

IS04 - New CICS Connectivity Options



Ingo Franzki



Trademarks

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

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business (logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

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 trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Notice Regarding Specialty Engines (e.g., zIIPs, zAAPs and IFLs):

- Any information contained in this document regarding Specialty Engines ("SEs") and SE eligible workloads provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (e.g., zIIPs, zAAPs, and IFLs). IBM authorizes customers to use IBM SE only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the "Authorized Use Table for IBM Machines" provided at http://www.ibm.com/systems/support/machine_warranties/machine_code/aut.html ("AUT").
- No other workload processing is authorized for execution on an SE.
- IBM offers SEs at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT.

Agenda

- **CICS Overview**
- **CICS TS V2.1 – z/VSE 6.1**
 - Channels and Containers
- **CICS TS V2.2 – z/VSE 6.2**
 - HTTP 1.1 Support
 - CICS Web Support with OpenSSL
- **CICS Connectivity options**
 - CICS Explorer
 - CICS Web Support
 - CICS Transaction Gateway
 - z/VSE SOAP Engine
 - z/VSE REST Engine
- **EZA Multiplexer & EZA OpenSSL Support**



CICS TS for VSE History

- **CICS TS for VSE/ESA 1.1.0 - released June 1999**
 - VSE ported up to 100 OS/390 services
 - New CICS TS port from OS/390
 - Released with VSE/ESA 2.4

- **CICS TS for VSE/ESA 1.1.1 - September 2000**
 - Provides CICS Web Support (CWS), 3270 Bridge, REXX for CICS, Subsystem Storage Protection (SSP), ...
 - Released with VSE/ESA 2.5

- **CICS Explorer “display only” support - released June 2012**

- **CICS TS for z/VSE V2.1 – November 2015**
 - Comes with z/VSE V6.1
 - Support for Channels and Containers
 - CICS Explorer update capability

- **CICS TS for z/VSE V2.2 – December 2017**
 - Comes with z/VSE 6.2



Agenda

- **CICS Overview**
- **CICS TS V2.1 – z/VSE 6.1**
 - Channels and Containers
- **CICS TS V2.2 – z/VSE 6.2**
 - HTTP 1.1 Support
 - CICS Web Support with OpenSSL
- **CICS Connectivity options**
 - CICS Explorer
 - CICS Web Support
 - CICS Transaction Gateway
 - z/VSE SOAP Engine
 - z/VSE REST Engine
- **EZA Multiplexer & EZA OpenSSL Support**



CICS TS for z/VSE 2.1



- **A new CICS TS version for z/VSE**
 - The first major CICS TS update since 1999

- **Only available for z/VSE 6.1 and later, replaces CICS TS for VSE/ESA 1.1.1**
 - CICS TS for VSE/ESA 1.1.1 still delivered with z/VSE Version 5

- **New CICS TS for z/VSE V2.1 includes**
 - CICS Explorer update capability
 - Channel & Container support
 - The following CICS requirements are fulfilled
 - More current cypher suites (AES128/256) to CICS Web Support
 - Support for EXEC CICS INQUIRE SYSTEM OSLEVEL
 - Millisecond support in EXEC CICS ASKTIME
 - Millisecond option to EXEC CICS FORMATTIME

- **CICS Distributed Data Management (CICS/DDM) is not supported**

Channels and Containers

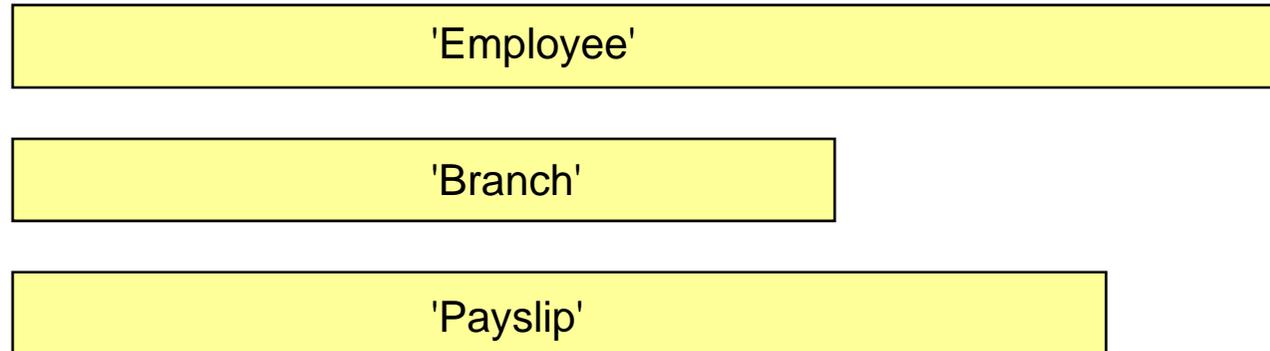


- **Channels and containers** lift the 32K Commarea limitation
 - Applicable for both LINK and XCTL, Distributed Program Link (DPL)
 - Affects the exchange of data between CICS tasks
 - Local and transaction routing
 - START with data

- **z/VSE ported the channel and container APIs based on CICS TS for z/OS 3.1**
 - Language support is provided for C, COBOL, HLASM, and PL/I

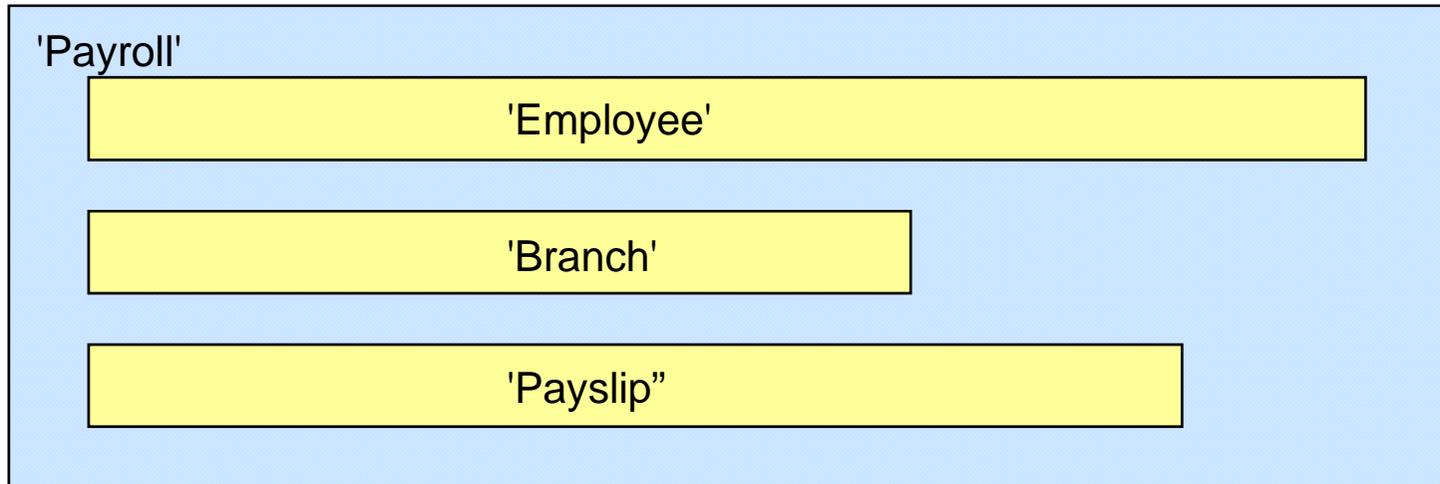
- **Channels and Containers limitations**
 - In 31 bit virtual storage only
 - No support for
 - External CICS Interface (EXCI), External Call Interface (ECI)
 - EXEC CICS WEB ... commands to receive/send data directly into/from containers
 - Business Transaction Services (BTS)

Containers



- **To solve the 32K Commarea problem a new construct is provided**
- **Named block of data designed for passing information between programs**
 - Like named COMMAREAs
- **CONTAINER API**
 - Created using (EXEC CICS) PUT CONTAINER, defines the size of the container
 - Read using (EXEC CICS) GET CONTAINER
 - Delete using (EXEC CICS) DELETE CONTAINER, to free storage, if no longer required
- **No CICS enforced size limitation**
 - Containers are stored within the CICS EDSA (31 bit partition virtual storage)

Channels



- **A group of Containers**
 - No limit on the number of Containers in a Channel
- **A Channel is a sort of program interface**
 - Passed on LINK, XCTL, pseudo-conversational RETURN, and START commands
- **Non-persistent**
 - Non-recoverable resource similar to commareas

Agenda

- **CICS Overview**
- **CICS TS V2.1 – z/VSE 6.1**
 - Channels and Containers
- **CICS TS V2.2 – z/VSE 6.2**
 - HTTP 1.1 Support
 - CICS Web Support with OpenSSL
- **CICS Connectivity options**
 - CICS Explorer
 - CICS Web Support
 - CICS Transaction Gateway
 - z/VSE SOAP Engine
 - z/VSE REST Engine
- **EZA Multiplexer & EZA OpenSSL Support**



CICS TS for z/VSE 2.2



- **Only available for z/VSE 6.2 and later, replaces CICS TS for z/VSE 2.1**
 - CICS TS for VSE/ESA 1.1.1 still delivered with z/VSE Version 5
 - CICS TS for z/VSE 2.1 still delivered with z/VSE 6.1

- **New CICS TS for z/VSE V2.2 includes**
 - **Enhancements to the CICS Explorer to more easily manage CICS resources:**
 - Define new CICS resources and modify or delete existing resources
 - Monitor, control, and update dynamic storage areas and global temporary storage queue statistics
 - Support "definitions" views for selected CICS resources
 - **HTTP 1.1 Support for CICS Web Support:**
 - Persistent connections, pipelining, and chunking
 - **Enhancements to the CICS API to provide:**
 - Support for UTF-8 and UTF-16 with the channels and containers API
 - Support for the APPEND parameter for PUT CONTAINER
 - Support for the BYTEOFFSET parameter for GET CONTAINER
 - Support for Internet-type date and time stamp formats
 - Support for Language Environment (LE) MAIN for Assembler applications.
 - **Support for OpenSSL with CICS Web Support**

HTTP 1.1 Support



- **CICS Web Support has been upgraded to comply with HTTP 1.1**
 - Provides support for the latest web browsers and applications
 - Ported from CICS TS for z/OS 3.1, CICS acting as a server
 - TCPIP SERVICE PROTOCOL(HTTP|ECI|USER)

- **New function has been added:**
 - **Persistent connections**
 - Allows to keep a connection open so that additional HTTP requests can flow over the same connection
 - Avoids connection establishment overhead for frequent requests
 - **Pipelining**
 - Allows to flow multiple HTTP requests over a single (persistent) connection
 - Subsequent request can be transmitted before the response of the first one has been received
 - Response must be returned in the same sequence as request was received
 - **Chunking**
 - Allows to send data in several smaller chunks, each with its own size and data
 - No longer need to know the complete size of the data before sending the data
 - **Support for additional HTTP methods:**
 - OPTIONS: To get capabilities of the server without requesting a resource
 - TRACE: Client can see what the other end received

Misc. CICS enhancements



- **Relative addressing instructions in Assembler programs (without base register)**
 - New operands added to DFHEIENT and DFHEIRET macros
 - Beneficial for translated programs that are greater than 4095 bytes

- **Common date and time stamp formats used on the internet**
 - Define correct date and time stamp in HTTP header
 - New CONVERTTIME command and new option for FORMATTIME

- **Language Environment (LE) MAIN for Assembler applications**
 - New translator option LEASM to enable LE functions and setup LE environment
 - Assembler programs translated with LEASM can be used as task-related user exits (TRUEs) or global user exits (GLUEs)

- **New SIT parameter: MAXSOCKETS**
 - Specifies the maximum number of TCP/IP sockets, that can be handled by CICS

UTF-8 and UTF-16 support for Channels and Container



- **Most textual data in the internet is encoded in UTF-8 nowadays**
 - XML, JSON, HTML, etc.
- **PUT and GET can be used for data conversion**
- **Uses CICS conversion tables or LE provided conversion services (ICONV)**
 - CICS conversion tables are used for simple codepages
 - ICONV is used for UTF8 and UTF-16
- **Simple example of converting data to UTF-8:**

```
EXEC CICS PUT CONTAINER('temp') CHANNEL('dummy')  
        FROM(ebcdic-data)  
        CHAR
```

```
EXEC CICS GET CONTAINER('temp') CHANNEL('dummy')  
        SET(utf8-ptr) FLENGTH(utf8-len)  
        INTOCCSID(1208)      (CCSID 1208 = UTF-8)
```

OpenSSL support for CICS Web Support



- **CICS TS for VSE/ESA 1.1.1 and CICS TS for z/VSE 2.1:**
 - Only supports CSI 's SSL/TLS implementation
 - OpenSSL can not be (natively) used
 - Circumvention: use IPv6/VSE's SSL Proxy or ATTLS as pass-through
 - But: CICS Web Support works with any TCP/IP stack (socket calls)
 - CSI, BSI or LFP

- **CICS TS for z/VSE 2.2:**
 - Allows to use the OpenSSL or the CSI SSL/TLS implementation
 - Default remains CSI's implementation
 - SSL implementation can be chosen via // SETPARM statement in JCL
 - SSL implementation is now independent of TCP/IP stack
 - OpenSSL requires a slightly different key and certificate setup
 - PEM files instead of .PRVK, .ROOT, .CERT members

Agenda

- **CICS Overview**
- **CICS TS V2.1 – z/VSE 6.1**
 - Channels and Containers
- **CICS TS V2.2 – z/VSE 6.2**
 - HTTP 1.1 Support
 - CICS Web Support with OpenSSL
- **CICS Connectivity options**
 - CICS Explorer
 - CICS Web Support
 - CICS Transaction Gateway
 - z/VSE SOAP Engine
 - z/VSE REST Engine
- **EZA Multiplexer & EZA OpenSSL Support**



CICS Explorer

- **CICS Explorer “display only” in z/VSE Version 5**
 - System management framework for CICS TS
 - Consists of CICS Explorer client and a CICS TS server extension
 - CICS Explorer client
 - Read-only capabilities (like CEMT INQUIRE)
 - Eclipse-based user interface on workstation
 - Connects to CICS TS via TCP/IP - Communication via HTTP requests
 - One CICS Explorer client for z/VSE and z/OS
 - CICS Explorer server extension
 - Delivered as PTF for CICS TS for VSE/ESA 1.1.1

- **Integrated into CICS TS for z/VSE 2.1 (z/VSE 6.1)**
 - Provides update capability to CICS resources (like CEMT SET)
 - Update resources as you would do with transactions on your CICS terminal
 - Enable / disable CICS resources, change selected CICS definitions, ...

- **Integrated into CICS TS for z/VSE 2.2 (z/VSE 6.2)**
 - Define new CICS resources and modify or delete existing resources (like CEDA)
 - Monitor, control, and update dynamic storage areas and global temporary storage queue statistics.



CICS Explorer – connected ...

The screenshot displays the IBM CICS Explorer application window. The main area shows a table of CICS programs for the PRODCICS region. A red arrow points to the 'Programs' tab in the top navigation bar. An overlaid terminal window shows the output of a CICS command, listing program details such as length, language, and status.

Region	Name	Status	Use Count	Concurrent Us...	Language	Share Status	CEDF Status	NEWCOPY Status
PRODCICS	\$EDCTCPM	✓ ENABLED	0	0	C	N/A	CEDF	NOTREQUIRED
PRODCICS	\$EDTCPV	✓ ENABLED	0	0	C	N/A	CEDF	NOTREQUIRED
PRODCICS	ARXITCPU	✓ ENABLED	0	0	ASSEMBLER	N/A	CEDF	NOTREQUIRED
PRODCICS	BSTADMII	✓ ENABLED	0	0	ASSEMBLER	N/A	CEDF	NOTREQUIRED
PRODCICS	CEEBINT	✓ ENABLED	1	1	ASSEMBLER	N/A	CEDF	NOTREQUIRED
PRODCICS	CEEBNATX	✓ ENABLED	0	0				
PRODCICS	CEECBLDY	✓ ENABLED	0	0				
PRODCICS	CEECICS	✓ ENABLED	1	1				
PRODCICS	CEECDATX	✓ ENABLED	0	0				
PRODCICS	CEECMI	✓ ENABLED	0	0				
PRODCICS	CEECOPT	✓ ENABLED	1	1				
PRODCICS	CEECRHP	✓ ENABLED	0	0				
PRODCICS	CEECXITA	✓ ENABLED	1	1				
PRODCICS	CEECXTAN	✓ ENABLED	1	1				
PRODCICS	CEECZST	✓ ENABLED	0	0				
PRODCICS	CEEDATE	✓ ENABLED	0	0				
PRODCICS	CEEDATM	✓ ENABLED	0	0				
PRODCICS	CEEDAYS	✓ ENABLED	0	0				
PRODCICS	CEEDCOD	✓ ENABLED	0	0				
PRODCICS	CEEDSHP	✓ ENABLED	0	0				
PRODCICS	CEEDYWK	✓ ENABLED	0	0				
PRODCICS	CEENV	✓ ENABLED	0	0				
PRODCICS	CEEEV000	✓ ENABLED	0	0				
PRODCICS	CEEEV001	✓ ENABLED	0	0				
PRODCICS	CEEEV002	✓ ENABLED	0	0				
PRODCICS	CEEEV003	✓ ENABLED	1	1				
PRODCICS	CEEEV004	✓ ENABLED	0	0				
PRODCICS	CEEEV005	✓ ENABLED	1	1				
PRODCICS	CEEEV006	✓ ENABLED	0	0				
PRODCICS	CEEEV007	✓ ENABLED	0	0				
PRODCICS	CEEEV008	✓ ENABLED	0	0				
PRODCICS	CEEEV009	✓ ENABLED	0	0				
PRODCICS	CEEEV010	✓ ENABLED	1	1				
PRODCICS	CFFFV011	✓ FNARI FD	0	0				

```

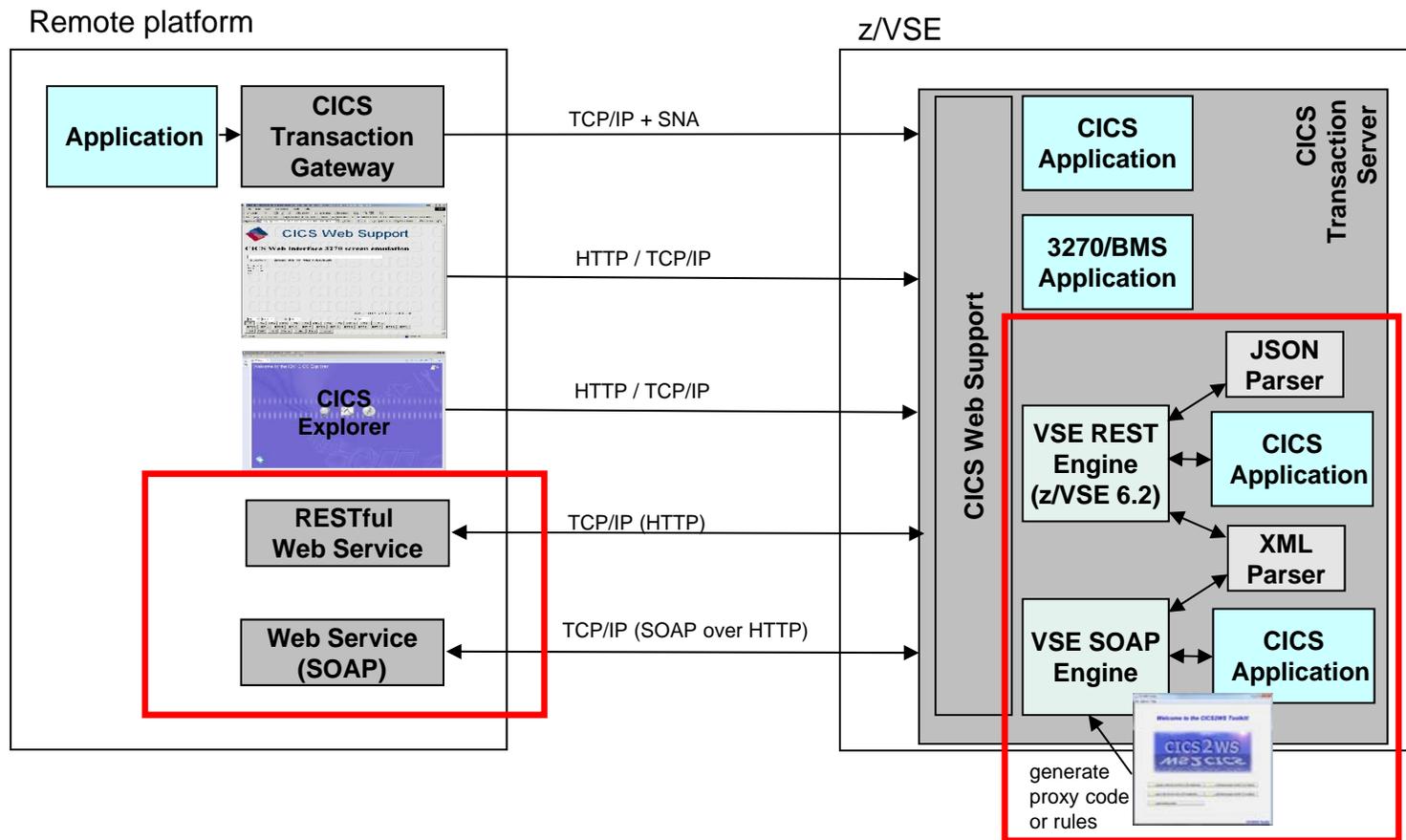
I PROG
STATUS: RESULTS - OVERTYPE TO MODIFY
  Prog($EDCTCPM) Len(0000000) C Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Uex Ful
  Prog($EDTCPV) Len(0000000) C Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Uex Ful
  Prog(ARXITCPU) Len(0000000) Ass Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Uex Ful
  Prog(BSTADMII) Len(0000000) Ass Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Cex Ful
  Prog(CEEBINT ) Len(00000008) Ass Pro Ena Pri Ced
    Res(001) Use(0000000001) Bel Uex Ful
  Prog(CEEBNATX) Len(0000000) Ass Pro Ena Sha Ced
    Res(000) Use(0000000000) Bel Uex Ful
  Prog(CEECBLDY) Len(0000000) Ass Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Uex Ful
  Prog(CEECICS) Len(0043464) Ass Pro Ena Sha Ced
    Res(001) Use(0000000001) Bel Uex Ful
+ Prog(CEECDATX) Len(0000000) Ass Pro Ena Pri Ced
    Res(000) Use(0000000000) Bel Uex Ful

                                SYSID=CIC2 APPLID=PRODCICS
RESPONSE: NORMAL                                TIME: 16.21.55 DATE: 09.28.12
PF 1 HELP      3 END                          7 SBH 8 SFH 9 MSG 10 SB 11 SF
    
```

At the bottom of the terminal window, the connection information is visible: IZE0100I Connected user SYSA to host l...m1.boeblingen.de.ibm.com on port 27283.

CICS Connectivity

CICS Web Support is the base of CICS connectivity



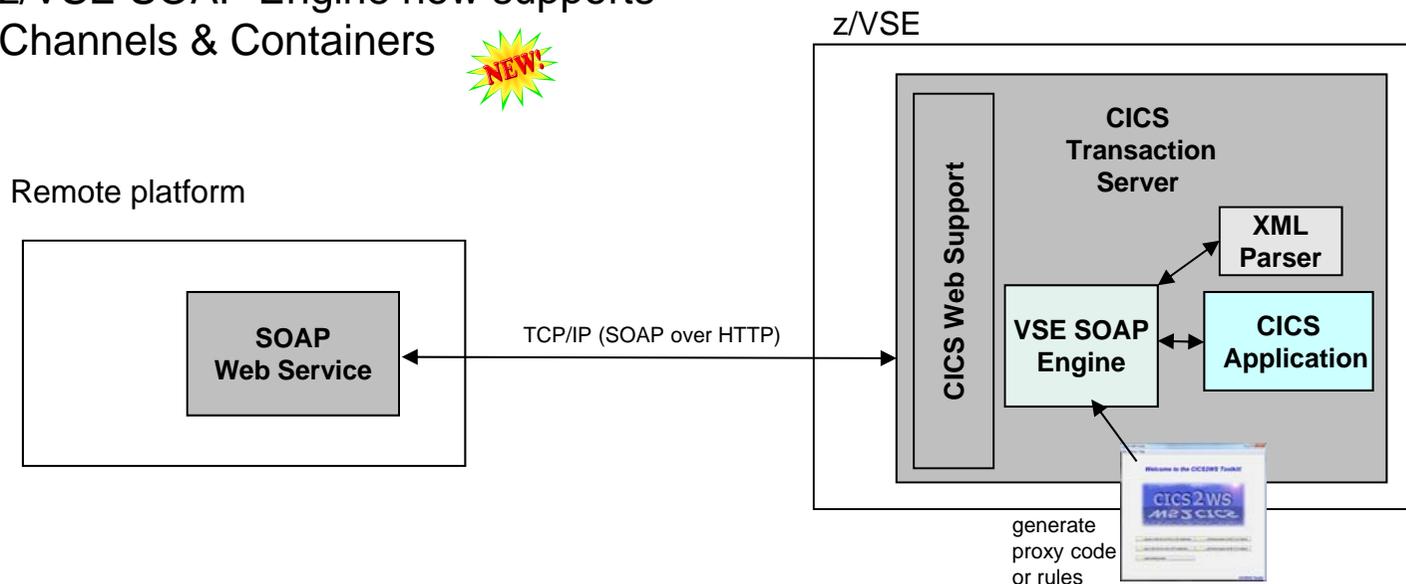
SOAP / Web Services support

- **Web Service-enable z/VSE CICS applications**

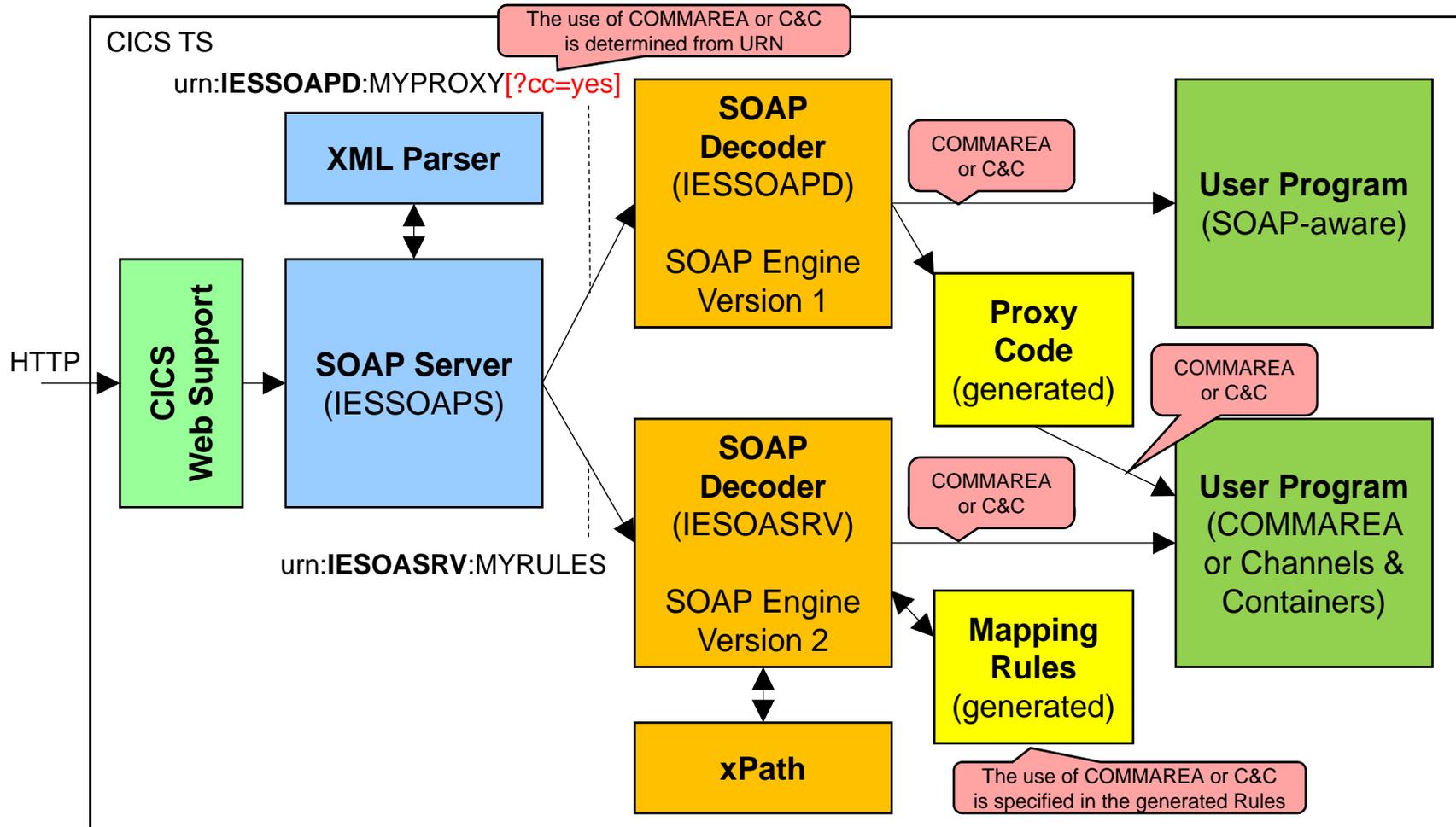
- Provide existing CICS applications as Web Service to the outside world
 - z/VSE as the SOAP server
- Use/call external Web Services from within z/VSE CICS applications
 - z/VSE as the SOAP client
- CICS2WS Tool is used to generate proxy code or mapping rules

- **z/VSE 6.2:**

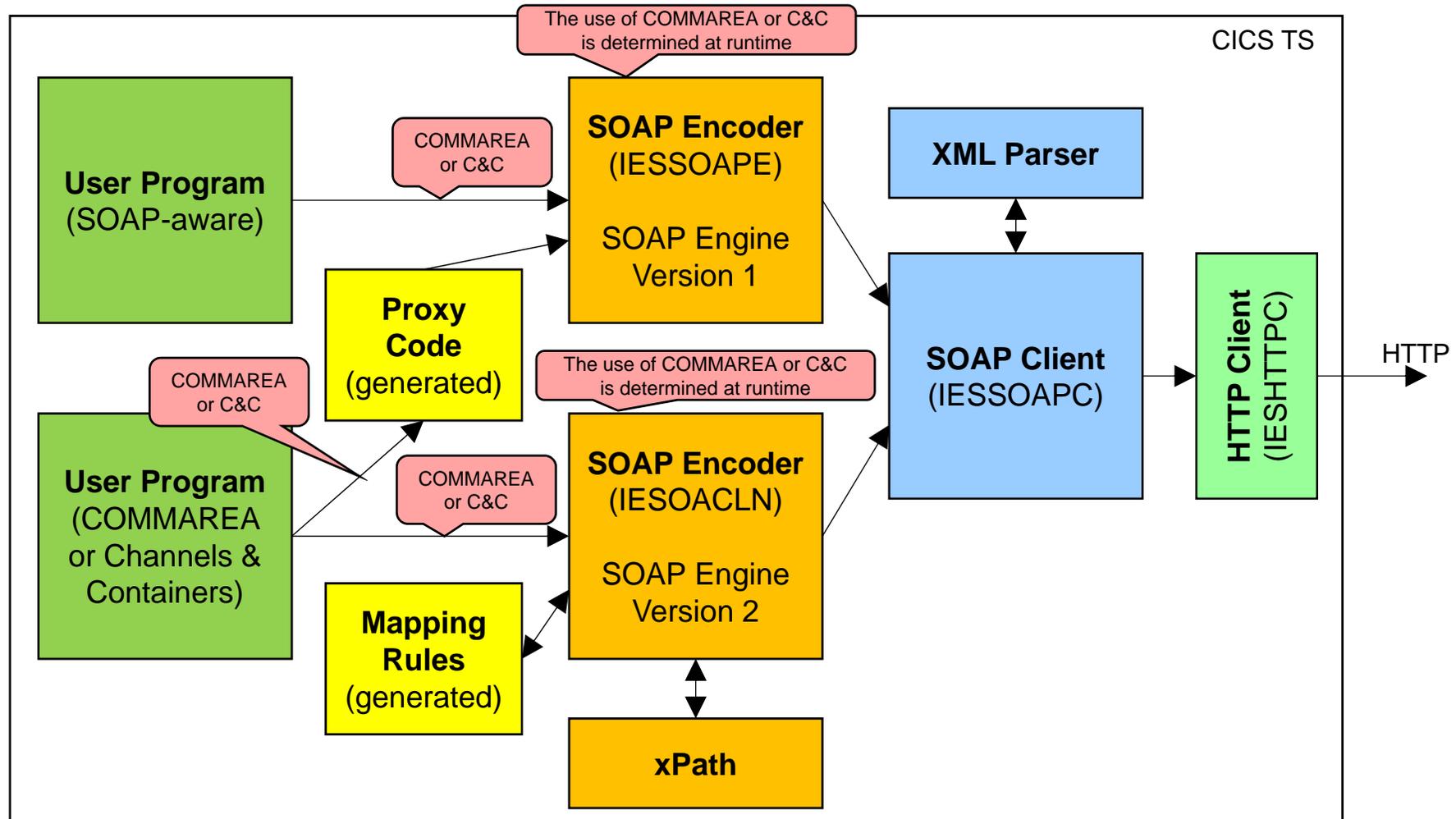
- z/VSE SOAP Engine now supports Channels & Containers



z/VSE as SOAP Server



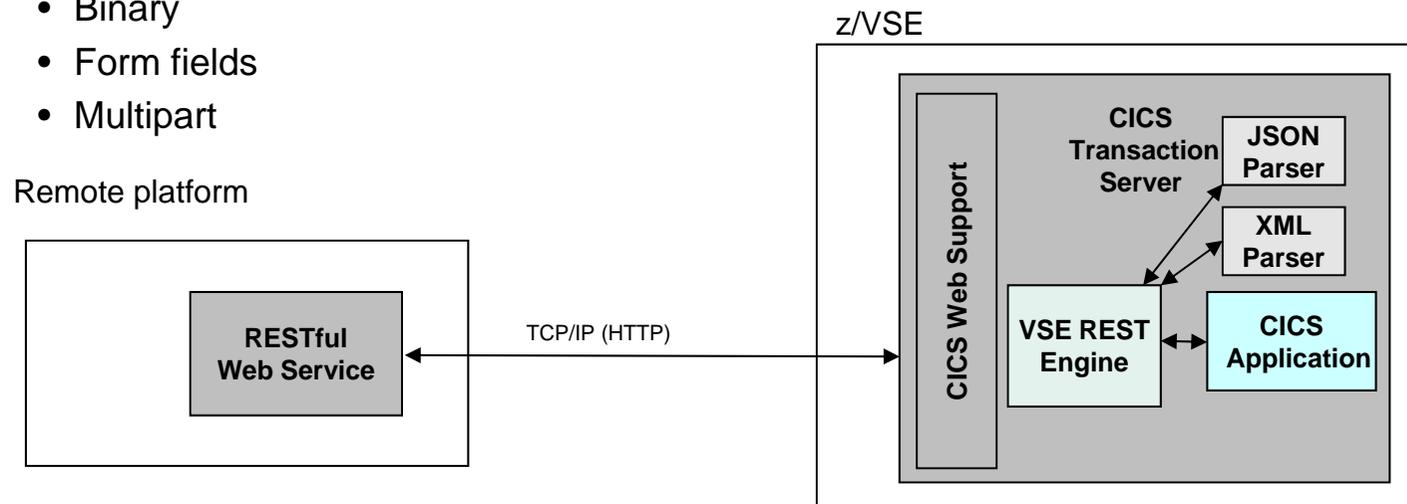
z/VSE as SOAP Client



z/VSE 6.2: RESTful Web Services support



- **Use REST (Representational State Transfer) with CICS applications**
 - Provide existing CICS applications as RESTful Web Service to the outside world
 - z/VSE as the REST server
 - Use/call external RESTful Web Services from within z/VSE CICS applications
 - z/VSE as the REST client
 - Payload can be:
 - JSON (JavaScript Object Notation)
 - XML
 - Plain text
 - Binary
 - Form fields
 - Multipart



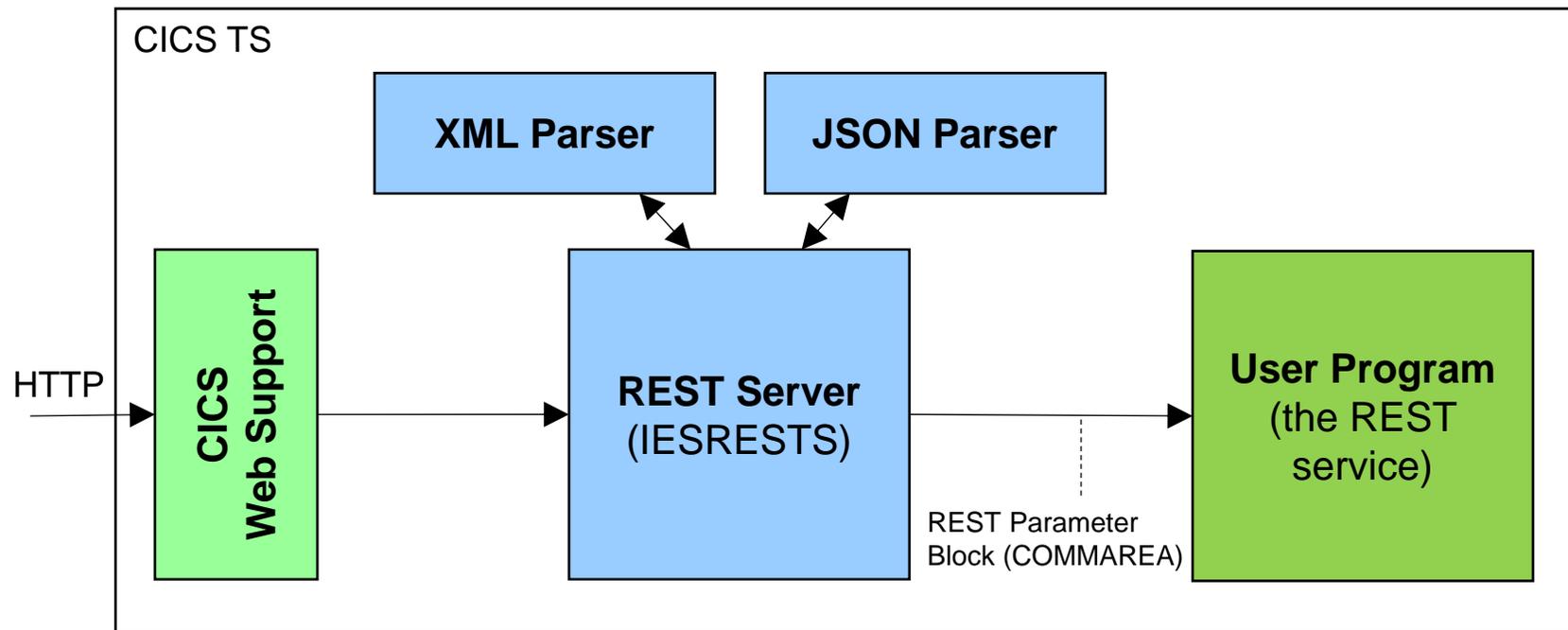
What is REST (Representational State Transfer)?



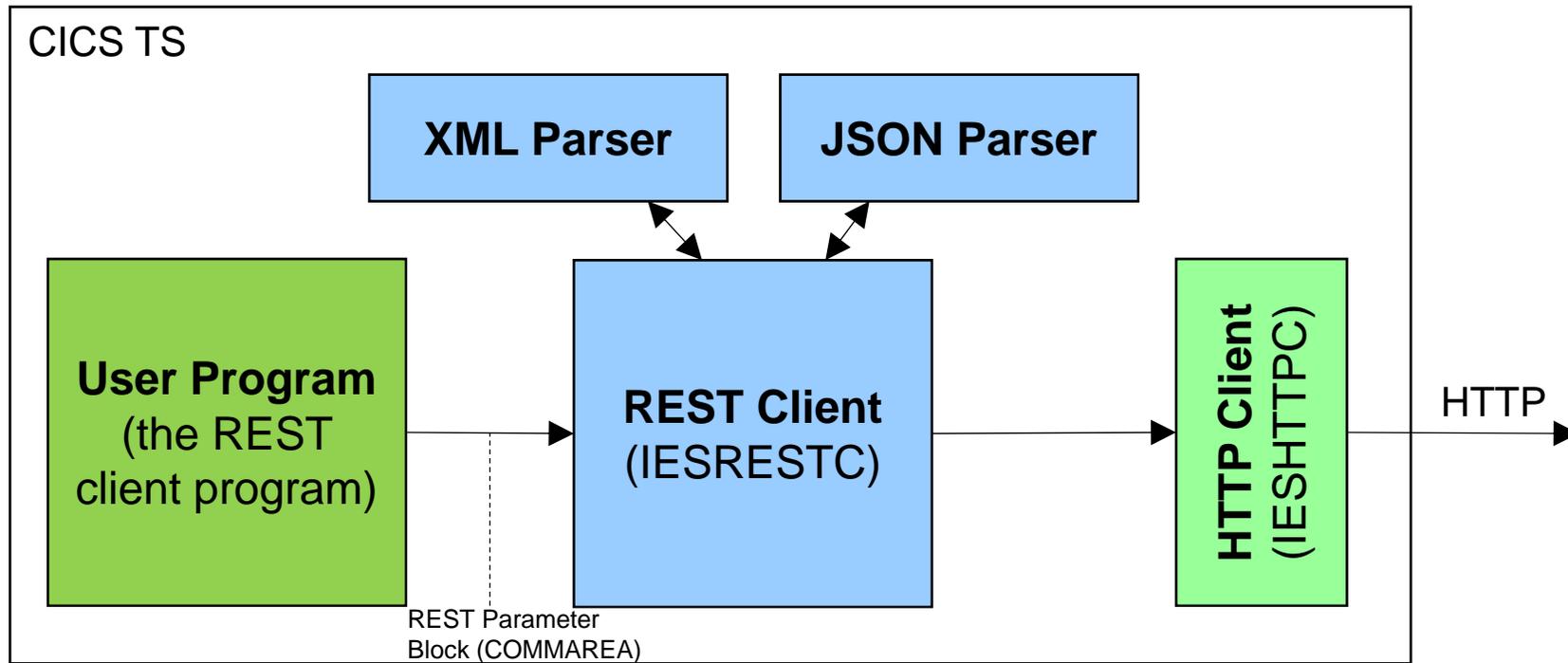
- Representational State Transfer (REST) is a **software architecture style consisting of guidelines and best practices** for creating web services
 - REST has gained widespread acceptance across the web as a **simpler alternative to SOAP** and WSDL-based web services
- RESTful systems typically communicate over the Hypertext Transfer Protocol (**HTTP**) with the same HTTP verbs (GET, POST, PUT, DELETE, and so on) used by web browsers
- The payload (message) transported by RESTful web services can be of various types (content types)
 - Commonly used is **JSON** as well as **XML**, but it can also be plain text, or even binary data
- A RESTful web service typically operates on a certain '**object**' on a server
 - The object is typically addressed through the URI (part of the URL), for example **http://example.com/resource**
- Actions on such resources are typically denoted by the HTTP request types:
 - **GET** would typically **read** the resource
 - **PUT** would typically **update/replace** the resource
 - **POST** would typically **create** the resource
 - **DELETE** would typically **delete** the resource
- RESTful web services are typically **stateless**
 - Each request from any client contains all the information necessary to service the request
 - The session state is therefore held in the client

→ As denoted by the term 'typically' in above descriptions, there is no hard requirement for any of the described properties

z/VSE 6.2: z/VSE as a REST Server



z/VSE 6.2: z/VSE as a REST Client



Agenda

- **CICS Overview**
- **CICS TS V2.1 – z/VSE 6.1**
 - Channels and Containers
- **CICS TS V2.2 – z/VSE 6.2**
 - HTTP 1.1 Support
 - CICS Web Support with OpenSSL
- **CICS Connectivity options**
 - CICS Explorer
 - CICS Web Support
 - CICS Transaction Gateway
 - z/VSE SOAP Engine
 - z/VSE REST Engine
- **EZA Multiplexer & EZA OpenSSL Support**

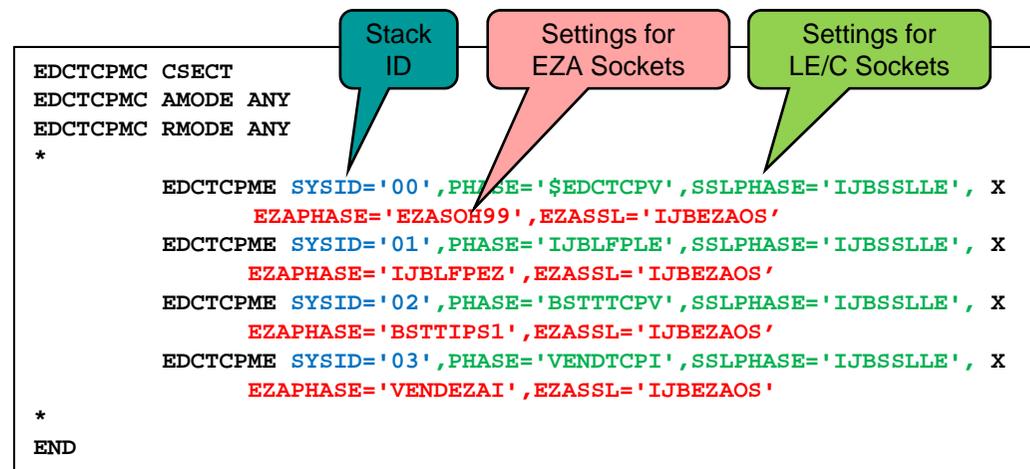


z/VSE 6.2: EZA Multiplexer



- With the **EZASOKET** and **EZASMI** interfaces you can specify which socket interface module to the TCP/IP partition is to be used
 - Default: EZASOH99 (for TCP/IP for z/VSE)
- **Select the EZA socket interface routine:**
 - Via JCL statement: // SETPARM [SYSTEM,]EZA\$PHA='phasename'
 - Via parameter ADSNAME on the EZAAPI/EZASOKET INITAPI call
- The **EZA Multiplexer** can be used to ease the correct setup of socket interface modules for the corresponding stack IDs
 - The multiplexer allows you to perform a **one time setup** and to assign the corresponding socket interface modules to the stack IDs
 - The use of the multiplexer is transparent for your application

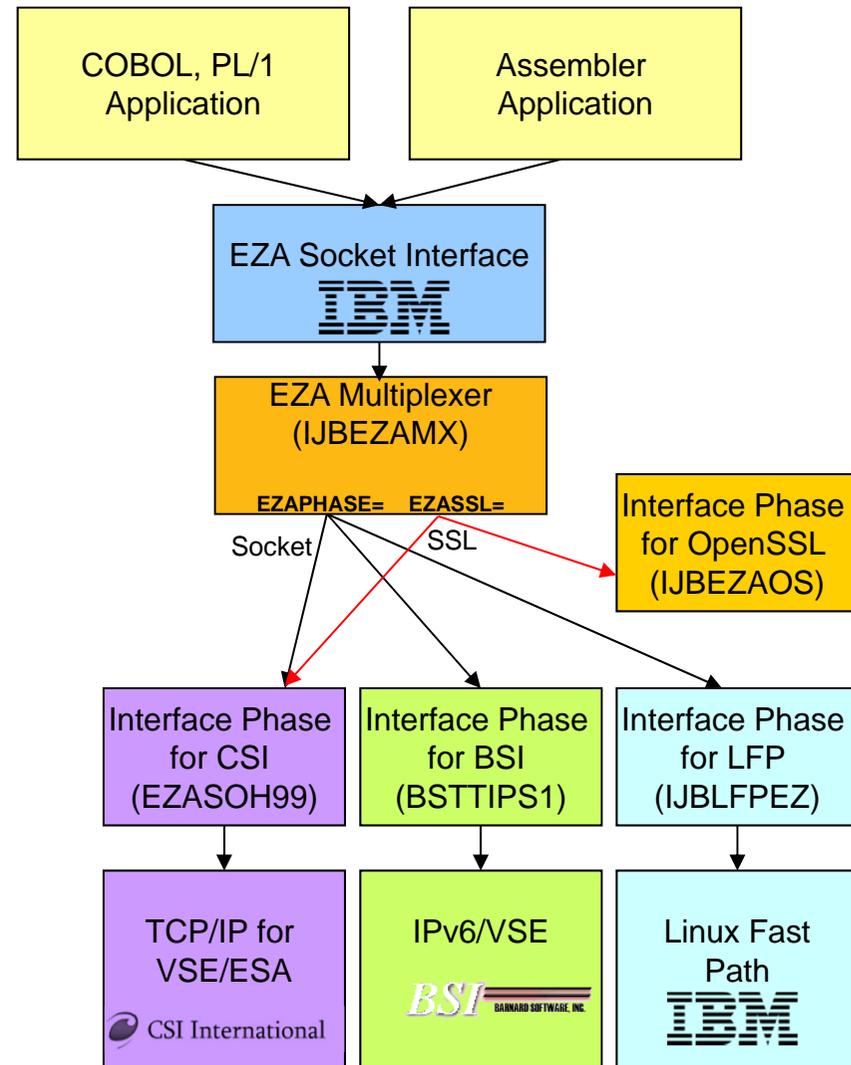
- **Setup:**
 - Select **IJBEZAMX** as EZA interface phase for all applications (via SETPARM SYSTEM)
 - Configure Multiplexer via **EDCTCPMC** in ICCF library 62
 - Same configuration as for LE/C Socket API Multiplexer
 - Additional parameters now allow to specify EZA interfaces



z/VSE 6.2: EZA OpenSSL Support

- Besides the EZA socket interface routine, the EZA Multiplexer also allows you to specify an **alternative EZA SSL interface routine**
 - Default: The same as the EZA socket interface routine
- The new EZA SSL interface routine **IJBEZAOS** provides an interface to z/VSE’s OpenSSL implementation
- The use of an alternative EZA SSL interface routine is transparent for your application
 - OpenSSL uses different key and certificate formats (e.g .PEM instead of .PRVK, .ROOT, .CERT)

→ This makes z/VSE’s OpenSSL support available for non-LE/C applications (i.e. COBOL, PL/1, HLASM)



Questions ?



THANK YOU