# The evolution of CICS/ESA in the sysplex environment

by T. Banks
K. E. Davies
C. Moxey

***The IBM CICS/ESA® transaction processing subsystem has been enhanced in a way that enables transparent exploitation of Parallel Sysplex™ technology, even by existing large-scale applications. This paper describes how CICS/ESA takes advantage of and integrates new Multiple Virtual Storage (MVS) features to provide an external view of the sysplex as a single entity with workload-sensitive routing algorithms for passing work requests between nodes of the sysplex. Sysplex-wide access is provided to application scratch-pad data, to security data, and to database and file data with complete integrity.***

The IBM Customer Information Control System/Enterprise Systems Architecture product (CICS/ESA*) has evolved over the last 25 years to become a core component of many of IBM's customers' business applications. In the last few years it has undergone the significant metamorphosis from its foundation as a single task within a Multiple Virtual Storage (MVS) system to a structure that consists of many components cooperating within an MVS system complex, or sysplex.

The services provided by CICS* are those of a transaction processing (TP) monitor[1] that receives and processes multiple concurrent requests from a network of client terminals or systems. The resultant updates to shared data are processed in a way that provides a simple programming environment, yet safeguards the integrity of the data. In respect of this concurrent activity, the workload falls obviously into a pattern that allows multiple processors to be applied to the individual requests, and advantage can be taken from the lower cost of hardware in an MVS sysplex configuration.

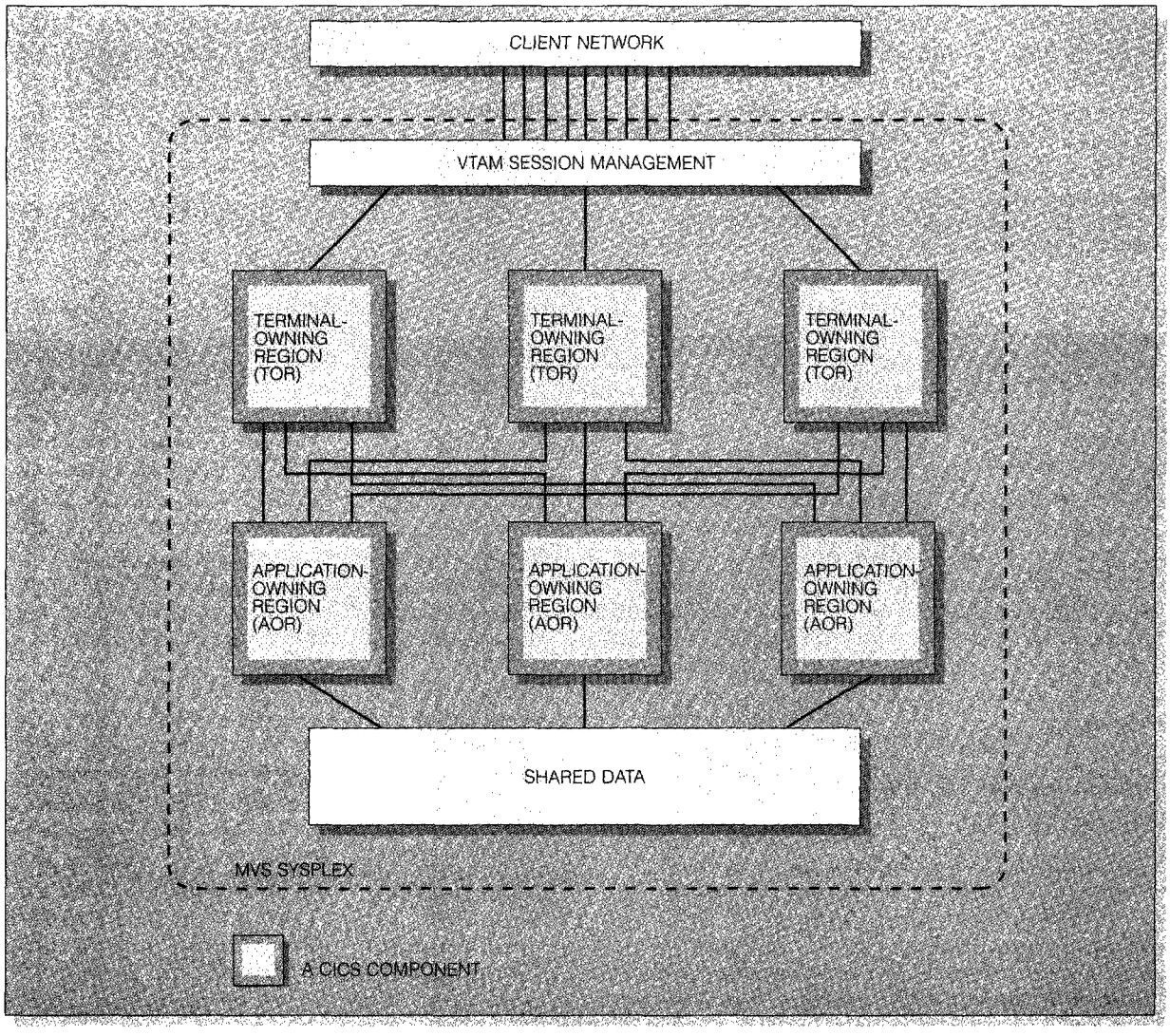
There are, however, points at which the processing of a request requires access to data or services that are shared or coordinated throughout the sysplex. The ability to access shared application databases is the most obvious, and there are also logical requirements to share control data for system functions, such as directing requests from the network to particular regions. Additional requirements for sysplex-wide services stem from the need for high performance. For example, the accessibility of data from any processor makes the allocation of processors more flexible and can enable more efficient use of them. In the case of security functions, the results of work on one processor are made available to others, which saves repeating the work.

The succeeding sections of this paper explain some of the services provided by MVS and how they have been integrated in CICS/ESA to produce a highly scalable transaction processing environment. Figure 1 shows the major components of a CICS/ESA configuration within a sysplex. Each CICS component (shaded) in the figure can potentially execute on a separate MVS image. Each MVS image must also con-




0018-8670/97/$5.00 © 1997 IBM

**Figure 1 Replicated CICS regions within an MVS sysplex**




tain VTAM* (Virtual Telecommunications Access Method) and shared-data components that access data shared throughout the sysplex. Requests arrive from the network of clients, which may be end-user workstations, terminals, or systems acting as concentrators or local servers. The role of the VTAM component is to act as an interface between the sysplex and the network. It provides transport and routing functions[2] that direct session initiation requests to one of the terminal-owning regions (TORs). Once established, a session typically persists for many transaction requests, each of which has a lifetime of a few tenths of a second. A complete interaction with an end user of the system involves a sequence of transaction requests that are related to one another by data that persist in the sysplex. These data may consist of updates to databases, the end result of the sequence of requests, and of intermediate "scratchpad" status, which is discarded once an interaction is complete.

The role of the TOR is to manage sessions with the terminals in the network and direct each transaction request to an application-owning region (AOR) that

can execute the transaction. The relationship between the request and the AOR is determined by the available capacity in the AORs and by scratch-pad data, which are in a form that is accessible from one particular AOR and not others. A further role of the TOR is to manage presentation of the transaction results for nonintelligent "IBM 3270-like" terminals. The role of the AOR is to execute the application logic in response to the initial request. The AOR must, therefore, have access to all of the resources necessary to the application. The application may depend on the recoverability of some of these resources. Recovery information is contained in a log that is managed by the AOR. In CICS, these resources consist of databases, files, and queues with various functional and performance characteristics that are needed for output to printers or asynchronous processes initiated by the application.

The databases that are accessed by CICS applications include the hierarchical DL/I,[3] and relational DB2* (DATABASE 2*).[4] CICS customers also place data in files managed by the Virtual Storage Access Method (VSAM)[5] to exploit different performance characteristics. In many cases the data are managed solely as files, with CICS and application logic fulfilling the functions of managing data relationships and recovery, which are traditionally associated with a database manager.

## The clients' view of the sysplex

Internally, the sysplex consists of replicated components enabling the use of lower-cost hardware and redundancy that can be exploited in the case of failure of one of the copies. Externally, the clients of the sysplex can be presented with a consolidated view, if the clients identify the sysplex using a name representing any of the members of the group of CICS TORs that can accept the received session request. This group is called a VTAM *generic resource*.

As session requests are received from clients, they are associated with a member of the generic resource based on an assessment of workload and available capacity. The session is bound with that member in the normal way. In the case where the session carries data that have no significance for data integrity, a failed session may be reestablished through a different MVS image in the sysplex, and the client may continue new work. In some cases, however, sessions are used to carry recoverable data to and from clients that are transactional systems. If a failure occurs in this case, the session must be reconnected to the same CICS TOR to complete the protocol exchanges and restore the integrity of the data.

The relationships between sessions and the CICS TORs are called VTAM *affinities* and are maintained in a global sysplex coupling facility. In this way the affinities are preserved over any failure of an MVS image, and can be used to reestablish the relationships that correctly preserve the integrity of any data updated via the session.

## Communications within the sysplex

Fast communications are necessary to enable CICS/ESA regions to cooperate in processing a request. CICS/ESA has for many years provided a communications facility called *multiregion operation* (MRO) that enables efficient communications within an MVS image. In the examples used so far, the communication function has been used to pass requests for transaction execution from the point of arrival in the TOR to an appropriate AOR. This facility can also be used to invoke execution of a program, or a single CICS command (such as READ FILE) in a different CICS region. Facilities are also provided to conduct conversations using a SEND/RECEIVE interface.

Recently MRO has been enhanced to use the cross-system coupling facility (XCF[6]), which is part of the Multiple Virtual Storage/Enterprise Systems Architecture (MVS/ESA*) base control program. XCF provides high-performance communication links between MVS images that are linked in a sysplex by channel-to-channel links, Enterprise Systems Connection (ESCON*) channels, or coupling facility links. The selection of XCF to communicate across MVS boundaries is automatic and transparent to application programs, and supports all of the uses of MRO.

## Workload-sensitive routing

The efficient use of processor and storage resources within the sysplex depends on directing each request from the TOR, where it is first examined, to any of the (identical) AORs best able to execute it. The choice of AOR must be made according to any relationship the request has to data within an AOR (see a later section, "Reducing the affinity of requests to CICS regions") and according to workload based on an algorithm whose sophistication depends on the amount of information available.

The simplest technique is a *round-robin* approach, where only the existence of the AOR is taken into

account. A more advanced scheme, which is sensitive to the variations in load on individual processors, is the *shortest queue*, where the least-loaded AOR is selected based on the number of incomplete requests. Most sophisticated is a *goal-oriented* approach that involves interaction with the MVS workload manager.[7] This component of MVS controls allocation of processor and storage to the AORs and TORs as a result of:

- Gathering real-time data from the subsystems that reflect performance at an individual request level
- Monitoring MVS- and subsystem-level delays and waits that are contributing to overall request execution times
- Dynamically managing the resources of the sysplex, using the performance goals and the real-time performance and delay data as inputs to system resource management algorithms

In the goal-oriented mode, the workload manager works together with IBM's CICSPlex* SM[8] (system manager), the latter taking control of the routing of transactions from TOR to AOR based on response time data.

### Access to shared data

Access to shared updateable data is the chief characteristic of transactional systems. The data are made visible to application programs in a way that logically isolates requests from one another. This is done by making the data accessible at a very granular level and locking the data during the execution of a transaction to prevent update by any concurrent request for the same data.

In order that the data remain in a consistent state, even following a failure of software or hardware, logging facilities are provided that can restore data to their original state at the start of a *unit of work*.[9] CICS/ESA provides access to database managers via a generic connection facility called the resource manager interface that allows connection to multiple databases such as DL/I and DB2, and provides for the coordination of updates to these databases via CICS logging.

A significant alternative to the traditional database for CICS customers is the use of VSAM data sets. VSAM provides access to simply structured data such as *key-sequenced* or *entry-sequenced* records. Locking and logging have been provided by CICS to enable the use of VSAM data in transactional requests. The DL/I and DB2 database managers have been enhanced to provide locking facilities that span the multiple MVS images in the sysplex, which makes the data accessible from any AOR that is selected to process a request. Locking for VSAM data has similarly been converted to a sysplex scope, though logging for updates has been maintained within the CICS AOR. The next section develops the details of these changes.

### Sharing data with VSAM record-level sharing

CICS provides sysplex-wide sharing of VSAM data by use of the record-level sharing (RLS) function of the IBM Data Facility Storage Management System (DFSMS*) version 1 release 3.

CICS has for many years provided sharing of VSAM data via *function shipping*, in which all accesses to a data set are funneled through a single CICS region, the *file-owning region*. Sysplex sharing allows the CICS application-owning regions (AORs) to access the data set directly, which eliminates the need for a file-owning region, and avoids the problems of its becoming a potential bottleneck or being a single point of failure. Using RLS, if a CICS region or an MVS image fails, work can be dynamically routed to other regions in the sysplex, and still access the same data.

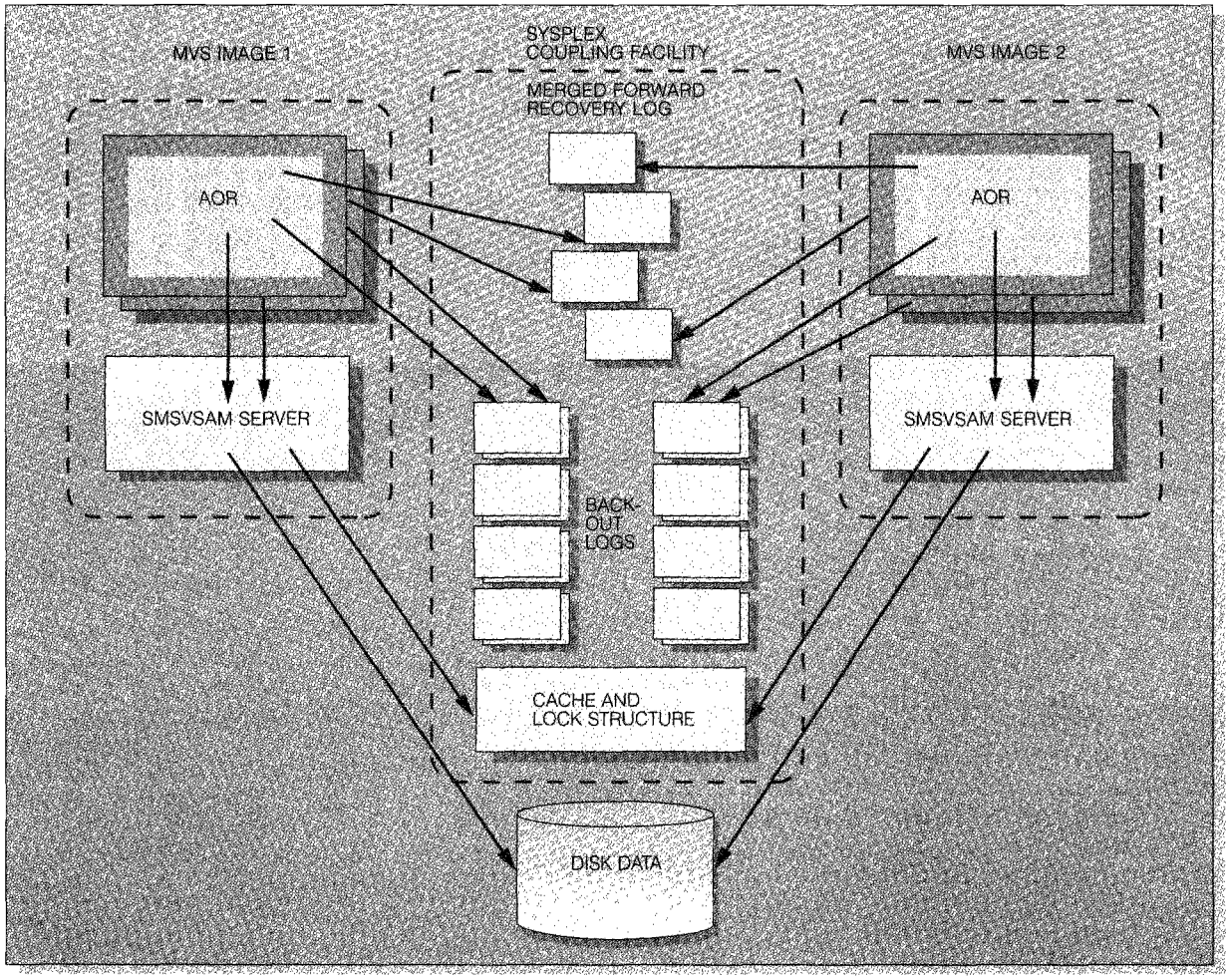
There is no need for any changes to the VSAM data in order to exploit record-level sharing, because RLS is a mode of access rather than an inherent property of the data set. CICS can access a given VSAM sphere (the base cluster data set plus any associated data sets, such as alternate index data sets) in either RLS or non-RLS mode. If any region has accessed any part of the sphere in one or another of these two modes, then access from all regions and to all parts of the sphere must be in the same mode.

CICS automatically registers with the storage management subsystem for the VSAM (SMSVSAM) server during initialization. The SMSVSAM server resides in a separate address space that processes RLS requests; there is one server per MVS image in the sysplex. VSAM RLS exploits a number of sysplex hardware and software features in order to provide data sharing. These facilities enable:

- Maintenance of cache structures and a lock structure in the coupling facility
- A buffer cross-invalidation mechanism

Buffer cross-validation is used to ensure that each SMSVSAM server knows when data it holds in local

**Figure 2 Major components of CICS with record-level sharing (RLS)**




buffers have been updated by any system within the sysplex, and that it must therefore get a refreshed copy from the cache structure. Figure 2 shows the major components of the CICS record-level sharing structure.

Since all users of an RLS-mode data set must be in agreement about its properties in terms of recoverability, the recovery attributes must be stored in a central location, and the VSAM user catalog is used for this purpose. These recovery attributes are:

- Whether or not the data set is recoverable, meaning that updates made to the data set within a unit of work must either all be committed or all be backed out atomically.
- Whether or not the data set is forward recoverable, meaning that the data set can be reconstructed by applying forward recovery log records to an earlier copy of the data set; for example, in the event of a media failure.
- The name of the MVS log stream, if the data set is forward recoverable, to which forward recovery log records are to be written.

If a data set is recoverable, then CICS performs backout logging so that the updates can be backed out if a failure occurs. These backout log records are writ-

ten to a CICS system log, of which there is logically one per CICS region. If a data set is forward recoverable then CICS also writes forward recovery log records to the forward recovery log stream. The MVS logger[10] ensures that the log records written by multiple CICS regions to the single forward recovery log stream for the data set are merged into the correct sequence. This merging is vital for sharing of forward recoverable data sets.

In order to allow shared access to a data set from multiple CICS regions in a sysplex, the locking is performed centrally by VSAM, using a single lock structure for the sysplex. VSAM RLS also provides a new *retained* state for locks, which facilitates the preservation of data integrity after transaction and system failures. For example, if a CICS region fails while a unit of work holds locks, then in order to preserve data integrity, the locks should not be released until the CICS region has restarted and backed out the unit of work. However, other regions trying to update the locked records will not want to wait for the locks to be released, so the locks are converted by VSAM into retained locks. When a request is made against a record held by a retained lock, the application immediately gets back an error response rather than having to wait.

Retained locks are also used in other situations where a failure means that the records will remain locked longer than usual. The most important of these is an *in-doubt failure*. If a portion of a distributed unit of work running in a particular CICS region reaches a synchronization point, or sync point, and connection to the coordinator is lost between the prepare and commit phases of two-phase commit, then the unit of work has suffered an in-doubt failure. The use of retained locks allows those records that the unit of work updated to remain locked until the link can be resynchronized and the in-doubt failure can be resolved.

The locking provided by RLS allows CICS to provide three levels of read integrity. The default is to read without integrity; this allows the reader to see data that have not been committed and could therefore be subsequently backed out. This method also has optimal performance since no read locking is involved. In addition to this, CICS exploits RLS to provide *consistent read integrity*, in which data will only be seen by the reader after the read has been committed, and *repeatable read integrity*, in which data that have been read within a unit of work cannot be updated until the unit of work has completed, so the reader can see the same data repeatedly. Repeatable read integrity allows an application that reads several records to ensure that the first record will remain unchanged when the last one is read.

Using RLS access mode, nonrecoverable data can be fully shared between any number of CICS regions and any number of batch jobs. Recoverable data can be read in batch processing mode while being read or updated by interactive CICS, but cannot be accessed for update by batch. Therefore, in order to update recoverable data from batch, the data set must be opened in non-RLS mode. This requires that the data set not be open in RLS mode from any CICS region, and RLS provides a *quiesce* function that facilitates the closing of the file from all CICS regions that have it open.

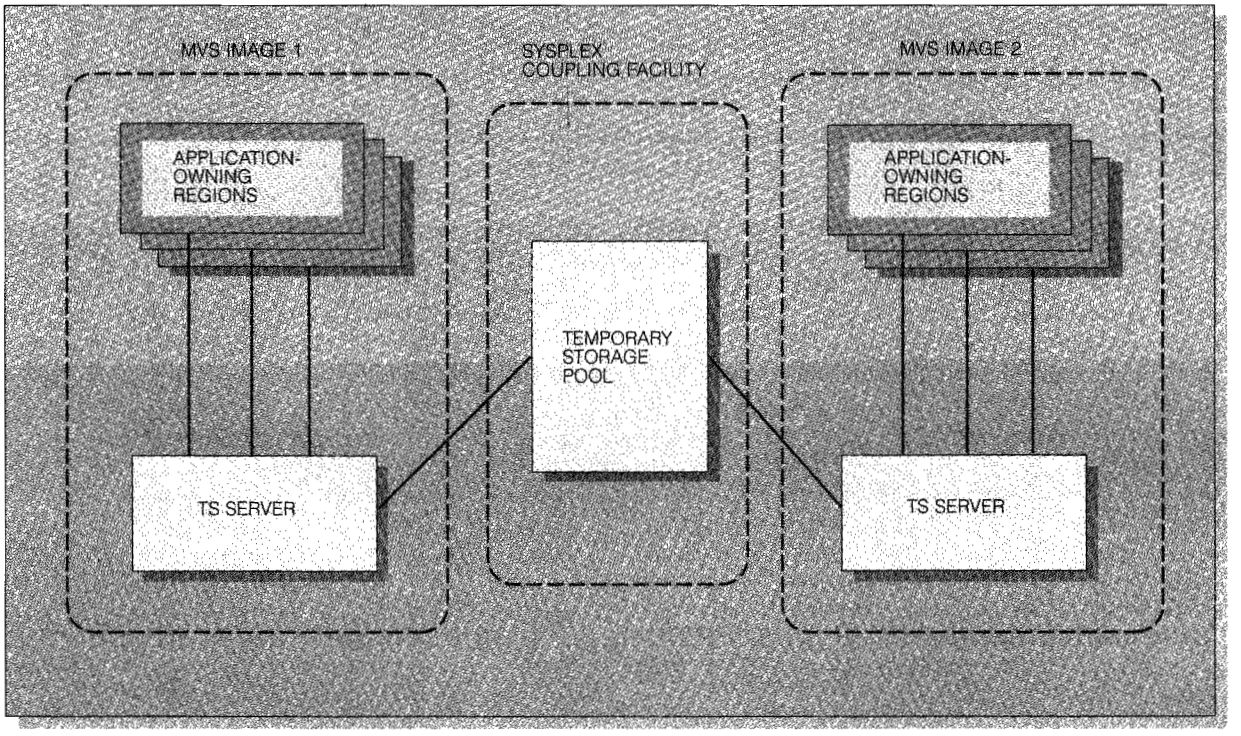
The quiesce function is implemented in a way that allows a request to quiesce a data set, issued from one CICS region, to be automatically propagated to all the CICS regions in the sysplex that have the data set open. The responses from the CICS regions that indicate that they have processed the request are also coordinated. This propagation and coordination mechanism also turns out to have uses for situations other than that of quiescing RLS access to a data set. For example, this mechanism is used to notify CICS regions that a backup copy of the data set is about to be performed, and to coordinate the responses from each CICS so that the copy will not start until all regions are ready for it to do so.

Record-level sharing introduces some new failure scenarios from which CICS must recover. These arise from the fact that (1) the SMSVSAM server is a separate address space from the CICS address space, which can therefore fail separately, and (2) the use of cache and lock structures in the coupling facility can either fail or suffer a loss of connectivity. The CICS design objective is to handle these failures and recover from them without the need for user intervention (other than to cure the underlying problem, such as a failed connection) and without loss of data integrity.

### Reducing the affinity of requests to CICS regions

As requests arrive in the TORs, their relationship to AORs must be established based on the existence of any scratch-pad data created by previous requests relevant to the new request. Scratch-pad data in CICS can take several forms—such as unformatted data

**Figure 3 CICS temporary storage (TS) data sharing**




in a main storage area or in a facility called *temporary storage* (TS) that is managed by CICS via a command interface.

In a nonsysplex configuration of CICS/ESA, transactions have generally been routed consistently from a TOR to a single AOR and the accessibility of scratch-pad data from outside that AOR has not been an issue. In the sysplex environment the advantages of multiple AORs—the balancing of workload between AORs and the existence of redundant copies to cope with failure—can be fully exploited only if the data are available from any AOR.

The temporary storage facility of CICS/ESA has been enhanced to provide a means of efficiently sharing scratch-pad data between regions. Figure 3 shows the conceptual view in which each MVS image has a CICS temporary storage server. The AORs access the TS server by means of cross-memory functions within MVS, and the TS servers access the shareable data within the coupling facility.

These shared data can be used transparently by applications that are programmed using the TS interface. This reduces the management work required to define the relationships between transactions executed as part of a sequence of requests.

The TS data-sharing facility also allows the TS data to be managed by division into pools that might represent test and production data, and can reside on different coupling facilities. Each pool is accessed using its own server on each MVS system, and the AORs can access several pools concurrently. Although TS data-sharing queues are not recoverable, they are normally preserved across a CICS region restart, or an MVS reinitialization (re-IPL), providing the coupling facility is not stopped and does not fail.

## Sharing security information within the sysplex

This section describes the use of sysplex facilities to optimize the performance and usability of CICS security functions. These functions are used to identify a user within a system and grant (or refuse) that user authority to access resources that make up an application (such as programs, queues, databases, and files), based on resource profiles. As CICS dis-

tributes its applications among the many regions within the sysplex, it also needs to distribute the security for those applications.

**Sharing of user access data across the sysplex.** The security characteristics of a user are associated with the terminal session and are established by a sign-on process at the start of the session. In a TOR the security characteristics for a transaction request are naturally inherited from the terminal session that begins the request, but as transactions initiated at that terminal are distributed to an AOR for execution, that user's security sign-on must be distributed also. The security characteristics are used to construct a description of the user that can be quickly referenced each time the user requests access to a resource. The description is known as the accessor control environment element (ACEE).[11,12]

An AOR may (depending on installation setup) receive sign-on information with each transaction request passed to it from the TOR. The information can consist of *userid* and *password*, which allows the AOR to reverify the user's identity, but it is usually sufficient to trust the verification performed by the TOR and use the userid alone to check resource access using the same ACEE mechanism used in the TOR.

In the sysplex environment the work of creating the ACEE in the AOR is reduced by the MVS Resource Access Control Facility (RACF*); lookaside information is saved in the coupling facility, and the scope of reuse for the information is therefore that of the sysplex rather than that of the CICS region or MVS image.

**Management of shared security resource profiles.** In RACF, the secured resources are represented by *resource profiles* containing *access lists* that describe the access authorities of users to those resources. The resource profiles are maintained in a RACF database on a direct access storage device. For performance reasons, CICS has always required the resource profiles be available in main storage during the lifetime of CICS and, consequently, if the master copy of the profiles was updated, a rebuild of the in-storage copies of the information was required. This has been managed within the scope of a CICS region and has necessitated operational procedures and suspension of CICS resource access during the rebuild (a period of many minutes in some installations).

These problems have been solved by the introduction of the *global* RACLIST feature.[13] When this feature is used, it allows the resource profiles to be loaded into a RACF-owned *data space*[14] instead of the private storage of individual CICS regions and used directly from there. This leads to an immediate saving in virtual storage, but a consequential benefit is that the profiles can be managed directly by RACF across the scope of the sysplex. RACF is aware of which profiles have changed, so it can selectively reload the changes. The reloading is performed as a phase-in process, which avoids the need to suspend access during the reload. The management of the rebuilding of data spaces following database changes is handled via XCF[6] sysplex communications facilities and requires no additional operational synchronization.

## Sysplex management

The advantage of tools to manage a sysplex configuration is clear even in a small configuration such as that suggested by Figure 1. In fact, large systems may consist of as many as 50 CICS regions, and at this scale the use of management tools becomes a necessary part of the facilities of the sysplex. CICSPlex SM[8] has been developed along with CICS/ESA to provide consolidated views of, and controls for, the resources and relationships that make up the complete system.

CICSPlex SM operates by collecting data from each CICS region and consolidating those data within a controlling address space using topology definitions for the sysplex. This consolidation allows the various components that make up an application to be monitored as a whole, and controlling commands to CICSPlex SM can be transformed into commands that are issued to the individual components automatically. CICSPlex SM has also been provided with interfaces to control routing of requests from TORs to AORs based on workload distribution within the sysplex, and on affinities between transactions, as suggested in the prior section on workload-sensitive routing.

## Summary

The design points for the sysplex structure are competitive price/performance, scalability, and continuous availability. These attributes are provided to a transaction workload by replicating and connecting software components that exploit low cost, replicated hardware, and which can redirect work dynamically based on processor capacity or hardware and soft-

ware component failure. Scalability is provided in two complimentary strategies. First, by dividing the workload wherever necessary to avoid bottlenecks while providing serialization appropriate to shared data access, and second, by the provision of automation tools that can manage the complex system that results from the division of the workload.

The facilities described have been developed over the period of several recent CICS/ESA releases. They provide an application platform that integrates the MVS sysplex facilities to provide a view of the complex processor and storage resources as a consolidated platform for transaction processing that is accessible to a large inventory of existing applications.

*Trademark or registered trademark of International Business Machines Corporation.

## Cited references and notes


1. J. Gray and A. Reuter, *Transaction Processing: Concepts and Techniques*, Morgan Kaufmann Publishers, San Mateo, CA (1993).
2. R. J. Cypser, *Communications for Cooperating Systems: OSI, SNA, and TCP/IP*, Addison-Wesley Publishing Co., Reading, MA (1991).
3. *IMS/ESA Version 5 General Information*, GC26-3467, IBM Corporation (1995); available through IBM branch offices.
4. *IBM DATABASE 2 Version 3 General Information*, GC26-4886, IBM Corporation (1994); available through IBM branch offices.
5. *DFSMS/MVS Version 1 Release 3 General Information*, GC26-4900, IBM Corporation (1995); available through IBM branch offices.
6. *MVS/ESA Programming: Sysplex Services Guide*, GC28-1495, IBM Corporation (1995); available through IBM branch offices.
7. *MVS/ESA Programming: Workload Management Services*, GC28-1494, IBM Corporation (1995); available through IBM branch offices.
8. *IBM CICSPlex System Manager for MVS/ESA: Concepts and Planning Release 2*, GC33-0786, IBM Corporation (1995); available through IBM branch offices.
9. A unit of work is a sequence of processing actions that must be completed before any of the individual actions can be regarded as committed. A unit of work is completed when a transaction takes a sync point (a point of synchronization), which occurs either when a transaction issues an explicit sync point request, or when CICS takes an implicit sync point at the end of the transaction. After changes are committed, they become durable, and are not backed out in the event of a subsequent failure of the transaction or system.
10. *OS/390 MVS Programming: Assembler Services Guide*, GC28-1762, IBM Corporation (1996); available through IBM branch offices.
11. The ACEE is created by an external security manager (ESM). The ESM can be any security product that conforms to the MVS programming interface[12] known as the System Authorization Facility (SAF). Since this paper explicitly describes the exploitation of the Parallel Sysplex capabilities of the Resource Access Control Facility (RACF), it will assume that the ESM is indeed RACF.
12. *External Security Interface (RACROUTE) Macro Reference for MVS*, GC23-3733, IBM Corporation (1995); available through IBM branch offices.
13. *RACF V2.1 Presentation Guide*, GG24-4281, IBM Corporation (1994); available through IBM branch offices.
14. *OS/390 MVS Programming: Extended Addressability Guide*, GC28-1769, IBM Corporation (1996); available through IBM branch offices.


*Accepted for publication December 19, 1996.*

**Timothy Banks** *IBM UK Laboratories Ltd., Hursley Park, Winchester, Hampshire S021 2JN, United Kingdom (electronic mail: tbanks@vnet.ibm.com).* Mr. Banks joined IBM in 1984 after a period of using computational techniques in the engineering analysis of mechanical structures and fluids. His first assignment in CICS was in system testing, concentrating on communication and availability features. He then spent two years supporting large systems in the Installation Support Center in London. After a short time in management he moved to CICS/ESA development where he has worked on the two-phase commit protocols used by CICS and on APPC support generally. He received an M.A. in mathematics from Cambridge University and has recently gained an M.Sc. in software engineering from Oxford University, where his main interest was in the use of mathematically based notation and analytical tools to assist in the design of concurrent systems.

**Ken E. Davies** *IBM UK Laboratories Ltd., Hursley Park, Winchester, Hampshire S021 2JN, United Kingdom (electronic mail: kedavies@uk.ibm.com).* Dr. Davies is a Senior Technical Staff Member currently working as a technical planner on CICS/ESA. He has worked on CICS for 21 years, joining the development team to design and implement the command level interface. He was chief designer of the multiregion architecture, transaction routing and resource definition on-line function. Dr. Davies has had three assignments outside of CICS. He has worked on high-level language compiler design, MVS communication architecture, and as a system availability consultant in the United States. More recently he has had responsibility for the architecture of CICS/ESA exploitation of Parallel Sysplex and is the CICS representative on the S/390® Software Design Council. Dr. Davies received his D. Phil. at the University of Sussex, specializing in nuclear spectroscopy.

**Catherine Moxey** *IBM UK Laboratories Ltd., Hursley Park, Winchester, Hampshire S021 2JN, United Kingdom (electronic mail: cmoxey@vnet.ibm.com).* Mrs. Moxey is a project programmer working on the CICS transaction processing product at IBM Hursley. She joined IBM in 1989, having previously worked for Digital Equipment Corporation and Software Sciences. She has worked on a number of data management projects for CICS on MVS/ESA, including shared-data tables, and most recently on the CICS support for VSAM record-level sharing. She is currently involved in further exploitation of the Parallel Sysplex environment. Mrs. Moxey holds an M.A. in chemistry from Oxford University. Her interests include unit testing and the interface between humans and computer systems, in particular the documentation of a complex software product such as CICS.

Reprint Order No. G321-5647.