built and the compute nodes should have SAN cards (in most of the cases FC-cards).

- **NAS** (Network-Attached Storage): IP based network and file sharing. Generally CIFS and NFS. Own operating system providing protocols, and storage features (snapshot, de-duplication, etc.).
- **IP-SAN** (Internet Protocol SAN): IP based SAN using iSCSI (block-level access)
- **CAS** (Content Addressed Storage): not frequently changed data, data objects (archived data) can be stored as object. Many benefits for object stores and prices, using redundant array of independent nodes (RAIN) communication via high speed interconnect channels. (Optical or data archives.)
- **VSAN** (Virtual SAN): this network technique is generally part of the VMware solution. It offers SAN access to disks managed and offered by V-Sphere.

These (NAS, DAS, SAN) storage networks can differently access the data on disks. It is important to see on Figure 5 that the red dashed line illustrates the computing unit. In case of DAS we have one unit, because the disk is attached directly to the server. In the other two cases the compute node (above mentioned as "Application") and the storage node are separated. The disks are attached to the storage node and there is a network connection between the storage and compute nodes. The main difference on the Figure 5 is the network place and type, because in case of NAS we have Ethernet network, but in case of SAN we have the fiber network.

There are many opportunities to connect the disks, virtual disks or storage provided disks to the compute nodes. The Figure 6 shows some example.

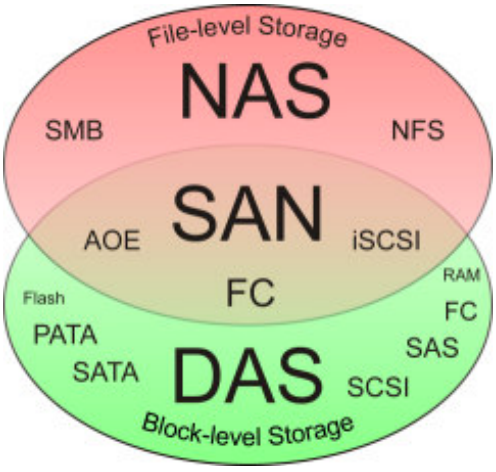

Figure 6 – Disk accesses on storage networks (source: mierebustelur.blogspot.com)

The importance of this Figure is in showing the access levels:

- Block-level access: the data is stored on disks as blocks and accessed by logical block addresses. Databases store the data in the database as well using blocks. The size of the database block and the storage block can play big role in performance (size is database activity dependent).
- File-level access: technically uses block access, but it is hidden, because the files are accessible by name and path.
- Object-level access: the data belonging to a logical or application level object are organized and managed together as one unit. (Not really used for general applications, but for object storage in clouds.)

We do not want to list the shown connection type, though some of them were already mentioned in this paper. They are not in the focus from perspective of our paper.

On the other hand, it is important to understand by the companies using ERP systems that the different data access can help in some special cases. Generally the file access is better if we have many smaller files, like pictures or log data, but block access can be helpful if the application reads binary data blocks. The block sizes should be comparable between the application and storage level. The ratio between them can increase or damage the performance. Theoretically we can say that the SAN is better for database access because the fiber channel should provide "light speed" access to the data and the block-access helps to read the required database blocks from the device. We have to consider that the block access is very difficult, because the data is not really tracked and can be easily damaged. That's why we use file systems to store and access data not using block addresses on compute nodes. NAS gives a different approach, we access the file systems of another computer in this case the storage nodes. It can be used in SMB or HOME environment if we have only pure network. But as the Figure 7 shows we can reach much better network parameters as well.

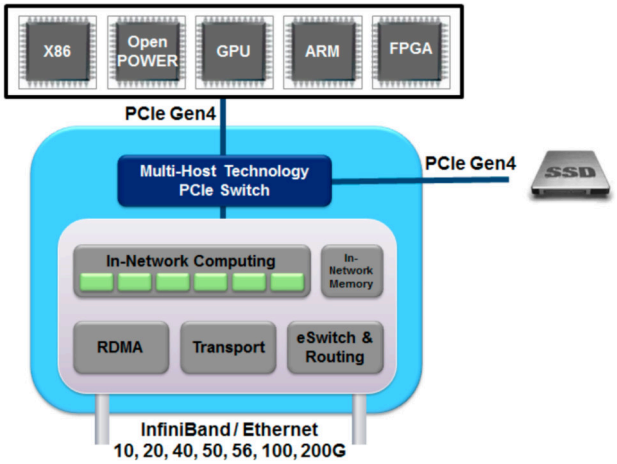

Figure 7 - Mellanox Infiniband card ©

There are some vendors, who could provide 100G or better Ethernet over InfiniBand. If we have so wide network access, the file-level access is much better than the SAN storage networks. First of all the FC card speed is limited around 16Gb, which is much less than 100 or 200Gb. (There are new 32-128Gb FC adapters (gen6, gen 8), but these cannot be connected to UNIX systems and the switches does not support each. [6]) Though these newer FC cards are attached to the PCIe bus, so they communicate to the CPU very fast. If we calculate all these, the result shows that a NAS could be faster even if the FC uses fiber cables.

We should also consider the Software Defined Storage (SDS) systems. These can bring some new ideas as well. If we listen to the voices of the storage arena we hear nowadays about convergent and hyper convergent storage techniques. These are very interesting and useful directions in case of ERP systems. The converged storage just tries to put the ideas together (but goes in another direction). In such a solution we integrate the compute node, the storage and the processing as well, so the latency can be small, the speed high. This technique is optimized and designed for virtualization and cloud platforms. The virtualization is not a bad idea in ERP environment. It could be used in many layers [2]. But we think according to our experience, that virtualization can decrease the disk performance in the database environment.

We can go further to see the hyper-convergent storage idea, where the storage nodes are connected to each other to build a virtual storage layer. In this case we can use DAS for faster storage nodes and InfiniBand network environment among the storage nodes. Each node can manage the disks using redundancy and we can have data copies among the nodes as well. Here we have the possibility to use other storage features as well, while we lose some speed (the features are always costly staffs). This solution is very tricky, because we can use even DAS for the storage nodes, but the communication to the compute nodes are made via NAS using the same InfiniBand network.

We have built such architecture under a production SAP ERP (HR) environment as shown in Figure 9.

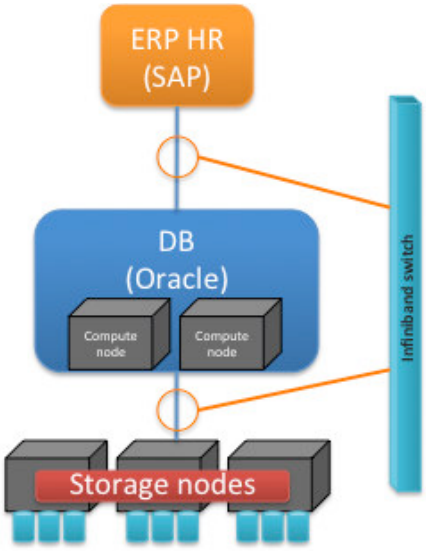

Figure 8 - POC environment with SAP ERP

The storage and compute nodes were Linux machines. We have to mention that the compute node is not the application level in this case, but the database engine (Oracle in this case) level. We have run the Oracle RDBMS on two nodes using ASM (Automatic Storage Management) volume manager and file system. The SAP system was installed on a similar machine running MS Windows Server as operating system. We had another environment built by the company experts. That configuration had a vendor delivered all flash SAN storage system and a huge, but virtualized DB server connected via FC cards. The SAP application layer was attached via standard 1Gb Ethernet as shown on Figure 9.

We have compared the present production environment with these two solutions using the same SAP ERP system. The result was impressive. Our small HW elements using SAS disks attached directly to each storage nodes and InfiniBand and NAS solution performed better (120%) then the present larger environment. On the other hand the new storage system fully loaded with SSDs and the preferred virtualized hosts could provide less performance (80%) than the original one. This information was a good example to the IT leaders to think about new technologies, services and solutions. It requires different approach on standard support and maintenance, because the workers should learn and manage a self-made environment, where they could take part on the implementation phase. In the other direction they learn and use products from vendors. They have ready solutions and limited compatibility. The new generation in ERP IT management should be able to work in a DevOps and LiveOps manner.

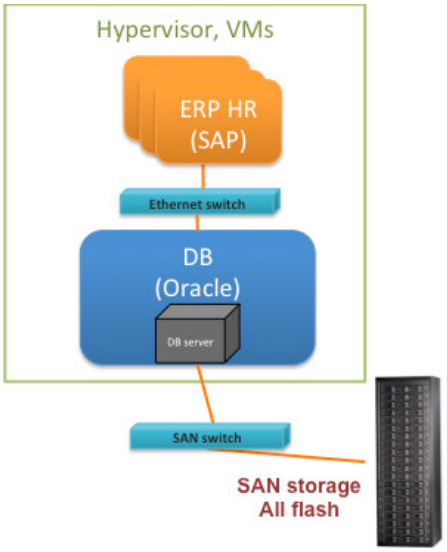

Figure 9 - Storage and VMs

V. OUTLOOK FOR OTHER SOLUTIONS

We have mentioned earlier that the GPU elements can be used as well. It is not really a storage area, but the graphical card designers work hard on providing better, quicker cards with larger amount of memory. Parallel they enhance the programming capabilities as well. The graphical cards are connected via PCI express buses, so they can communicate to the CPU efficiently. In an ERP environment we can think about other problems as well, like making statistics or planning according to the data. Some of these information need large amount of processing capacity, like ML (Machine Learning) or AI (Artificial Intelligence), others require larger dataset to be work on it. The graphical cards have several processors with smaller memories and shared memories. The problems, which could be parallelized, can be shifted to GPUs. Some special development is needed, but the cost and result will be better.

We have tried out another direction with ERP data as well. We have implemented the MAPD GPU based database system as it was released as community product

as well. We had only one graphical card in a server, but we could load many financial transactions and connected data to be searched. The result was amazing.

It can be imagined as an in-memory database having many processing units in one single system. It is useful for statistical analysis on read-only data sets. We plan to test in with ERP connection as well in the future.

VI. Conclusion

Almost the main point or component of an ERP system is the database. It should provide the data fast enough to be able to process it on application level. There are some special read-only data, which can be handled using other methods. The access time during write and read depends on the distance of the data and the CPU and the latency of the access. There are many technical, physical developments, where new technologies can and should be used under ERP environment.

The disk speeds are higher and the connection possibilities provide wider bandwidths as well.

We should consider not only use SSDs instead of HDDs, but relying on the practice, where more slower drives (or spindles) can provide faster environment, than less, faster and larger drives. We have also figured out that the fastest drives connected via PCI express bus to the CPU can provide speed, but the redundancy is needed for reliability in case of ERP systems.

The network speed and latency between the storage and compute nodes (where the database is running) are also important. Using wider bandwidth we can reach higher performance. That's why we recommend to use, but at least test the NAS architectures based on InfiniBand switches.

Our test environment presented that using software defined storage and having appropriate DevOps groups we can reach much better performance and reliability on lower price than the vendor provided solutions.

Our tests and paper has shown that the companies should think about using the new features under their ERP solutions.

References

[1] A. Selmeci, I. Orosz, Gy. Györök, T. Orosz: "Key Performance Indicators Used in ERP Performance Measurement Applications", 2012 SISY, Subotica Szerbia, pp. 43-48., ISBN:978-1-4673-4751-8

[2] A. Selmeci, T. Orosz, Gy. Györök: "Innovative ERP virtualization", 2013 SISY, Subotica Serbia, pp. 69-75., ISBN 978-1-4799-0303-0

[3] http://www.roceinitiative.org/wp-content/uploads/2017/01/RoCE-Accelerates-DC-performance_Final.pdf, ROCE 2017

[4] https://mierebustelur.blogspot.com/2013/06/macam-macam-storage-network.html, 2013

[5] https://abdullrhmanfarram.wordpress.com/2013/04/08/storage-technologies-das-nas-and-san/, 2013

[6] https://fibrechannel.org/roadmap/, 2018