

# ASR9K - eXR ‘不显示的show memory summary’所有物理存储器

## Contents

[Introduction](#)

[问题详细资料分析](#)

## Introduction

在聚合服务运行改进的XR (eXR)的路由器9000 (ASR9K)，亦称64位XR，版本6.2.1，并且更新命令“show memory summary”不显示在对应的数据页描述的所有内存，也许迷惑一些用户。

## 问题详细资料

在我们之下将使用ASR9901，运行6.4.2，为例。

根据[数据表](#)集成路由处理器(RP)有32十亿字节(GB)随机访问存储器(RAM)。

```
RP/0/RSP0/CPU0:R1#show platform
```

Node	Type	State	Config state
0/RSP0/CPU0	ASR9901-RP(Active)	IOS XR RUN	NSHUT
0/FT0	ASR-9901-FAN	OPERATIONAL	NSHUT
0/FT1	ASR-9901-FAN	OPERATIONAL	NSHUT
0/FT2	ASR-9901-FAN	OPERATIONAL	NSHUT
0/0/CPU0	ASR9901-LC	IOS XR RUN	NSHUT

“Show memory summary”在27GB附近只显示，尽管认可总内存的系统作为32GB在“admin show system资源下”。

```
RP/0/RSP0/CPU0:R1#show memory summary
```

```
node:      node0_RSP0_CPU0
```

```
-----  
Physical Memory: 27089M total (22185M available)  
Application Memory : 27089M (22013M available)  
Image: 4M (bootram: 0M)  
Reserved: 0M, IOMem: 0M, flashfsys: 0M  
Total shared window: 133M
```

```
RP/0/RSP0/CPU0:R1#admin show system resources
```

Node	Physical	Application	Boot				
Partition	CPUs	Shmwin					
	Total	Available	Cached	Total	Available	Cached	Total
Available							
0/RSP0-Host	32415M^	512M^	135M	31655M^	500M^	132	923M
542M	4	N/A					
0/RSP0-Admin	1940M	1072M	144M	1894M	1047M	141	2308M
1252M	1	N/A					
0/RSP0-XR	27739M	22548M	999M	27089M	22020M	975	N/A

N/A 2 6655M

<snip>

例如同一个工作情况在其他ASR9K设备可以被观察，：与eXR 6.2.3的ASR9010  
查看[数据表](#)A9K-RSP880-TR应该有RAM 16GB。

```
RP/0/RSP0/CPU0:R1#admin show system resources
```

Node Partition	Physical			Application			Boot
	CPUs	Shmwin		Total	Available	Cached	Total
Available	Total	Available	Cached	Total	Available	Cached	Total
0/RSP0-Host	32415M^	512M^	135M	31655M^	500M^	132	923M
542M	4	N/A					
0/RSP0-Admin	1940M	1072M	144M	1894M	1047M	141	2308M
1252M	1	N/A					
0/RSP0-XR	27739M	22548M	999M	27089M	22020M	975	N/A
N/A	2	6655M					

<snip>

“Show memory summary”显示12.5GB和“admin show system”资源显示16GB

```
RP/0/RSP0/CPU0:R2#show memory summary
node: node0_RSP0_CPU0
-----
Physical Memory: 12496M total (8465M available)
Application Memory : 12496M (8287M available)
Image: 4M (bootram: 0M)
Reserved: 0M, IOMem: 0M, flashfsys: 0M
Total shared window: 128M
```

```
RP/0/RSP0/CPU0:R2#admin show system resou
```

Node Partition	Physical			Application			Boot
	CPUs	Shmwin		Total	Available	Cached	Total
Available	Total	Available	Cached	Total	Available	Cached	Total
0/RSP0-Host	16217M^	276M^	29M	15837M^	271M^	28	923M
592M	8	N/A					
0/RSP0-Admin	1940M	1158M	78M	1894M	1131M	76	2308M
1451M	1	N/A					
0/RSP0-XR	12796M	8488M	888M	12496M	8288M	867	N/A
N/A	6	3071M					
0/1-Host	24491M	11510M	127M	23917M	11241M	124	923M
592M	6	N/A					
0/1-Admin	1008M	471M	90M	984M	460M	88	2308M
1528M	1	N/A					
0/1-XR	10948M	5902M	2343M	10691M	5764M	2288	N/A
N/A	5	3071M					

然而，如果我们检查32位XR，它显示总物理存储器作为documentated在[数据表](#)：

```
RP/0/RSP0/CPU0:R3#show memory summary
Physical Memory: 16384M total (12600M available)
Application Memory : 16043M (12600M available)
Image: 100M (bootram: 100M)
Reserved: 224M, IOMem: 0, flashfsys: 0
Total shared window: 51M
```

```
RP/0/RSP0/CPU0:R3#show platform
```

Node	Type	State	Config State
-----			

0/RSP0/CPU0	A99-RSP-TR(Active)	IOS XR RUN	PWR,NSHUT,MON
0/RSP1/CPU0	A99-RSP-TR(Standby)	IOS XR RUN	PWR,NSHUT,MON
0/1/CPU0	A9K-MOD400-SE	IOS XR RUN	PWR,NSHUT,MON
0/1/0	A9K-MPA-20X10GE	OK	PWR,NSHUT,MON
0/1/1	A9K-MPA-2X100GE	OK	PWR,NSHUT,MON
0/2/CPU0	A99-8X100GE-SE	IOS XR RUN	PWR,NSHUT,MON
0/4/CPU0	A9K-MOD400-TR	IOS XR RUN	PWR,NSHUT,MON
0/4/0	A9K-MPA-8X10GE	OK	PWR,NSHUT,MON
0/4/1	A9K-MPA-2X40GE	OK	PWR,NSHUT,MON

## 分析

从版本6.2.1向前，在ASR9K的eXR将使用虚拟机虚拟化型号，转换从使用Linux用于版本6.1.2和6.1.3的容器(LXC)型号。

容器比虚拟机能更大量包装应用程序到单个物理服务器。

VM能占去很多系统资源。每个VM运行所有硬件的操作系统，但是虚拟复制的不仅充分的复制操作系统(OS)需要运行。这可能添加很多内存和CPU周期。相反，容器要求的所有是运行一个特定程序的足够一个操作系统，支持计划和库和系统资源。然而，VM产生flexibility适应功能类似In-Service Software Upgrade (ISSU)。

因为Linux容器共享主机的物理资源和内核，而每个VM要求其自己的OS和虚拟化的硬件，这将解释每个VM为什么从导致不显示的show memory summary eXR版本6.2.1的整体总物理存储器及以后的整体主机分配特定量的资源。Linux Conainers能也有分配的内存的限制，当出现在NCS5000和NCS5500平台，并且‘show memory summary’ CLI的同样behavios在这些平台将被观察。