

31-bit address space private area

- The portion of the user's private area in each virtual address space that is available to the user's problem programs is called the *user region*.
- The use of address spaces enables z/OS to maintain the distinction between the programs and data belonging to each address space.
- The private areas in one user's address space are isolated from the private areas in other address spaces, and this address space isolation provides much of the OS's security.

Private areas

- There are two private areas: below the 16 MB line is the private (PVT), and above the 16 MB line is the extended private (EPVT).
- Their size is the complement of the common area's size.
- The virtual addresses in the private area is unique to the programs running in such areas.
- The private area is formed by the following areas:
 - Subpools 229, 230, and 249
 - This area enables private storage to be obtained in the requestor's storage protect key†.
 - The area is used for control blocks that can be obtained only by authorized programs (such as z/OS) having appropriate storage protect keys.
 - A subpool is a virtual storage area with the same properties regarding storage key, pageable or fixed, private or common, fetch protected or not, and so on.

Local system queue area (LSQA)

- This area contains tables, control blocks queues and content supervision associated with the address space, subpool queue elements (SQEs), descriptor queue elements (DQEs), and free queue elements (FQEs).
- It contains content management control blocks that define tasks and programs within the elements (CDEs).
- Note: LSQA is difficult to calculate because it depends on the number of loaded programs and the size of other subpools in the address space.
- LSQA is intermixed with SWA and subpools 229, 230, and 249 downward from the bottom of the CSA into the unallocated portion of the private area, as needed.
- ELSQA is also intermixed, but it is allocated downward from 2 GB into the unallocated portion of the extended private area, as needed. LSQA does not take space below the top of the highest storage currently allocated to the user region.

Scheduler work area (SWA)

- SWA contains control blocks that exist from task initiation to task termination.
- It includes control blocks and tables created during job control language (JCL) interpretation.
- Anything that appears in the jobstream for each STEP will create some sort of control block in this storage area.
- Generally, this area will increase with the increase in the number of DD statements.
- SWA usually is assigned subpools 236-237 and storage allocation depends on the release of the operating system.

Subpool 229

- Primarily used for staging of messages. JES uses this area for messages to be printed to the system log (i.e. JCL messages).
- This area is also used for SYSIN/SYSOUT buffers.

Subpool 230

- Used by VTAM for in-bound message assembly for segmented messages.
- RACF uses this area for RACFLIST and data management such as data extent blocks (DEBs) for open datasets. NOTE: This area will increase as more datasets are opened.

Subpool 249

- Used by TCP/IP cache queues for event control blocks.

A 16 KB system region area

- This area is reserved for GETMAINS by all system functions (for example, ERPs) running under the region control tasks. It comprises 16K (except in the master scheduler address space, in which it has a 200K maximum) of each private area immediately above the PSA.
- The V=V region space allocated to user jobs is allocated upwards from the top of this area.
- This area is pageable and exists for the life of each address space. (V=V- see right column)

User region

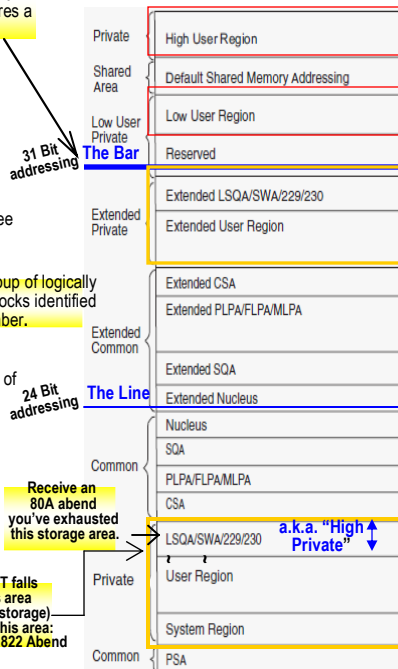
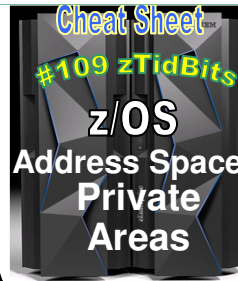
- This region is used for running user program applications (loaded at subpools 251/252) and storing user program data (subpools from 0 - 127).
- When a module is loaded into the private area for an address space, the region available for other components is reduced by the amount of storage used for the module.
- The amount of private virtual storage that a job can use for subpools 251/252 and from 0 - 127 (the low addresses of the two private areas) can be limited through the REGION keyword on the JOB or EXEC JCL statements.
- The region size can be controlled and overridden through the SMF exit IEFUSI.
- A value equal to 0 KB or 0 MB gives the job all private storage available.

† Information in central storage is protected from unauthorized use by means of multiple storage protect keys. A control field in storage called a key is associated with each 4K frame of central storage. If the key associated with the request does not match the storage key, the system rejects the request and issues a program exception interruption..

Each address space has its own unique private area allocation.

2**50 - 2**64 The high user region non-shared area starts at 2**50 or wherever the shared area ends and goes to 2**64.

2**31 to 2**32 From 2 GB - 4 GB is considered the bar. Below the bar can be addressed with a 31-bit address. Above the bar requires a 64-bit address.



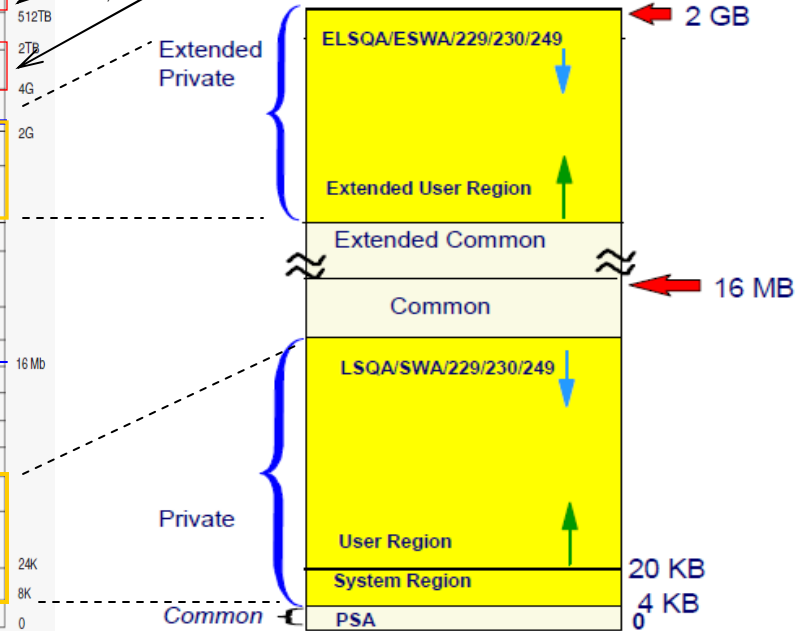
User Region addressing category

- There are two variety of user regions: virtual (or V=V) and real (or V=R).
- Virtual and real regions are mutually exclusive; private areas can be assigned to V=R or V=V, but not to both.
- The installation determines the region to which jobs are assigned.
- Usually, V=R should be assigned to regions containing jobs that cannot run in the V=V environment, or that are not readily adaptable to it.
- Programs that require a one-to-one mapping from virtual to central storage, such as program control interruption (PCI) driven channel programs, are candidates for real regions.

Two significant differences between virtual and real regions are:

- How they affect an installation's central storage requirements
 - How their virtual storage addresses relate to their central storage addresses.
- For **virtual regions**, which are pageable and swappable, the system allocates only as many central storage frames as are needed to store the paged-in portion of the job (plus its LSQA). The processor translates the virtual addresses of programs running in virtual regions to locate their central storage equivalent.

Programs obtain storage above the bar in "chunks" of virtual storage called **memory objects**. The system allocates a memory object as a number of virtual segments; each segment is a megabyte in size and begins on a megabyte boundary. A memory object can be as large as the memory limits set by your installation and as small as one megabyte (IARV64, IARCP64 and IARST64 services).



What is region size?: The region size is the amount of storage in the user region available to the job, started task, or TSO/E user. The system uses region size to determine the amount of storage that can be allocated to a job or step when a request is made using the STORAGE or GETMAIN macros and a variable length is requested. The region size minus the amount of storage currently allocated, determines the maximum amount of storage that can be allocated to a job by any single variable-length GETMAIN request.

What is region limit?: The region limit is the maximum total storage that can be allocated to a job by any combination of requests for additional storage using the GETMAIN or STORAGE macros. It is, in effect, a second limit on the size of the user's private area, imposed when the region size is exceeded.

Specifying region size: Users can specify a job's region size by coding the REGION parameter on the JOB or EXEC statement. The system rounds all region sizes to a 4K multiple. The region size value should be less than the region limit value to protect against programs that issue variable length GETMAINS with very large maximums, and then do not immediately free part of that space or free such a small amount that a later GETMAIN (possibly issued by a system service) causes the job to fail.

- For **real regions**, which are nonpageable and nonswappable, the system allocates and fixes as many central storage frames as are needed to contain the entire user region.
- The virtual addresses for real regions map one-for-one with central storage addresses.
- Virtual regions:** Virtual regions begin at the top of the System Region (see Figures) and are allocated upward through the user region to the bottom of the area containing the LSQA, SWA, and the user key area (subpools 229, 230, and 249).
- Virtual regions are allocated above 16 megabytes also, beginning at the top of the extended CSA, and upward to the bottom of the extended LSQA, SWA, and the user key area (subpools 229, 230, and 249).
- For **V=V jobs**, the region size can be as large as the entire private area, minus the size of LSQA/SWA/user key area (subpools 229, 230, and 249) and the System Region.
- For **V=R jobs**, the REGION parameter value cannot be greater than the value of the REAL system parameter specified at IPL.
- Note: If the user *does not* explicitly specify a V=R job's region size in the job's JCL, the system uses the VRREGN system parameter value in the IEASYS00 member of SYS1.PARMLIB.
- For V=R requests, if contiguous storage of at least the size of the REGION parameter or the system default is not available in virtual or central storage, a request for a V=R region is placed on a wait queue until space becomes available.
- Note: VRREGN should not be confused with the REAL system parameter. REAL specifies the total amount of central storage that is to be reserved for running all active V=R regions. VRREGN specifies the default subset of that total space that is required for an individual job that does not have a region size specified in its JCL.