

The System Data Infrastructure

This chapter describes how Com-plete system data are organized. It covers the following topics:

- Introduction
 - Sharing Data Among Multiple Com-plete Nuclei
 - The System Data Access Method (SDAM) API
 - The SDAM Control Block
 - SDAM Views
-

Introduction

Com-plete and user profile data are stored on System Data Containers (SDCs) that are VSAM (KSDS) clusters accessed through the System Data Access Method (SDAM) API. Com-plete currently uses the following four SDCs:

Internal Data Set ID	VSAM DDname	Record Size	Nr. of Records	Contents
BASE	COMSYS1	4089	58	Vital Com-plete data
USERDEF	COMSYS3	2041	150	User profiles (User ID, menu,..)
CATALOG	COMSYS4	313	80	Com-plete program / file catalog

Notes:

1. The Nr. of Records is the approximate number of records added to the SDCs during the installation process and may differ from SM to SM.
2. The SDCs, especially USERDEF, can be heavily accessed data sets, depending on the number of users logged on and the type of work they carry out. You are therefore recommended to monitor access against those data sets and place them on a low-contention volume / disk pack to guarantee optimum performance.
3. The space required for the USERDEF SDC is dependent mainly on the number of User IDs defined. As this tends to be the least "static" SDC, its VSAM allocation status should be reviewed with IDCAMS LISTCAT even more often than for the other SDCs. When the report shows many extents, the primary allocation should be increased. With a high number of CI/CA splits, reorganization of the file should be considered as soon as possible.

SDC data records are structured as follows:

Position 1 to 16	The key, consisting of
	-Physical Record ID (binary halfword)
	-System ID (one byte alphanumeric) with a value of x'00' (global record) or the local ID (patch character).
	-Object ID (twelve bytes alphanumeric), e.g. the User ID, message id, ..)
	-Subrecord ID (one byte binary) serves as a sequence number, should there be more than one record for an object ID be required.
Position 17 to 57	Various SDAM control information
Position 58 to nn	where nn represents the maximum record size depending on type of data and the Data Container it is kept on. Only this portion of the record is transferred from/to the requestor of a SDAM function.

Sharing Data Among Multiple Complete Nuclei

The SDCs can be shared among two or more Com-pletes. However, the VSAM shareoptions must be changed from the installation default value of (2 3) to (3 3) to allow concurrent updates from all Com-pletes.

The System ID, described in the table above, controls the individual data record's scope. The patch character (set by sysparm PATCHAR) defines the local system ID for every Com-plete nucleus. Switch Applymod 81 on to force Com-plete to look for local data ahead of global records. With Applymod 81 off, local data will be ignored. Note that having Applymod 81 switched on will approximately double access rates to the SDCs, as Com-plete will try to find a local copy of every record read before considering the global one.

This allows users and the system administrator to choose between common environment definitions for all Com-pletes and different profile definitions on one or more systems. Using shared data allows installations with the need to operate more than one copy of Com-plete to greatly reduce administration effort as well as disk space, and reduces data redundancy.

When opting to share the data set(s) please take the following into consideration when choosing the VSAM shareoption:

- Shareoption 3 induces additional VSAM overhead to the system compared to SHR(2), but on the other hand does not fully guarantee data integrity. This should only be used when maintenance is applied to the SDC(s) from a central location (that is, a specific Com-plete).
- Shareoption 4 guarantees full data integrity on the cost of even more VSAM overhead. Nevertheless, this should preferably be used to be 100 % proof. Additionally, note that VSAM will NOT allocate additional extents for a component defined with SHR(4) when one fills. You are strongly encouraged to reserve a number of spare blocks to accommodate unavoidable additions to the SDC(s), again especially the USERDEF dataset.

The System Data Access Method (SDAM) API

The calling sequence is:

```
SDAM (SACB, i/o_area)
```

The SACB (SDAM Control Block) is mapped by CMSACB (BAL), COBSACB (COBOL), and PLISACB (PL/I).

The SDAM Control Block

Field	Usage and possible value(s)
SACRCOD	Return code. See the Messages and Codes documentation for SDAMnnnn messages, where nnnn is the SDAM return code.
SACFDBK	Feedback information. When non-zero, describes the error (SACRCOD) further (that is, VSAM returncode/feedback, Adabas return code, ..).
SACMESS	Message as seen in the Messages and Codes documentation.
SACWORK	Work area for SOFTWARE AG internal use
SACSYST	The System ID (patch character). Low-value means global, otherwise local to the system specified here.
SACOBID	Object ID (that is, item name). This actually describes the specific item within the group (SACSYST / SACVIEW), for example, the message ID, User ID, ...
SACSUBI	Sub-identification. For items spanning more than one record or groups of records for the same object (SACOBID), this specifies the actual item within the group of items.
SACVIEW	A View name that is required to access data through the SDAM API. More than one logical record can be held within one physical record. This allows related data to be kept independently but tightly coupled. Every view can be manipulated independent of other data held in the same physical record, but nevertheless results in performance gains when, for example, more than one view is accessed in a program due to the fact that the underlying physical record has to be read only once and the problem program receives just the logical part it requested. However, a view name can be assigned to entire records as well. A list of available views can be found at the end of this chapter.

Field	Usage and possible value(s)
SACOPER	Operation Code.
	OPEN Set up the environment, allocate workareas etc.
	CLOSE Clean up the environment, free workareas etc.
	READ Retrieve a record
	WRITE Store a new record
	UPDATE Replace an existing record
	DELETE Remove an item from COMSYS
	POINT Specify starting point for sequential retrieval
ENDREQ Terminate sequential processing	
SACFLG1	Processing Option(s), an array of 8 elements.
	SYSOK Retrieve data for SACSYST only, ignore applymod 81 / 82.
	NEWCOPY When the current call (either logical or physical) results in access to the same physical record as manipulated with the last call, the request will normally be satisfied from internal buffers and will not result in a physical read to the storage media. NEWCOPY will force a physical read.
FULLKEY For sequential operations with a non-zero SACGENL specifies that positioning is to be performed using the full key and not the partial key as specified by SACGENL.	
SACTSTM	When reading a record this will contain the record's time stamp. This field is input only.
SACARLN	Length of the area specified as the second input parameter to SDAM, the record area.
SACOCCR	For tabular items specifies the index of the specific item to be accessed (for example, a message record contains the text for various languages, the language is the index into this array).
SACGENL	For sequential retrieval specifies the generic key length. When the (part of) the key specified via SACVIEW, SACOBID, etc, does not / no longer matches the key of a record retrieved by the access method, logical EOF will be signalled.

SDAM Views

The Complete distribution source library contains macros describing the record layout for the various records accessible through the SDAM API. The macro names are SDAVxxxx, where xxxx represents the hexadecimal physical record ID behind the VIEW definitions. The following table gives an overview of the views currently available:

View Name	Physical ID	Purpose
USERID	x0080	Complete user definition record
USERPROF	x0080	Complete user profile record
USERLAST	x0080	Complete user's last defaults for UQ, UEDIT, ...
CUSTDATA	xF000	For your use to store site-specific data