



Software Group | Enterprise Networking and Transformation Solutions (ENTS)

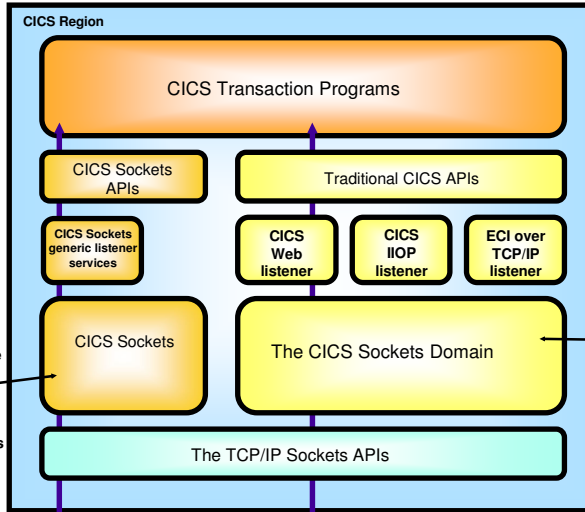
# CS z/OS CICS Sockets Background



## Direct TCP/IP communication into the CICS region - an overview

A CICS Sockets transaction has direct access to the TCP/IP socket and can issue native sockets calls to receive and send data over the socket.

These services are based on the Sockets Extended sockets APIs (provided by Communications Server)



A CICS Sockets Domain transaction does not have direct access to the socket, but communicates with CICS Sockets Domain services to receive a request and to send a reply over a socket.

The listeners are the 'servers' as seen from a TCP/IP perspective.

These services are based on the UNIX System Services C/C++ sockets API (provided by Language Environment) and the UNIX System Services callable APIs

A conversational model - or a request/reply model

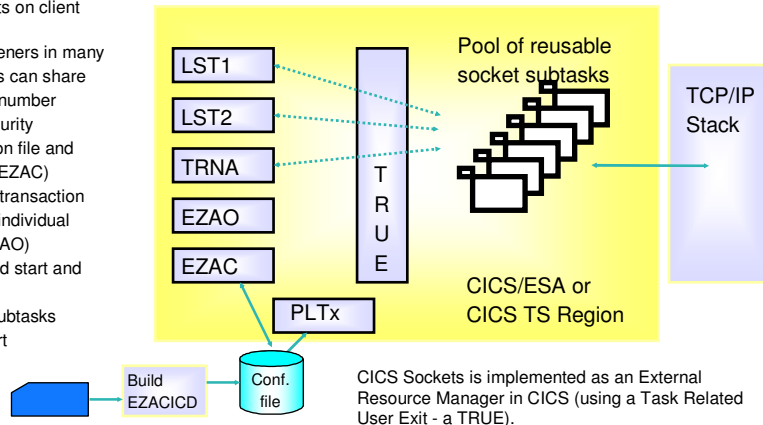
A request/reply model

## CICS Sockets overview

- Multiple listeners - each instance separately configurable
- Enhanced listener has no requirements on client input data
- Multiple listeners in many CICS regions can share listener port number
- User ID security
- Configuration file and transaction (EZAC)
- Operations transaction to start/stop individual listeners (EZAO)
- PLT-enabled start and termination
- Reusable subtasks
- IPv6 support

CICS Sockets is a component of the Communications Server for z/OS, not CICS TS itself.

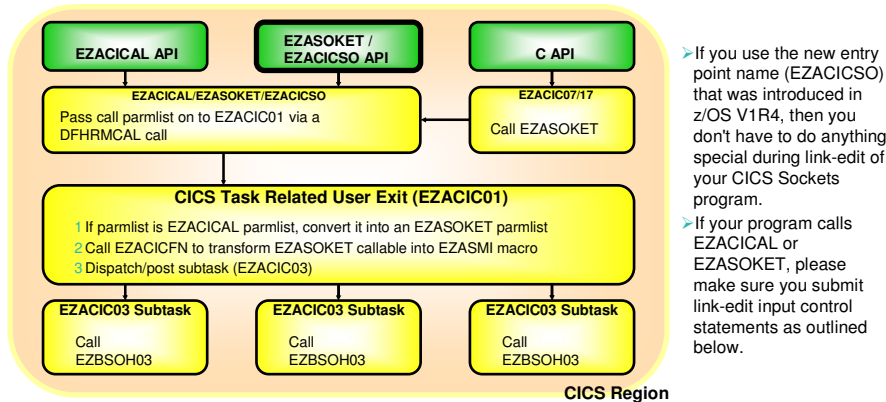
It is a general-purpose sockets programming API to be used by CICS application programmers for implementing native (low-level) sockets communication in z/OS CICS transaction programs.



CICS Sockets is implemented as an External Resource Manager in CICS (using a Task Related User Exit - a TRUE).

Functional enhancements: configuration file with info on multiple listeners per CICS image. Each listener can be controlled via port number, queue length, ASCII/EBCDIC translation, security exit, timeouts. Security exit may use CICS user ID security instead of terminal-related security (requires CICS V4 and write own EZACICSE security exit routine, that returns the user ID to start the transaction under). Configuration file may be built by batch utility and/or maintained with the EZAC transaction. Start/Stop can be done via CICS PLT processing - incl. enabling TRUE and starting listener transactions. Individual listeners can be stopped/started via the EZAO transaction. Special gethostbyname() module (EZACIC25) should be used by CICS

## CICS Sockets APIs



- If you use the new entry point name (EZACICSO) that was introduced in z/OS V1R4, then you don't have to do anything special during link-edit of your CICS Sockets program.
- If your program calls EZACICAL or EZASOKET, please make sure you submit link-edit input control statements as outlined below.

PQ28963 ships re-entrant version of EZACIC07, called EZACIC17.

<p><b>CICS C-Socket Program</b> Linkage Edit control:</p> <pre> //SYSLIN DD * INCLUDE SYSLIB(EZACIC07) NAME MYCPGM(R) *</pre>	<p>CICS Call EZACICAL and Call EZASOKET program Linkage Edit control:</p> <pre> //SYSLIN DD * INCLUDE SYSLIB(EZACICAL) NAME MYSOKPGM(R) *</pre>
---	---



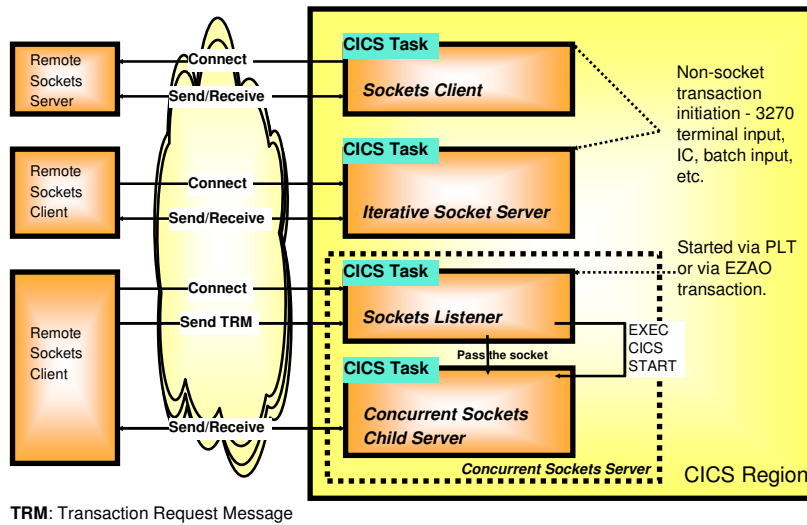
**Remember this, Please!!**

Both CICS C-sockets and Call EZACICAL socket programs are transformed into calls to the sockets extended callable API before the socket calls are passed down to the socket communicating subtasks, making the full CICS socket implementation much more streamlined. The subtasks now only have to do call routing on behalf of the CICS task.

Really, EZACICAL calls are transformed directly into EZASMI macro calls by EZACIC01, there's not a transform to EZASOKET first. (According to Bill Kelsey, Oct 2001).

A CICS task may use sockets extended callable sockets, including assembler callable sockets; but not the sockets extended

## CICS Sockets program categories in CICS





# Trademarks, Copyrights and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM logo	Cloudscape	Informix	OS/90	WebSphere
e/logo/business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
 IBM Corporation  
 North Castle Drive  
 Armonk, NY 10504-1785  
 U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2005. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.