White Paper

Even Faster Data Transmission

Discover RMDT – the Reliable Multi Destination Transport Protocol for Big Data Delivery

FILa
Future Internet Lab Anhalt

Future Internet Lab Anahlt by University of Applied Sciences Anhalt in cooperation with Dexor Technology GmbH
2020

Content

Introduction: The problem with multi-gigabit big data transport

Have you wondered why, when transferring data between your servers, the transfer rate rare reaches even half of the bandwidth you paid for? This is especially problematic for connections with a large bandwidth or a high delay. Maybe the problem is the cross traffic? Maybe it's just that the other half of your bandwidth is used by another user or application? But if that is the case, why doesn't the data transmission over private networks show better results either?

We have solved this problem with the development of the Reliable Multi Destination Transport protocol (RMDT). RMDT is designed to efficiently transfer large amounts of data over links with high bandwidth and latency. RMDT outperforms conventional TCP-based setups in all tested categories and can significantly improve the capabilities of your network.

Enough bandwidth for everyone?

Today, we consider modern high-performance networks to operate at speeds of 10, 40 and even 100 Gbps. However, the era of high-speed Internet only began recently. About 10 years ago, 10 Gbps interfaces became an important part of the system integrators' portfolios, increasing the common bandwidth of production servers by a factor of 10 compared to previous hardware. Nowadays, we have again increased the bandwidth by factor 10 and this growth isn't likely to stop soon. Although having such large bandwidths sounds promising, it does not automatically mean we will be able to use these benefits to their maximum for our application traffic.

The fairness problem of Internet

Consider a scenario where data is sent over a network path that has a data rate bottleneck due to a component slowing down the whole connection. As the result of interaction with cross-traffic in this "narrow link", the data rate of our transfer method drops dramatically. In the vast majority of cases, the reason for this drop is the fairness of the transport method.

The Internet was developed with a *fair* distribution of resources in mind. As a result, transport protocols are required to be fair to each other: they have to share the available bandwidth equally. While this is great in theory, it doesn't work very well in practice – even the latest version of the most frequently used Transmission Control Protocol (TCP) is widely unfair. TCP acts unfairly even to itself when using different delays for different streams. The stream with the lower latency in such a scenario is more aggressive than a stream with a higher delay, thus disadvantaging the higher-delay stream.

Created without streaming in mind

In the early days of the Internet, the idea of fairness seemed quite promising. All users should share all available resources equally. Today, however, there are many applications that do not obey the fairness rule, for example streaming services, such as video streaming, Video over IP, and Voice over IP. These applications produce real-time traffic and emit their data with a fixed rate, thus aggressively ignoring the fairness principle of TCP in regard to other network traffic. Considering that streaming services are rapidly gaining popularity and thus further growing their share in the total Internet traffic, your reliable data transfers will be increasingly repressed by streaming services in the future.

One option to circumvent the problem, although a very expensive one, is to deploy global L2 connections for big data transport. A much better and more cost-efficient solution is to use the RMDT, which provides a tradeoff between the fairness towards TCP and bandwidth exploitation in the presence of aggressive streaming traffic.

TCP: The algorithm is the bottleneck

The Transmission Control Protocol is the de-facto standard for reliable data transmission across nearly all IP-networks. Even with the increasing portion of multimedia traffic, about 90% of the bytes and packets transferred over the Internet are sent using TCP. To avoid congestion in a network, TCP uses a mechanism which reduces the so-called congestion window whenever it detects congestion. This leads to a short interruption of data transmission. According to this congestion avoidance mechanism, packet losses are considered as a sign of congestion. However, in real networks, there are several different reasons for packet losses. If a packet loss is not caused by a network congestion, reducing the transfer rate is useless and causes an unnecessary slowdown of data transmission. While there is a variety of techniques how users can fine-tune TCP to make it perform better, the main reason for poor performance still remains unchanged: the protocol itself.

RMDT, in contrast to TCP, uses a novel rate-based congestion control and adapts the transfer rate independent of packet losses in the network. The protocol was designed for data transmission over networks with high bandwidths. To achieve this, it uses special algorithms for increasing and decreasing the data rate in order to act more efficiently in these networks.

What can be achieved with RMDT?

RMDT is a UDP-based transport protocol that was developed at the Future Internet Lab Anhalt of Anhalt University of Applied Sciences and is now actively marketed by Dexor Technology GmbH. The protocol was developed with the vision of creating a best-in-class protocol for fast and efficient data transfer and has proven to be superior to actual TCP implementation, as well as to proprietary data transfer solutions. The keys to success for RMDT are its algorithms for the analysis of the available bandwidth and very efficient data management in the protocol stack.

Furthermore, RMDT can not only operate in the classic domain of one-sender-to-one-recipient delivery, but deliver data with multi-gigabit speed to up to ten destinations simultaneously.

This process has propelled the project from its initial idea of overcoming the limitations of TCP towards the creation of a completely new, high-speed solution that fully utilizes the capacity of the available data bandwidth. To demonstrate the capabilities of RMDT, we have conducted a number of tests and compared the performance of several applications. The following experiments compare FTP, the most popular file transfer method based on TCP; UFTP, a reliable UDP-based multicast solution; and DataClone, a powerful RMDT-based application for remote file copy. The experiments are carried out for point-to-point and point-to-multipoint topologies, each with varying transmission speeds and MTU sizes.

RMDT behavior

Protocol is equipped with a set of congestion controls to satisfy different networking needs.

Fair share

The protocol will attempt to fairly share the bandwidth with another flows in the link. It means that this high-speed transfer solution will harmonically participate the resource sharing in your infrastructure. Resources will be equally distributed, so that no one transport in the network will get advantages, except, if other protocols, by their nature, cannot earn their share. In this case RMDT will earn all available bandwidth of the connection, and will respectfully let new transport stream join the share.

Low Priority

This mode allows for RMDT to operate gently in the network. Other connection in the network will not be affected – RMDT will not participate in share. However, it will use entire available bandwidth, which, by any reasons is not allocated by others transport protocols. Within this mode, data transmission will not affect your existing communication and you will benefit by using your infrastructure by the maximum. Experimental setup for point-to-point (P2P)

The topology used for this experiment is shown in Figure 1. In the experiment, two servers are connected to each other via a network emulator with a maximum link capacity of 10 Gbps. Each server is equipped with:

- CPU Intel(R) Xeon(R) CPU E5-2643 v4 @ 3.40 GHz;
- RAID array on SSD, with write/read rate that exceeds 11 Gbps (1400 Mbytes per second)

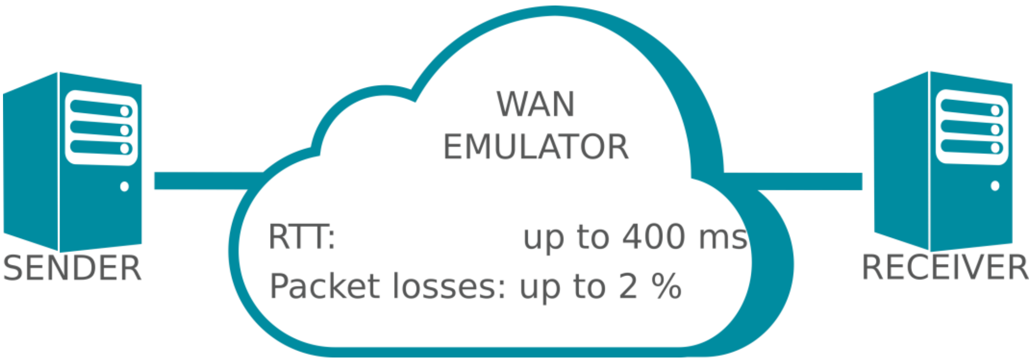

Figure 1. Topology for point-to-point data transfer experiment.

Within each set of tests, the sender transfers a 50 GB file to the receiver. The round trip time (RTT) changes in 50 ms increments from 0 up to 400 ms. Packet loss varies from 0 to 2 % in 0.5 % increments.

P2P on a 1 Gbps connection with MTU 1500 Bytes

The performance results of experiments that included FTP over standard TCP and DataClone over RMDT are shown in Figure 2. During this experiment, the maximum transmission unit (MTU) consisted of 1500 Bytes and the throughput of the emulator was set to 1 Gbps, which is considered the bottleneck of this connection.

As can be observed from the graph, FTP performed quite well in the network without delays. However, the increased RTT and packet loss values have a dramatic impact on the send rate. A noticeable drop in the performance can be observed starting from minimal value of observed packet loss: 0.5 % in presence of any nonzero latency. By adding 100 ms of round-trip time, even without any losses, FTP provided only about 20 % of overall throughput.

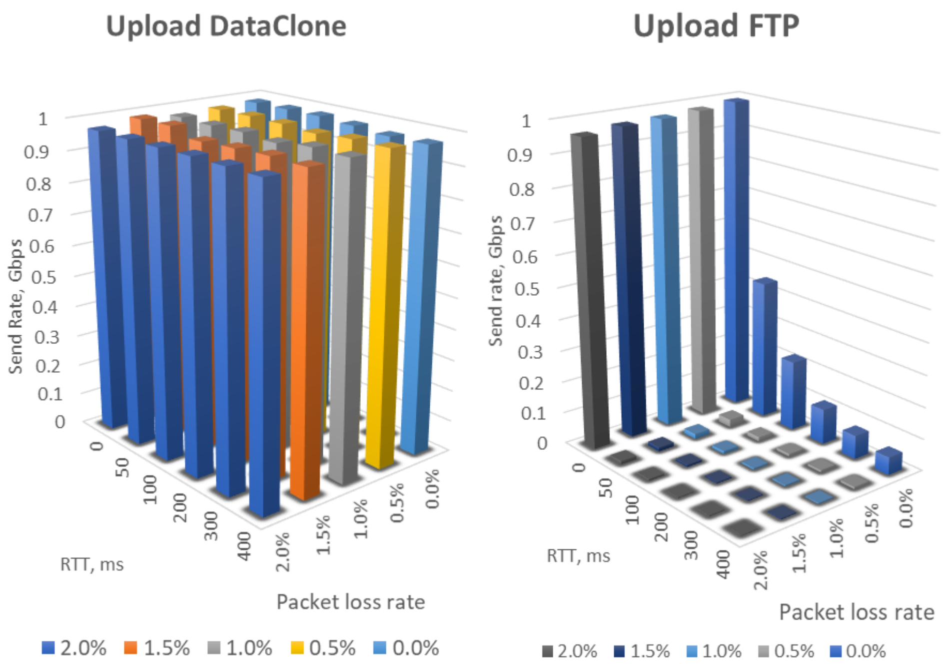

Figure 2. Performance of DataClone (RMDT) vs. FTP (TCP) on a 1 Gbps connection with a standard MTU of 1500 Bytes.

P2P on a 10 Gbps connection, with MTU 9000 Bytes

The next evaluations were performed under conditions in which the network throughput reached up to 10 Gbps with an MTU value of 9000 Bytes. The performance results of these experiments are presented in Figure 3, where the limitations of FTP can also be observed.

Even without network delays and losses, FTP provided at maximum 6 Gbps. In comparison, DataClone achieved much better results on all measured levels of packet losses, which demonstrates the effectiveness of RMDT independent of loss rate.

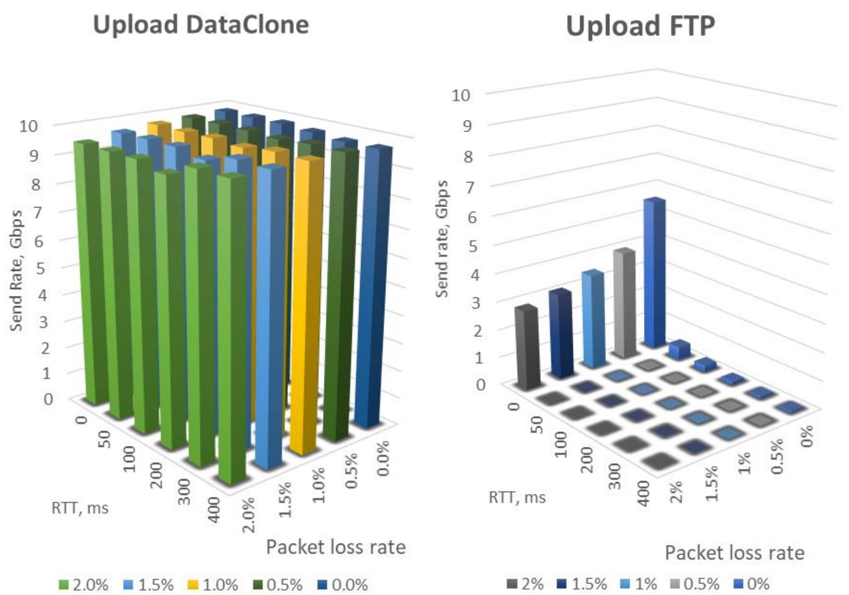

Figure 3. Performance of DataClone (RMDT) vs. FTP (TCP) on a 10 Gbps connection with an MTU size of 9000 Bytes.

Experimental setup for point-to-multipoint (P2M)

To compare the point-to-multipoint capabilities of the protocol, the DataClone application was tested against UFTP. UFTP is an IP-multicast-based reliable file transfer application and is widely used in industrial production environments today. A more detailed comparison of different systems for reliable multipoint data transmission has been outlined by Bakharev and Siemens[1]. At the time of testing, UFTP was the leading reliable point-to-multipoint protocol in terms of its capabilities. For this reason, we have tested DataClone against UFTP. It is worth noting that UFTP for that setup was configured to send data with a constant bit rate. In other words there was no congestion mechanism on board for these cases.

The network topology of the P2M experiment is shown in Figure 4. There are four nodes, three receivers and one sender, which are connected to each other via a network emulator. The maximum link capacity is 10 Gbps for the sender and 1 Gbps for each receiver. Each server is equipped with:

- CPU Intel(R) Xeon(R) CPU E5-2643 v4 @ 3.40 GHz;
- RAID array on SSD, with write/read rate exceeding 11 Gbps (1400 Mbytes per second).

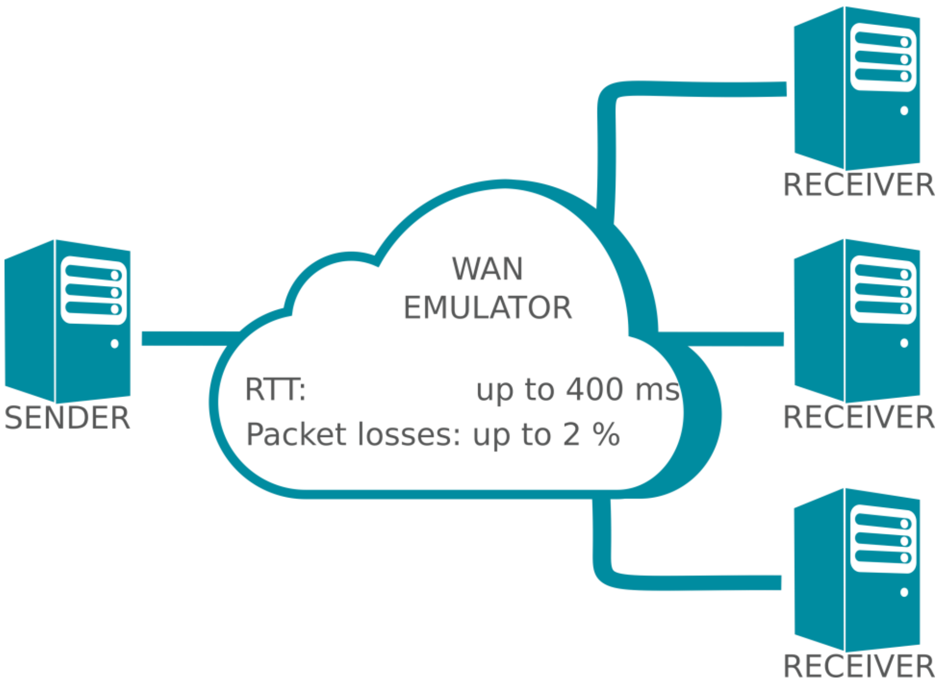

Figure 4. Topology for point-to-multipoint experiment.

[1] Contemporary Protocols for Multicast-based Reliable Data Transport and Their Features in Terms of CDN Network by A. Bakharev and E. Siemens, in "Polzunovsky vestnik" Journal 3/2, 2012, http://new.elib.altstu.ru/journals/Files/pv2012_03_02/pdf/100bakharev.pdf

P2M performance

The average data rate of all involved recipients is shown in Figure 5. With no network latencies and 2 % packet loss, UFTP performs at about 700 Mbps, which is quite satisfying. However, the data rate drops rapidly to 200 Mbps or less when network delays are increased to at least 200 ms. In general, the figure illustrates the unstable behavior of UFTP. In comparison, the unique algorithms used by RMDT within the DataClone implementation show very stable and reliable behavior, as well as a maximum utilization of the available bandwidth.

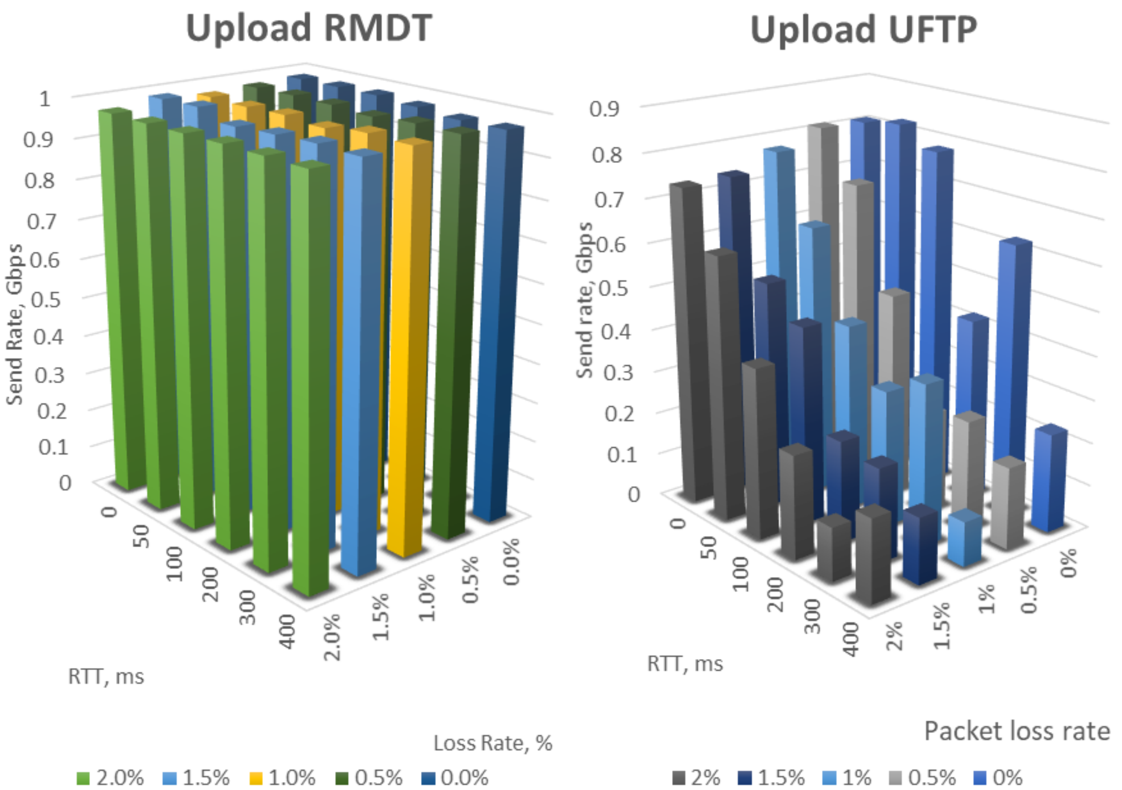

Figure 5. Comparison of average data rate of all 3 recipients of DataClone (RMDT) vs. UFTP (UDP).

How is RMDT superior to other solutions?

A number of different control algorithms have been proposed by the research community in recent years to overcome the limitations of TCP. Below we outline the key challenges and explain how RMDT solves them in comparison to other existing solutions.

- **Congestion control**: Finding an effective algorithm for congestion control is still one of the main issues of efficient high-speed data transfer. Several commercial products (e.g. from companies such as BitSpeed or XDT) still attempt to achieve higher transmission speeds by tweaking the TCP window. However, all window-based congestion controls within TCP have to be unsuccessful since all of them are based on the assumption that only packet loss can be a sign of congestion. However, control algorithms implemented for RMDT - in contrary to all TCP flavors - estimate the congestion based on the observation of achieved data rate values on the path, which provides a much more precise assessment of the path capacity.
- **Specific network pathways**: Some of the other solutions target only specific network pathways without considering the nature of the real-world Internet, ignoring dedicated IP-based private or leased lines. This leads to problems concerning the effective utilization of bandwidth, reduced overall network performance, and increased packet loss with the interference of

cross-traffic on the path. RMDT, on the other hand, offers a suitable trade-off between robustness against real-time traffic and fairness towards TCP streams.

- **Low delay transmission issues**: Some existing solutions are based on so-called bulk data transport technology (e.g. products by Aspera and File Catalyst). This technology aims to fully utilize the available bandwidth and can be used for bulk file transfer in cases when the order of delivery of the particular chunks of a file doesn't matter. However, applying such bulk transfer for sequential byte stream delivery is inappropriate, since retransmissions of lost data are performed with huge delays. Therefore, bulk data transfer is not a suitable technique for file transport scenarios that assume very low transport latency for particular data segments. In contrast, RMDT is based on a simple stream which effectively copes with the problem of packet losses and reaches multi-gigabit rates in a live stream transmission due to the inherently low delay of all the data segments of a data block.
- **Multi-point**: Finally, an efficient data exchange between multiple locations makes RMDT even more attractive. Currently, with solutions apart from RMDT, a successful one-to-many recipients distribution can only be achieved by the standard IP-multicast method. However, almost all commercial IP network providers are blocking multicast traffic due to its security vulnerabilities. Overcoming this obstacle was the initial motivation behind the development of RMDT. From the network's point of view, an RMDT multi-point session is merely a set of unicast streams, tied to a single transport session, thus avoiding IP multicast and the issues associated with it. RMDT's native support of the multi-destination mode ensures stream transmissions to multiple locations with the highest possible speed.

Other obstacles in high-speed data transmission

Even having ideal network conditions and perfect algorithms for congestion control, some challenges have to be handled on the way of high-speed data transport from your application to the receiver:

- Data acquisition: Even getting data from a fast RAID system, traditional data transmission tools use outdated so-called synchronous data read and write mechanisms. In contrast, the RMDT implementation uses highly sophisticated asynchronous read and write algorithms and thus assures the fastest data acquisition from your storage and delivery to the remote storage.
- System interface limitations of your OS: Spreading different data transport tasks over multiple CPUs in your system requires a deep understanding of the hardware and OS architecture of your system. A talk[2] presented during a recent Linuxconf conference pointed to the fact that the IP stacks of current operating systems are not able to leverage the capabilities of today's available hardware. As a basis for this observation, the speaker used the following simple calculations. The time required for sending a single data packet with the standard size of 1500 Bytes is:
    - For a 1 Gbps Interface: 15 microseconds;
    - For a 10 Gbps Interface: 1.5 microseconds;
    - For 100 Gbps: 0.15 microseconds.

  RMDT tries to optimize the interaction between the CPUs of your system, the network hardware, and the storage.

  Besides this, we recommend using so-called jumbo frames with packet sizes of up to 9000 Bytes. With jumbo frames, a 10 Gbps connection can be fully utilized by good hardware.

[2] 100G Networking Technology Overview by Christopher Lameter and Fernando Garcia, Linuxconf 2016 http://events.linuxfoundation.org/sites/events/files/slides/100G%20Networking%20Berlin.pdf

Conclusion

The technology overview provided in this document has highlighted common problems and challenges faced by Internet users in regard to high-speed data transport over large distances. Our mission is to provide the fastest and most reliable way of delivering your data. We use a set of novel algorithms and technologies which are the result of many years of research and development at the Future Internet Lab Anhalt. These technologies have been brought to a production grade and are marketed by Dexor. Several algorithms of RMDT are filed to patents. Our experiments and evaluations presented in this paper confirm that RMDT can significantly accelerate data transfer while coping effectively with packet losses and high latency.

Regardless of the needs of your business, you are likely concerned about the ability to:

- reliably distribute data over big distances via public or private Internet
- utilize up to 99 % of the available bandwidth
- transfer to up to 10 destinations with speeds of 1 Gbps to each destination
- use a point-to-point mode of up to 10 Gbps in one stream

and additionally enjoy the benefits of:

- an easy API for your own applications capable of multi-gigabit data transfers
- no special network equipment needed

Please feel free to contact us with any questions or concerns you may have regarding the advantages of faster data transmission.

Germany | phone +49 157 8234 3452 | email: info@dexor.io

To learn more about RMDT, visit dexor.io

THE INFINITE NETWORKING EXPERIENCE