HOT TOPICS

Container Growth

zCX Constraint Relief

By: Anthony Giorgio

Published: September 2nd, 2021
Read time: 5 minutes

Introduction

IBM® z/OS® Container Extensions (IBM zCX) makes it possible to integrate Linux on Z software applications as Docker Containers within z/OS without requiring a separate Linux server. Application developers can develop and data centers can operate popular open source packages, Linux applications, IBM software, and third-party software together with z/OS applications and data. The ability to colocate the Linux and z/OS parts of a workload improves operational control, leverages z/OS Qualities of Service, and typically improves performance. The overall improvement is possible because data can be more easily moved between the zCX address space hosting Linux and the address spaces hosting the z/OS parts.

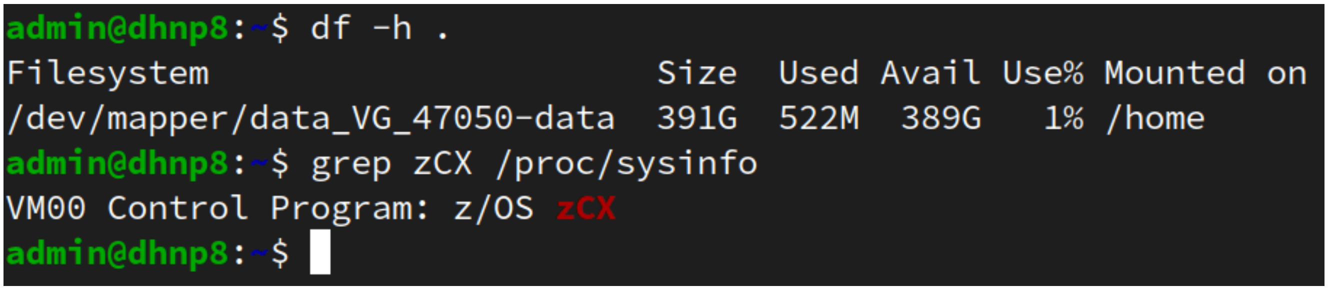
Scalability Improvements

IBM has made multiple enhancements to zCX so it can host larger workloads. The enhancements improve the scalability of the zCX address space and also grow the resources that it can access.

Larger Disks

The maximum size of a disk that zCX can use has been increased to 1 TB. The previous limit was 46 GB, so this is a significant enhancement. You can create a larger data disk using the **provision** workflow, or you can add one to an existing instance by using the **add data disks** workflow. The larger disk size gives you the flexibility to have a single large disk or a pool of multiple smaller disks.

This support is included in APAR OA60920.



```
admin@dhnp8:~$ df -h .
Filesystem                      Size  Used Avail Use% Mounted on
/dev/mapper/data_VG_47050-data  391G  522M  389G   1% /home
admin@dhnp8:~$ grep zCX /proc/sysinfo
VM00 Control Program: z/OS zCX
admin@dhnp8:~$
```

More Disks

In a zCX instance, user data disks are used to store data such as Docker containers and the contents of the home directory. The number of user data disks that you can allocate and associate with a zCX instance has been increased to 245. With the increase, you can leverage both the large disk size capability and the increased number of disks to run very large workloads in zCX. The data disks that zCX provides to the Linux guest are combined into a single logical volume.

The increased number of disks can be utilized in several ways. When the system programmer creates an instance, the zOSMF **provision** workflow will allow you to define up to 10 data disks. If the instance requires more space later, the **add data disks** workflow can be run to add up to 10 more data disks. This process can be repeated as necessary until the instance has at most 245 data disks.

This support is included in APAR OA60452.

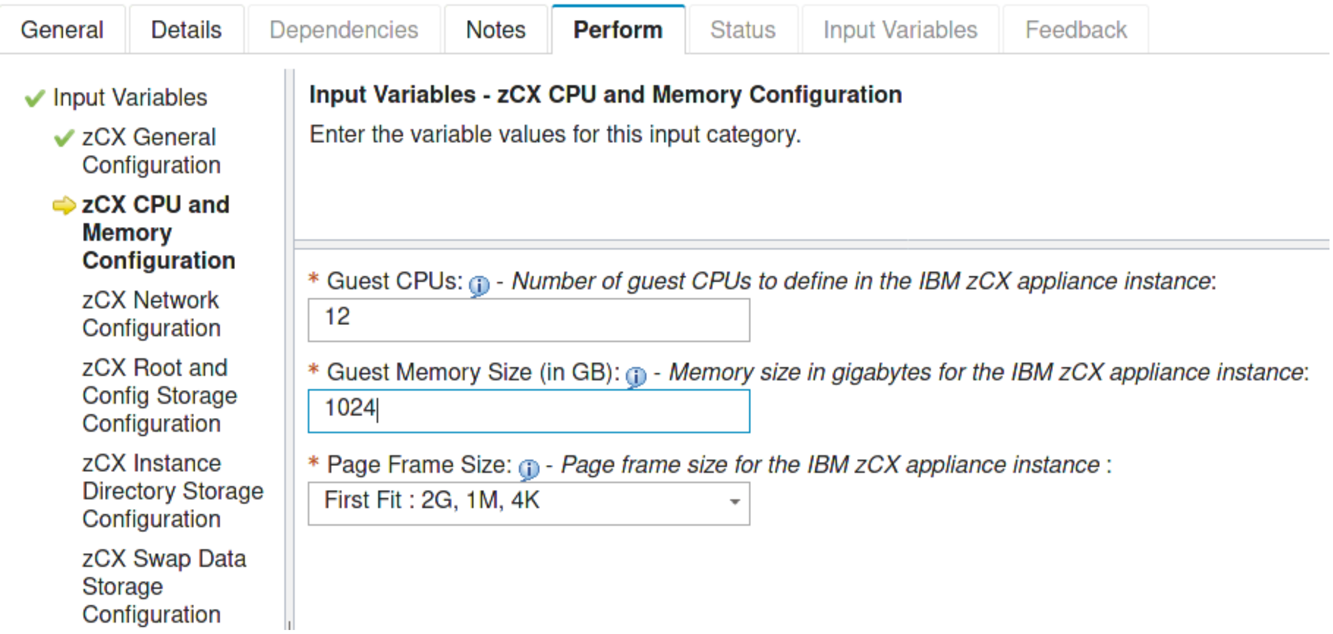
More Memory for More Containers

The maximum amount of memory that can be allocated to a zCX instance has been increased to 1 TB (1024 GB). This is a 10x increase over the previous limit of 100 GB. The memory limit for an instance is defined during provisioning and can be modified using the **reconfigure** zOSMF workflow. In order to exploit this memory, the maximum number of simultaneously running Docker containers has been raised to 1000.

Using a high memory limit enables you to run a larger number of containers or run containers that have very large memory requirements. If you assign a large amount of memory to a zCX instance, you need to ensure that the z/OS system has enough memory for normal operation. All memory assigned to zCX is page-fixed, meaning that it is unavailable for any other purpose.

If you will be using large amounts of memory in a zCX instance, you should consider configuring the instance to use 1 M (large) or 2 G (huge) pages (APAR OA59573). The larger page sizes reduce the page table overhead and can significantly improve startup and processing times.

This support is included in APARs OA59864 and OA59943.


General
Details
Dependencies
Notes
Perform
Status
Input Variables
Feedback
✓ Input Variables
✓ zCX General Configuration
zCX CPU and Memory Configuration
zCX Network Configuration
zCX Root and Config Storage Configuration
zCX Instance Directory Storage Configuration
zCX Swap Data Storage Configuration

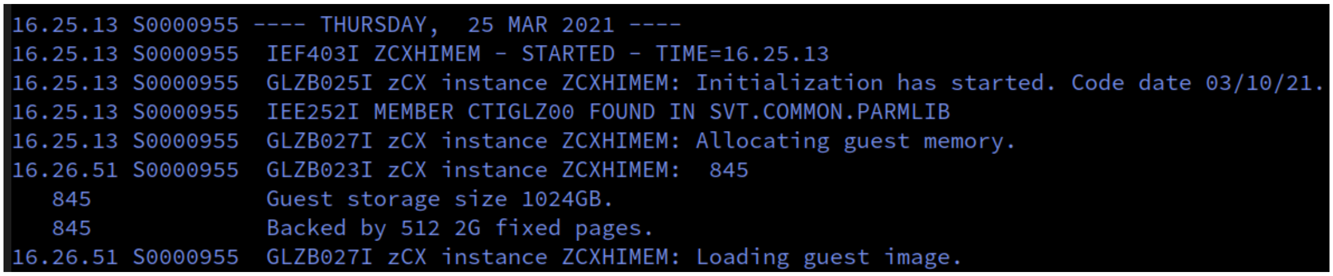
Input Variables - zCX CPU and Memory Configuration

Enter the variable values for this input category.

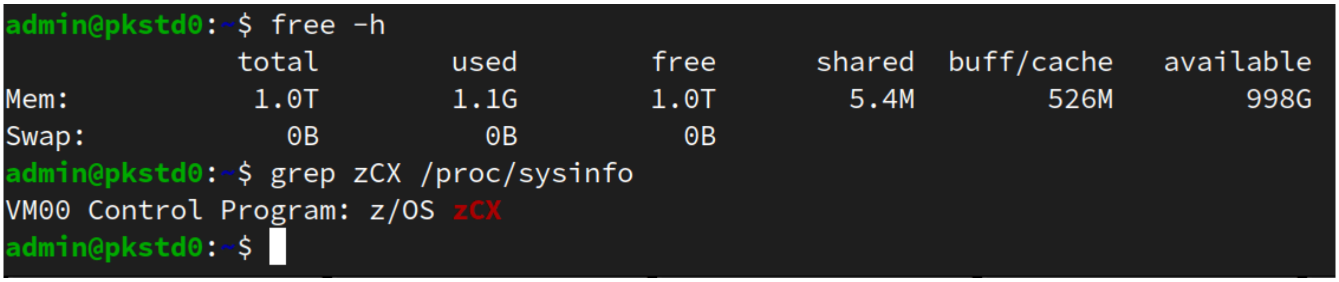
* Guest CPUs: - *Number of guest CPUs to define in the IBM zCX appliance instance*:
12

* Guest Memory Size (in GB): - *Memory size in gigabytes for the IBM zCX appliance instance*:
1024

* Page Frame Size: - *Page frame size for the IBM zCX appliance instance* :
First Fit : 2G, 1M, 4K



```
16.25.13 S0000955 ---- THURSDAY,  25 MAR 2021 ----
16.25.13 S0000955  IEF403I ZCXHIMEM - STARTED - TIME=16.25.13
16.25.13 S0000955  GLZB025I zCX instance ZCXHIMEM: Initialization has started. Code date 03/10/21.
16.25.13 S0000955  IEE252I MEMBER CTIGLZ00 FOUND IN SVT.COMMON.PARMLIB
16.25.13 S0000955  GLZB027I zCX instance ZCXHIMEM: Allocating guest memory.
16.26.51 S0000955  GLZB023I zCX instance ZCXHIMEM:  845
   845             Guest storage size 1024GB.
   845             Backed by 512 2G fixed pages.
16.26.51 S0000955  GLZB027I zCX instance ZCXHIMEM: Loading guest image.
```



```
admin@pkstd0:~$ free -h
              total        used        free      shared  buff/cache   available
Mem:           1.0T        1.1G        1.0T        5.4M        526M        998G
Swap:            0B          0B          0B
admin@pkstd0:~$ grep zCX /proc/sysinfo
VM00 Control Program: z/OS zCX
admin@pkstd0:~$
```

Usability Improvements

Consolidated Version Message

A new message has been added to zCX that displays the versions of each of its constituent components. These components are:

- The virtualization layer that hosts the embedded Linux operating system.
- The zCX appliance instance which hosts the Linux kernel and Docker engine.
- The z/OSMF workflows that perform lifecycle operations on a zCX instance.

The new consolidated version message is displayed during startup of a zCX instance as message `GLZB022I` . This message shows the version information for each of the above components. It also displays the version number of the available appliance, which helps identify whether an appliance is downlevel. You can display this message from a running zCX instance with the following console command, where < instname > is the name of the zCX instance:

MODIFY < instname >,DISPLAY,VERSION

An example message is as follows:

```
GLZB022I zCX instance ZCXANGIO version information
Bootloader:           HZDC7C0             oa59934
                      3.12.0              1.2.1

Current Appliance:    HZDC7C0             oa60923
                      3.17.0              1.16.0
                      20210217T195101Z

Available Appliance:  HZDC7C0             oa60923
```

Show more

The **Bootloader** line refers to the Linux bootloader version. When zCX service is applied, this may be updated. A simple restart of the zCX guest will pick up the new version, and it will be reflected here.

The **Current Appliance** line refers to the version of the Linux guest. This value remains constant until the appliance is upgraded to a more recent version. To upgrade an appliance version, ensure that zCX service is applied, and then run the zOSMF **upgrade** workflow. The workflow will walk you through the steps required to upgrade the appliance. When the upgrade is complete, a restart of the zCX instance will pick up the new version level. Note that the string ending in “Z” is an ISO 8601 timestamp and can be used to determine the build date of the Linux guest.

If you have a number of zCX instances running in z/OS, you can check the version numbers of each instance to determine whether they would benefit from an upgrade.

The **Available Appliance** line is similar to the previous line, but instead refers to the version of the appliance available in z/OS. When zCX service is applied to the system, this value will be updated. If this value is different than the **Current Appliance** value, it indicates that this zCX instance would benefit from an upgrade.

The **Virtualization Layer** line refers to the version level of the zCX hypervisor address space. This code is updated when zCX service is applied and is shared among all the running zCX instances on the system. To pick up the latest level of this code, you can restart any running zCX instances.

The message also indicates when the zCX instance was started, which can help to identify long-running instances for service purposes.

The **Workflows Performed** messages indicate what workflow operations (if any) were performed on this specific instance. For example, the above message shows that the provision workflow was run on March 26, 2021, by workflow version 1.0.36. No other workflows have yet been performed, as indicated by the text “N/A”.

This support is included in APARs OA58996 and OA59835.

Resource Monitoring

zCX can now monitor the resource usage of the Linux guest within an instance. This monitoring covers the following areas:

- Memory
- Swap disk
- Root disk
- Data disk

If the utilization of one of those areas crosses a monitoring threshold, a message is printed to the job JESMSGLG. It identifies the affected resource and the level of urgency. A message is only issued when a threshold is crossed. If a resource continually remains at a given level, only a single message is printed.

These messages are intended to alert the z/OS system programmer that a particular zCX instance is suffering from resource starvation. If any of the resources are exhausted, the instance could unexpectedly terminate.

A set of resource alert messages is as follows:

```
GLZM011I zCX instance ZCXANGI1 alert: memory warning.

Utilization at 80%.

GLZM011I zCX instance ZCXANGI1 alert: memory critical.

Utilization at 99%.

GLZM011I zCX instance ZCXANGI1 alert: memory normal.
```

Show more

In addition, a message is printed to the job SYSPRINT that shows the utilization levels for each resource:

```
Mar 29 15:24:01 azd_resmon: crit    MEM: 99 SWAP: 61 ROOT:  7 DATA: 65
```

To resolve a constrained resource situation, first identify the resource and then consult the following table:

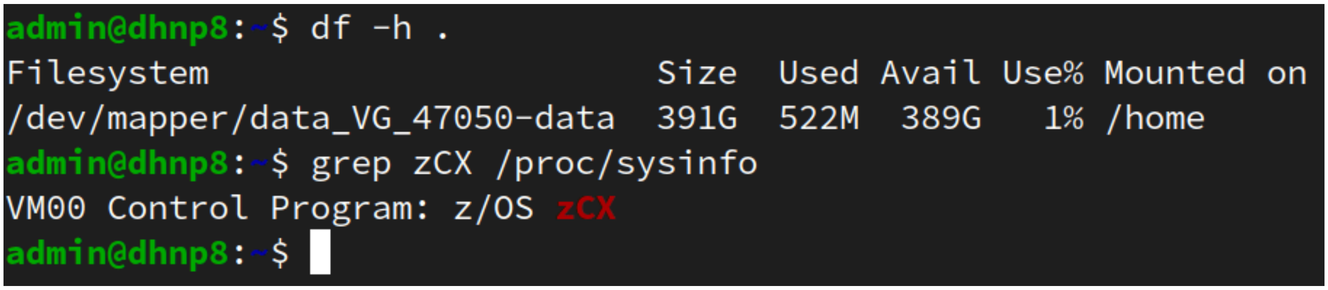
Resource	Possible Actions
Memory	• Reduce the number of running containers. • Add more memory to the zCX instance with the zOSMF **reconfigure** workflow.
Swap Disk	• Reduce the number of running containers. • Add more memory to the zCX instance with the zOSMF **reconfigure** workflow. • Add one or more swap disks to the zCX instance with the zOSMF **add data disks** workflow.
Root Disk	• Increase the size of the root disk with the zOSMF **upgrade** workflow.
Data Disk	• Add one or more data disks to the zCX instance with the zOSMF **add data disks** workflow.

This support is included in APAR OA60303.

IPv6 Support

zCX can now use an IPv6 address in addition to an IPv4 address for network communications. This capability enables the Linux guest to communicate over an IPv6 network as needed. An IPv4 address is still required.

To utilize this feature, set the IPv6 address in the zOSMF **provision** or **reconfigure** workflows. Make sure to modify the appropriate DVIPA settings in the z/OS TCPIP profile.



```
admin@dhnp8:~$ df -h .
Filesystem                      Size  Used Avail Use% Mounted on
/dev/mapper/data_VG_47050-data  391G  522M  389G   1% /home
admin@dhnp8:~$ grep zCX /proc/sysinfo
VM00 Control Program: z/OS zCX
admin@dhnp8:~$ 
```

This support is included in APAR OA59508.

About the author

Anthony Giorgio is an Advisory Software Engineer with 20 years of mainframe experience. He currently works on the z/OS Container Extensions (zCX) team in Poughkeepsie, NY. Anthony holds a BS and MS in Computer Science from the NYU Tandon School of Engineering.

Debbie Noll contributed to the editorial review of this article.