# The benefits of colocating workloads on IBM Z

Colocation refers to the presentation, business logic, and data serving layers of a multitier workload that is deployed onto a single physical server.

### Key benefits

- Reductions in latency and improvements in throughput
- Improved security and availability
- Cost savings

## Does colocation matter?

Colocation can make a significant difference. Businesses and IT organizations must provide fast access to data across the web, application, and database tiers that comprise most enterprise workloads.

The physical location or proximity of the workloads can make a difference when these multitiered workloads have communication patterns that are network intensive. This means, they either frequently communicate, exchange many messages to complete a single transaction, or exchange large amounts of data. For example, big data type solutions such as analytics related workloads.

On IBM Z®, the workloads running on Linux®, IBM z/OS®, IBM z/VSE®, IBM z/TPF, or Red Hat® OpenShift® Container Platform can be collocated.

## IBM Z technologies supporting co-location

IBM HiperSockets™, Shared Memory Communication (SMC), and zdsfs are IBM Z technologies that provide benefits to customers who colocate workloads on the IBM Z platform.

HiperSockets provide a very efficient memory to memory transfer of standard packets without requiring physical networking hardware, resulting in low latency.

SMC can enable applications to replace the TCP/IP stack with an extremely thin layer, transmitting data using RDMA[1] technology with very low latencies and high throughput, while even saving on CPU cost at the same time.

zdsfs allows read access to z/OS data sets hosted on DASD[2] devices. Data can be transferred at speeds close to FICON® limits, no detour through the networking stack as required by other solutions, e.g., FTP for data exchange required.

## Benefits of colocation on IBM Z

### Reductions in latency and improvements in throughput

Workloads that access data on IBM Z can do so at memory speeds with the lowest possible latency, bypassing network traffic and delays. Network latency can affect the overall throughput capabilities for a workload.

The result of an IBM internal study demonstrates the reduced latency:

*When accessing your database while running an OLTP workload on Red Hat OpenShift Container Platform, achieve 4.2x more throughput by co-locating the workload on IBM z16™ versus running the workload on compared x86 platform connecting remotely to the IBM z16.* [3]

### Improved security and availability

External components are vulnerable to physical attacks. Because HiperSockets, SMC-D, and zdsfs do not have any physical interfaces, these technologies are protected against any attacks from the outside. And since there are no network switches, routers, adapters, or wires that can break or that have to be maintained, the availability is improved.

### Cost savings

Colocated workloads run on one server. Therefore, hardware boxes can be eliminated, and floor space and power consumption can be reduced. In addition, the central processing unit (CPU) overhead for the networking software stack is significantly reduced. With HiperSockets and SMC, there are zero external components or cables to pay for, to replace, to maintain, or to wear out. Also, the complexity that is associated with procuring and managing physical network equipment is reduced and simplified.

## Bottom line

IBM Z enables you to co-locate workloads not only to support reduced response times, operational efficiency, and improve your ability to meet service levels, but also leverages investments in existing assets.

Cloud-native applications can be located close to existing and noncontainerized workloads, empowering organizations to integrate and modernize without disrupting current services journey.

[1] *Remote Direct Memory Access*
[2] *Direct Access Storage Device*
[3] *This is an IBM internal study designed to replicate banking OLTP workload usage in the marketplace deployed on OpenShift Container Platform (OCP) 4.9 on IBM z16 using z/VM versus on compared x86 platform using KVM accessing the same PostgreSQL 12 database running in an IBM z16 LPAR. IBM z16 configuration: The PostgreSQL database ran in a LPAR with 12 dedicated IFLs, 128 GB memory, 1 TB IBM FlashSystem™ 900 storage, RHEL 7.7 (SMT mode). The Compute nodes ran on z/VM 7.2 in a LPAR with 30 dedicated IFLs, 188 GB memory, DASD storage, and OSA connection to the PostgreSQL LPAR. LPAR with 2 IFL, 4 GB memory and RHEL 8.5 with OCP Proxy server. x86 configuration: The Compute nodes ran on KVM on RHEL 8.5 on 32 Cascade Lake Intel® Xeon® Gold CPU @ 2.30 GHz with Hyperthreading turned on, 192 GB memory, RAID5 local SSD storage, and 10Gbit Ethernet connection to the PostgreSQL LPAR. Results may vary.*