Sysplex Network Technology Update

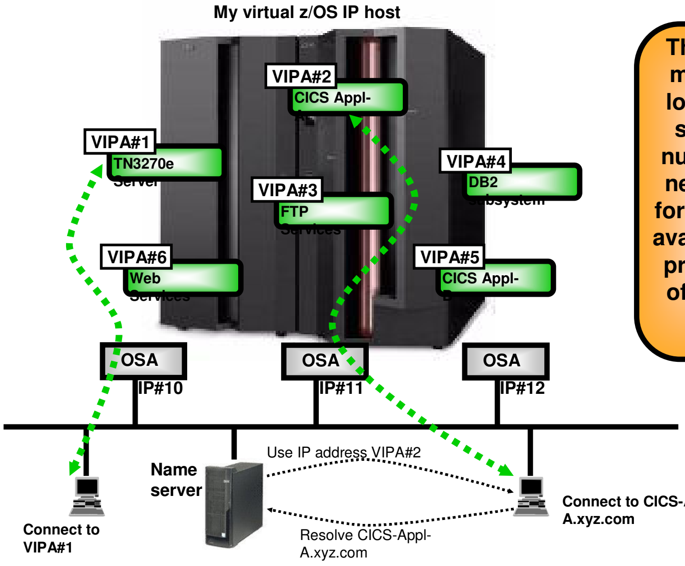
IBM Software Group, Enterprise Networking and Transformation Solutions

Content

- Review recent CS for z/OS Sysplex Network Enhancements:
  - z/OS V1R6
  - Sysplex Autonomics (z/OS V1R6 and V1R7)
  - Improving the load balancing decisions
  - Improving utilization of XCF Communication resources

The network view of a Parallel Sysplex

A single large server with many network interfaces and many services



The objective is to make the Sysplex look like one large server that has a number of physical network interfaces for performance and availability - and that provides a number of highly available and scalable services.

- ✓Single-system image (SSI)
- ✓Scalable
- ✓Highly available
- ✓Secure

Sysplex Enhancements
z/OS V1R6

Support for IPv6 (z/OS V1R6)

- Most Sysplex functions are enabled for IPv6 exploitation

- Dynamic VIPAs
  - VIPADEFINE/VIPABACKUP/VIPADELETE/VIPARANGE

- Dynamic XCF

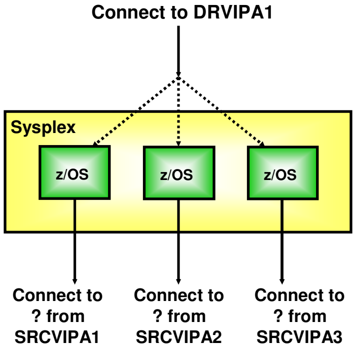
- Distributable DVIPAs
  - VIPADISTRIBUTE

- Sysplex Enhancements
  - Sysplex Ports
  - Sysplex Sockets
  - TCPStackSourceVipa

- No IPv6 support for
  - Sysplex-Wide Security Associations (SWSA)
  - Multinode Load Balancing (MNLB)
  - HiperSockets (Available with z9 Processor and z/OS V1R7)

Single system image (SSI) from an IP perspective in the Sysplex

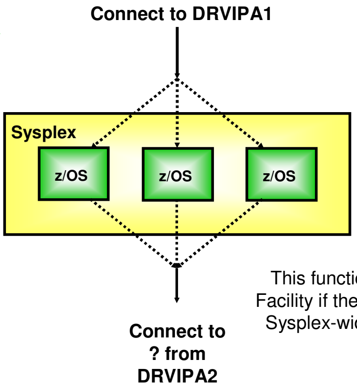
Inbound SSI


Connect to DRVIPA1
Sysplex
z/OS
z/OS
z/OS
Connect to ? from SRCVIPA1
Connect to ? from SRCVIPA2
Connect to ? from SRCVIPA3

- We have single system image capability for inbound connections where a single distributed VIPA address can represent all images in the Sysplex - and remote users do not need to select a specific image when connecting to their server application.
- But if we establish outbound connections from the images in the Sysplex, each image has its own source VIPA address - so there is no single system image from an outbound connection perspective - which has implications in firewall filter setup, etc.

Single system image (SSI) from an IP perspective in the Sysplex

Outbound SSI

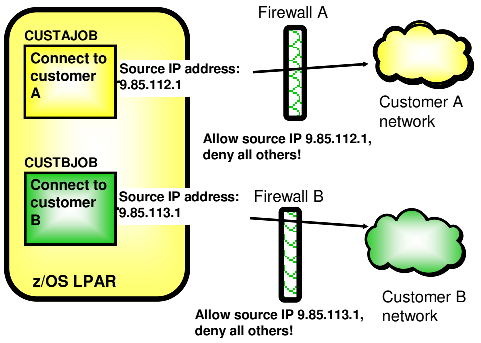


This function requires a Coupling Facility if the source VIPA address is Sysplex-wide (SYSPLEXPORTS).

- z/OS V1R4 introduced new capabilities (TCPSTACKSOURCEVIPA) that allow a single Sysplex-wide source VIPA address to be used for outbound TCP connections by all images in the Sysplex - resulting in single system image capabilities for both inbound and outbound connections.
- z/OS V1R5 significantly improved the performance when using a Sysplex-wide source VIPA

Job-specific source IP address control

Added in V1R6 for easier firewall filter rule administration


Firewall A
CUSTAJOB
Connect to customer A
Source IP address: 9.85.112.1
Customer A network
Allow source IP 9.85.112.1, deny all others!
CUSTBJOB
Connect to customer B
Source IP address: 9.85.113.1
Firewall B
Customer B network
Allow source IP 9.85.113.1, deny all others!
z/OS LPAR

```
BEGINSRCIP
      CUSTAJOB    9.85.112.1
      CUSTBJOB    9.85.113.1
      User1*      888:555::222     ===> Wildcards
allowed!
ENDSRCIP
```

Extending configuration control over which local IP address to use for outbound connections from z/OS

- ✓ Outbound connections can use same IP addresses as inbound connections to same application without application change:
  - Easier for accounting and management
  - Easier for security (firewall admin)
  - Permits source IP address selection controls for applications even when application doesn't provide for this programmatically (most don't, but some do!)
- ✓ Introducing Job-specific Source IP Addressing
  - A new TCPIP.Profile statement BEGINSRCIP/ENDSRCIP allows the selection of a source IP address for outbound TCP connections by job name
  - Overrides TCPSTACKSOURCEVIPA and SOURCEVIPA specifications

Job-specific source IP address control (cont.)

Supports all IP addresses:

- ✓Physical IP addresses
- ✓Static VIPAs
- ✓Dynamic VIPAs
- ✓Distributed DVIPAs

Socket Application

```
socket()
bind(inaddr_any, port)     <== Most TCP applications do not
                           issue an explicit bind (they allow
                           TCP to assign the local IP address
                           and port when the connect is issued)


connect(destination IP address, destination port)
```

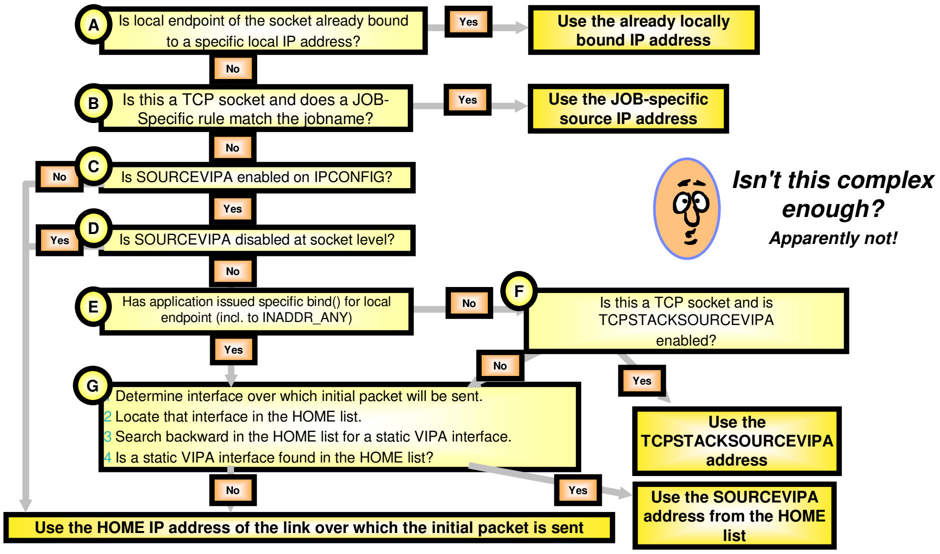
Supports TCP connections only!

Addresses client TCP applications that explicitly issue bind() socket API without specifying an IP address, prior to issuing the connect() socket API

- ✓TCPSTACKSOURCEVIPA did not support these types of applications
- ✓Job-specific source IP address controls can be used as a replacement for TCPSTACKSOURCE VIPA

```
BEGINSRCIP
     *   10.1.1.1   ===> Completely overides TCPSTACKSOURCEVIPA specifications!
ENDSRCIP
```

Selecting source IP address for outbound IPv4 connections or associations in CS z/OS V1R6



A Is local endpoint of the socket already bound to a specific local IP address?

Yes → Use the already locally bound IP address

No

B Is this a TCP socket and does a JOB-Specific rule match the jobname?

Yes → Use the JOB-specific source IP address

No

C Is SOURCEVIPA enabled on IPCONFIG?

No

Yes

D Is SOURCEVIPA disabled at socket level?

Yes

No

E Has application issued specific bind() for local endpoint (incl. to INADDR_ANY)

No

F Is this a TCP socket and is TCPSTACKSOURCEVIPA enabled?

Yes → Use the TCPSTACKSOURCEVIPA address

No

Yes

G Determine interface over which initial packet will be sent.
2 Locate that interface in the HOME list.
3 Search backward in the HOME list for a static VIPA interface.
4 Is a static VIPA interface found in the HOME list?

Yes → Use the SOURCEVIPA address from the HOME list

No

Use the HOME IP address of the link over which the initial packet is sent

Isn't this complex enough?

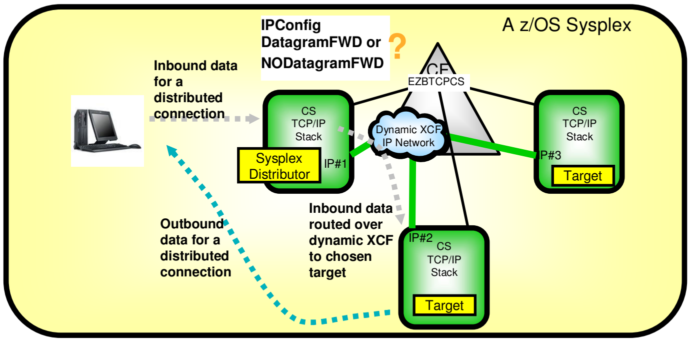
Apparently not!

General IP forwarding no longer required for Sysplex Distributor in z/OS V1R6

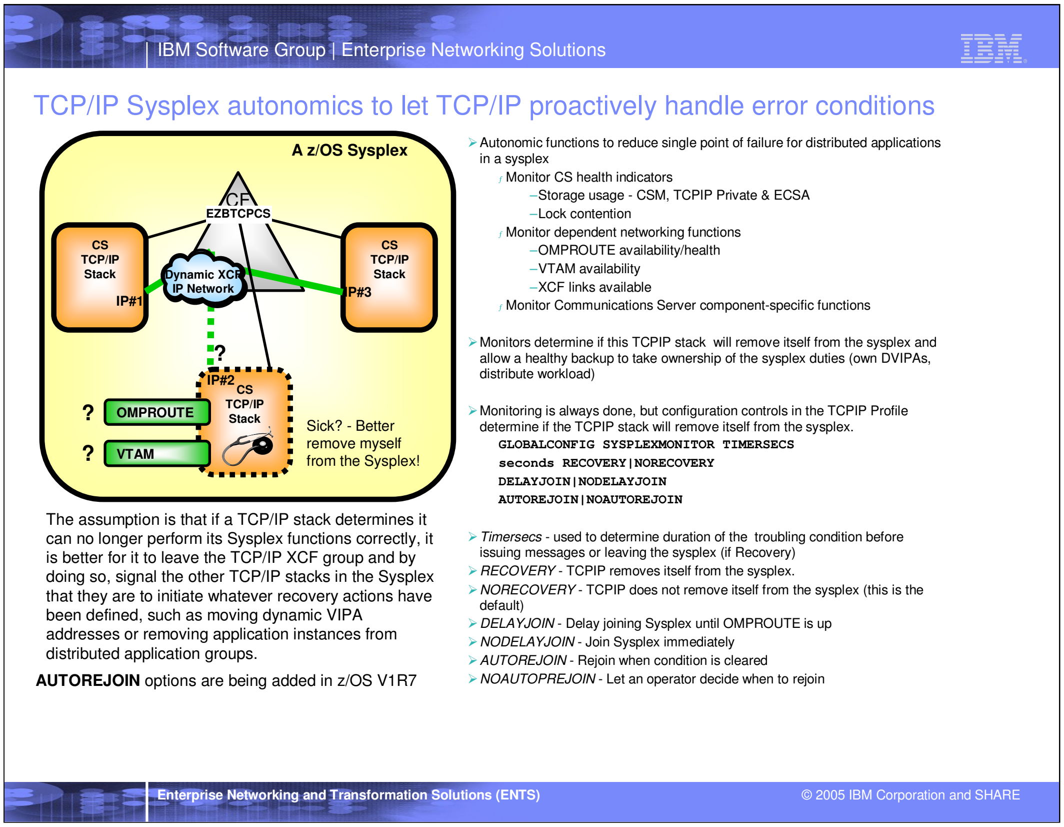
The distributing TCP/IP stack needs to forward both connection setup and inbound connection data over a dynamic XCF IP network to the chosen TCP/IP target stack in the sysplex.

- Previous to z/OS V1R6 it was a requirement that the distributing stack had to have DATAGRAMFWD enabled
  - This option means that the TCP/IP stack is allowed to route IP packets in general from any interface to any interface (only way to limit this general routing capability was via firewall filters on z/OS)

- In z/OS V1R6, use of Sysplex Distributor does not require DATAGRAMFWD to be enabled
  - Sysplex Distributor can now be deployed without any risk of using a z/OS stack as a general intermediate routing node



Sysplex Autonomics



IBM Software Group | Enterprise Networking Solutions

TCP/IP Sysplex autonomics to let TCP/IP proactively handle error conditions

Sick? - Better remove myself from the Sysplex!

The assumption is that if a TCP/IP stack determines it can no longer perform its Sysplex functions correctly, it is better for it to leave the TCP/IP XCF group and by doing so, signal the other TCP/IP stacks in the Sysplex that they are to initiate whatever recovery actions have been defined, such as moving dynamic VIPA addresses or removing application instances from distributed application groups.

AUTOREJOIN options are being added in z/OS V1R7

- Autonomic functions to reduce single point of failure for distributed applications in a sysplex
  - Monitor CS health indicators
    - Storage usage - CSM, TCPIP Private & ECSA
    - Lock contention
  - Monitor dependent networking functions
    - OMPROUTE availability/health
    - VTAM availability
    - XCF links available
  - Monitor Communications Server component-specific functions
- Monitors determine if this TCPIP stack will remove itself from the sysplex and allow a healthy backup to take ownership of the sysplex duties (own DVIPAs, distribute workload)
- Monitoring is always done, but configuration controls in the TCPIP Profile determine if the TCPIP stack will remove itself from the sysplex.

```
GLOBALCONFIG SYSPLEXMONITOR TIMERSECS
seconds RECOVERY|NORECOVERY
DELAYJOIN|NODELAYJOIN
AUTOREJOIN|NOAUTOREJOIN
```

- *Timersecs* - used to determine duration of the troubling condition before issuing messages or leaving the sysplex (if Recovery)
- *RECOVERY* - TCPIP removes itself from the sysplex.
- *NORECOVERY* - TCPIP does not remove itself from the sysplex (this is the default)
- *DELAYJOIN* - Delay joining Sysplex until OMPROUTE is up
- *NODELAYJOIN* - Join Sysplex immediately
- *AUTOREJOIN* - Rejoin when condition is cleared
- *NOAUTOPREJOIN* - Let an operator decide when to rejoin

Enterprise Networking and Transformation Solutions (ENTS)

© 2005 IBM Corporation and SHARE

Messages are always issued to the console when these conditions are detected regardless of SYSPLEXMONITOR Recovery specification
Messages are eventual action (deleted when the action is taken or problem is resolved)

New operator command is provided to allow TCPIP to leave the sysplex (ie. EZBTCPCS xcf group)
Vary TCPIP,,SYSPLEX,LEAVEGROUP

To have TCPIP rejoin the sysplex group, a Vary Obey of the TCPIP profile with sysplex configuration statements is needed.
Severe problems may require a TCPIP stack restart

Sysplex Autonomics:

What happens when a problem is encountered

Eventual Action WTOs are issued as the problem is detected

- Regardless of whether *RECOVERY* option is specified
- Warning messages can provide early warning

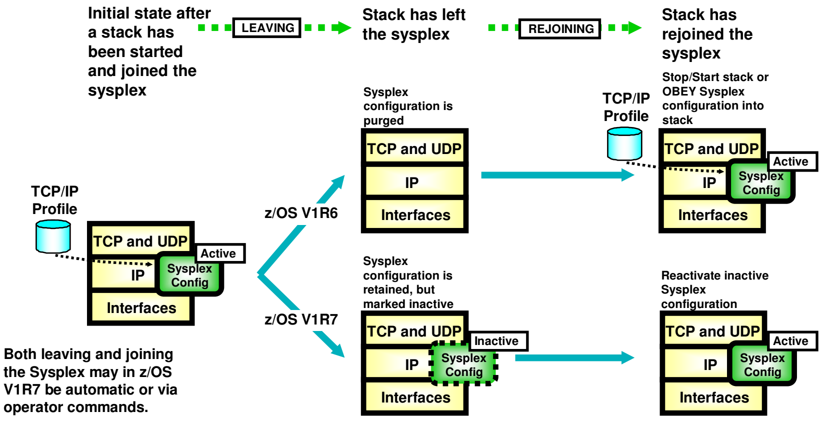
If *RECOVERY* is specified

- TCP/IP leaves its XCF Sysplex group (EZBTCPCS)
  - Allows other TCP/IP stacks to take over ownership responsibilities for DVIPAs (based on pre-defined VIPABACKUP configuration)
- All DVIPAs are deactivated on the affected system
  - Includes application instance DVIPAs (i.e. defined by VIPARANGE)
- The stack is no longer visible to other TCP/IP stacks in the sysplex
  - DynamicXCF connectivity is disabled on this stack
- The TCP/IP stack can continue processing for non-DVIPA workload
- Safety checks built to prevent unnecessary actions
  - For example, are any other stacks currently active in the sysplex?
- Designed for high availability configurations (i.e. VIPABACKUPs defined, etc.)

Manual Recovery actions can also be triggered via operator command

- V TCPIP,,SYSPLEX,LEAVEGROUP
- Same effect as above

TCP/IP stacks leaving and rejoining the sysplex



Initial state after a stack has been started and joined the sysplex

Stack has left the sysplex

Stack has rejoined the sysplex

Sysplex configuration is purged

Stop/Start stack or OBEY Sysplex configuration into stack

Sysplex configuration is retained, but marked inactive

Reactivate inactive Sysplex configuration

Both leaving and joining the Sysplex may in z/OS V1R7 be automatic or via operator commands.

- **Leaving the Sysplex in z/OS V1R6, purges sysplex configuration data from the stack's internal configuration blocks.**
  - To rejoin the sysplex, the sysplex configuration data must be reapplied to the stack's active configuration through a restart or an OBEY command
- **In z/OS V1R7, a stack's sysplex configuration data will be retained in an inactive status when a stack leaves the sysplex**
  - The inactive sysplex configuration data will be shown on the NETSTAT VIPADCFG report as inactive
  - Rejoining the sysplex will then reactivate the currently inactive sysplex configuration data

Determining saved DVIPA configuration once TCP/IP has left the sysplex

Example of a Netstat VIPADCFG/-F report after a TCP/IP stack Left the sysplex group (V1R7) via:

VARY TCPIP,,SYSPLEX,LEAVEGROUP

```
netstat vipadcfg
TCPCS is not a member of the TCP/IP sysplex group
ALL VIPADYNAMIC configuration for TCPCS is currently inactive
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS           20:59:46
Dynamic VIPA Information:

  VIPA Define:
    IpAddr/PrefixLen: 197.11.221.1/24
      Moveable: Immediate  SrvMgr: Yes
    IpAddr/PrefixLen: 197.11.221.2/24
      Moveable: Immediate  SrvMgr: No
```

How to rejoin

- **Rejoin can be**
  - Automatic
    - GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN
      - The stack will rejoin the sysplex, when the problem that caused it to automatically leave the sysplex has been relieved.
      - **Note:** Some problem conditions can not be relieved without recycling the TCP/IP stack (e.g. an error that caused the TCP/IP sysplex component to abend)
      - Is only supported in combination with the SYSPLEXMONITOR RECOVERY option (leave the sysplex automatically if a problem is detected)
      - Automatic rejoin is triggered by the events that clear the error condition (XCF links back up, OMPROUTE restarted, etc.)
      - Bounce prevention logic built into the storage condition logic if storage limits are set on GLOBALCONFIG
  - Operator command initiated
    - VARY TCPIP,[stackname],SYSPLEX,JOINGROUP
      - z/OS V1R7: Matching the vary command to leave the sysplex that was introduced in z/OS V1R6
      - Allowing full operator control over when to leave and when to rejoin the sysplex
  - OBEYing a new sysplex configuration into a stack that currently has left the sysplex
    - VARY TCPIP,[stackname],OBEY,DSN=my.sysplex.conf
      - Only supported on z/OS V1R6. In z/OS V1R7, a JOINGROUP command is required for a manual rejoin!
- **Rejoin will work under the same conditions as the initial join**
  - If DELAYJOIN is configured, the stack will ensure OMPROUTE is up and fully functional before the rejoin will take place

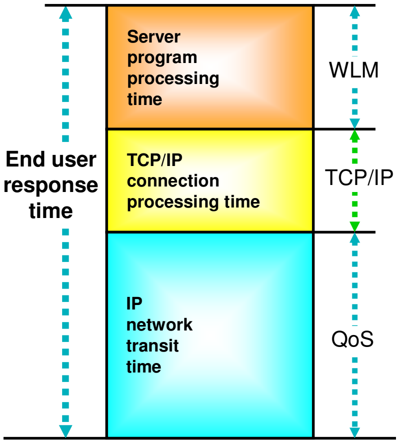
Sysplex Autonomics in z/OS V1R7: Improving the load balancing decisions

Improved workload distribution quality focus in z/OS V1R7

- **Sysplex Distributor uses Server-specific WLM Interfaces to determine if target server is meeting its goal**
  - ƒ Extracts WLM recommendations for each distributed server to determine which server(s) get new connections
  - ƒ More precise than existing WLM method which uses recommendations base upon displaceable capacity of the system

- **Sysplex Distributor will detect target server unresponsiveness**
  - ƒ Target stacks push key TCP/IP "health" statistics for target application(s) to distributor, such as number of connections dropped due to backlog.
  - ƒ When load balancing, the distributor uses these indicators

    along with values for WLM and QoS to determine which stack gets the connection
  - ƒ Strengthens overall evaluation of a server's health



End user response time

Server program processing time — WLM

TCP/IP connection processing time — TCP/IP

IP network transit time — QoS

Adresses some storm-drain scenarios, but not all.

Sysplex Distributor use of WLM and QoS feedback in z/OS V1R7

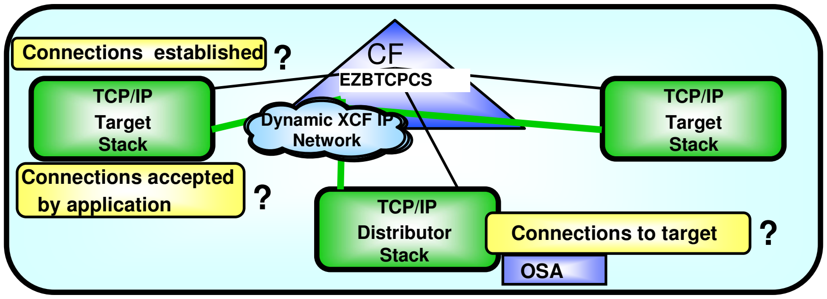
- **Workload Manager feedback has so far been a reflection of how much displaceable capacity the target LPARs have available at any point in time**
  - It has not been a reflection of how well the individual server address space meets its WLM performance goals

- **In z/OS V1R7, WLM will provide new interfaces that will allow Sysplex Distributor to query performance information for individual address spaces**
  - The information from WLM will reflect how well the address space meets its WLM performance goals
    - Base weight is still LPAR displaceable capacity but takes into account the server's WLM Importance Level (i.e. only displaceable cycles below that importance level are counted)
    - If server address space meets its WLM performance goal, WLM will report the LPAR displaceable capacity based weight
    - If server address space does not meet its WLM performance goals, WLM will augment the LPAR displaceable capacity based weight with a fraction that represents how much below the goal this address space currently performs
  - Sysplex Distributor will in z/OS V1R7 can make use of these enhanced WLM interfaces to obtain server-specific WLM recommendations
    - Must be specified on VIPADISTRIBUTE statement

      DISTMETHOD BASEWLM|SERVERWLM|ROUNDROBIN

- **Sysplex Distributor will continue to support modification of the WLM recommendations based on feedback from the Policy Agent about QoS:**
  - Loss ratio, Time-out, Connection limit thresholds

Sysplex Distributor to factor in TCP/IP connection processing performance in z/OS V1R7

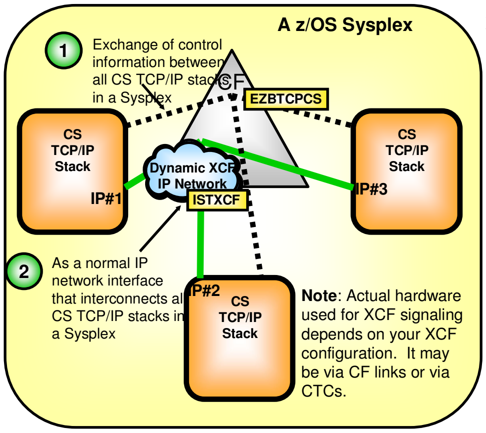


- **Sysplex Distributor will in z/OS V1R7 factor in new weight fractions that reflect how well TCP/IP connection processing is performing:**
  - Lost forwarded connections to the target stack (distributing stack forwards connection request, but doesn't receive a notification that target stack received the connection request)
  - Target stack unable to actually establish a connection with the client (can't complete 3-way TCP handshake)
  - Connections dropped due to server backlog queue full condition
  - A server instance building up a backlog queue while appearing to be "hanging", but not yet dropping connections due to backlog queue full condition

- **SHAREPORT logic will also in z/OS V1R7 be enhanced to factor in how well the individual server instances process new connections**

Improving utilization of XCF Communication resources

z/OS TCP/IP currently requires use of XCF signaling, but what is it used for?


A z/OS Sysplex
1
Exchange of control information between all CS TCP/IP stacks in a Sysplex
CF
EZBTCPCS
CS TCP/IP Stack
IP#1
Dynamic XCF IP Network
ISTXCF
CS TCP/IP Stack
IP#3
2
As a normal IP network interface that interconnects all CS TCP/IP stacks in a Sysplex
IP#2
CS TCP/IP Stack

Note: Actual hardware used for XCF signaling depends on your XCF configuration. It may be via CF links or via CTCs.

In z/OS V1R7, use of dynamic XCF connectivity for Sysplex Distributor and non-disruptive dynamic VIPA movement will be optional.

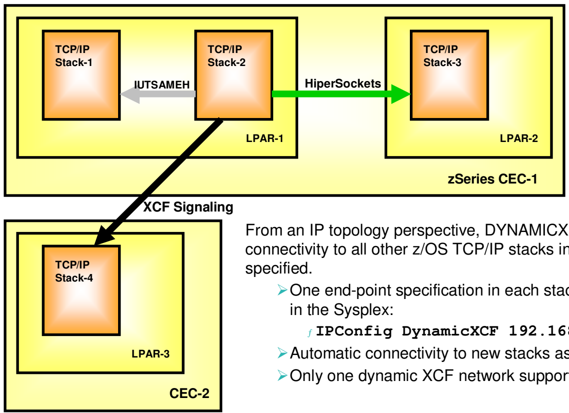
A Dynamic XCF network will therefore be a network connectivity option that can be used or not used depending on local requirements.

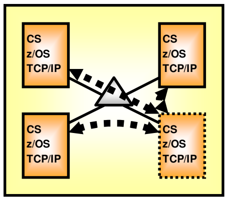
XCF signaling is used for two purposes:

1 When a CS TCP/IP stack starts in a Sysplex, it always joins a predefined XCF group. This group is used by all CS TCP/IP stacks in the same Sysplex to exchange control information over, such as which IP addresses each stack has in its home list and event notification when an IP address is added or deleted. This group is also the group that is used to keep track of which stacks are up and running, so that a stack that is defined as VIPABACKUP for a VIPA address that is active on a stack that goes down can take over the address at the point in time the first stack goes down. There are no configuration controls to enable or disable this use of XCF.

2 XCF can optionally also be used as an IP network interface over which CS TCP/IP stacks can send IP packets to each other. This use is under configuration control and can be defined using either static XCF links or allowing all stacks to join an IP XCF network dynamically (DYNAMICXCF). If one uses Sysplex Distributor or Non-disruptive Dynamic VIPA movement functions in a Sysplex, then dynamic XCF must be enabled.

Is XCF signaling always used for the DYNAMICXCF IP network?





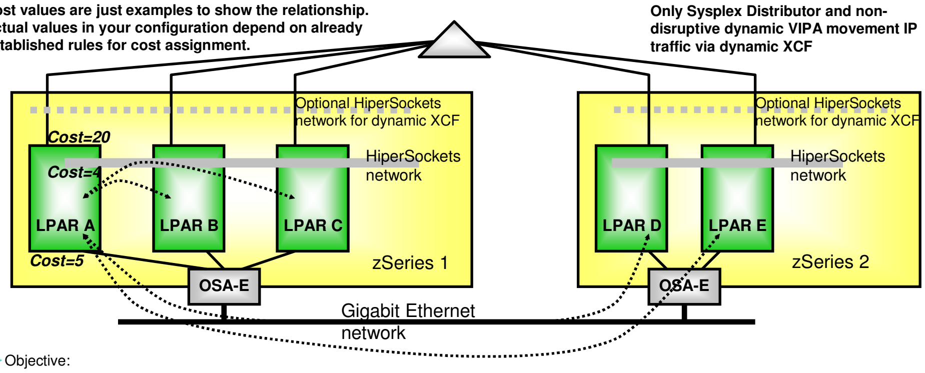
From an IP topology perspective, DYNAMICXCF automatically establishes fully meshed IP connectivity to all other z/OS TCP/IP stacks in the Sysplex that also have DYNAMICXCF specified.

- One end-point specification in each stack for fully meshed connecitivity to all other stacks in the Sysplex:
  - `IPConfig DynamicXCF 192.168.5.1 255.255.255.0 1`
- Automatic connectivity to new stacks as they start up in the Sysplex
- Only one dynamic XCF network supported per Sysplex

Under-the-covers DYNAMICXCF will choose one of three transport technologies depending on availability and location of partner z/OS TCP/IP stack:

- **Inside same LPAR**: IUTSAMEH (memory-link inside a z/OS system)
- **Inside same zSeries CEC**: HiperSockets (if enabled for that purpose via the IQDCHPID VTAM start option)
- **Outside zSeries CEC**: XCF signaling

Guidelines for how to control use of the DynamicXCF IP network for general IP routing - prior to z/OS V1R7



Cost values are just examples to show the relationship. Actual values in your configuration depend on already established rules for cost assignment.

Only Sysplex Distributor and non-disruptive dynamic VIPA movement IP traffic via dynamic XCF

- Objective:
  - Only use dynamic XCF network for the purposes where it at this point in time is required: Sysplex Distributor and non-disruptive dynamic VIPA movement
  - Use a HiperSockets network for IP communication between LPARs in the same CEC
  - Use a gigabit Ethernet infrastructure for IP communication between LPARs in different CECs
- Define the dynamic XCF network with a rather high routing cost so it will not be used for normal IP routing unless it is the only interface that is available - or define it is a non-OSPF interface.
- Define in each CEC a second HiperSockets network (through DEVICE/LINK definitions that interconnects all LPARs in that same CEC) - and use a low routing cost
- Define Gigabit Ethernet connectivity from all LPARs and use a low routing cost (at least one higher than the HiperSockets network)

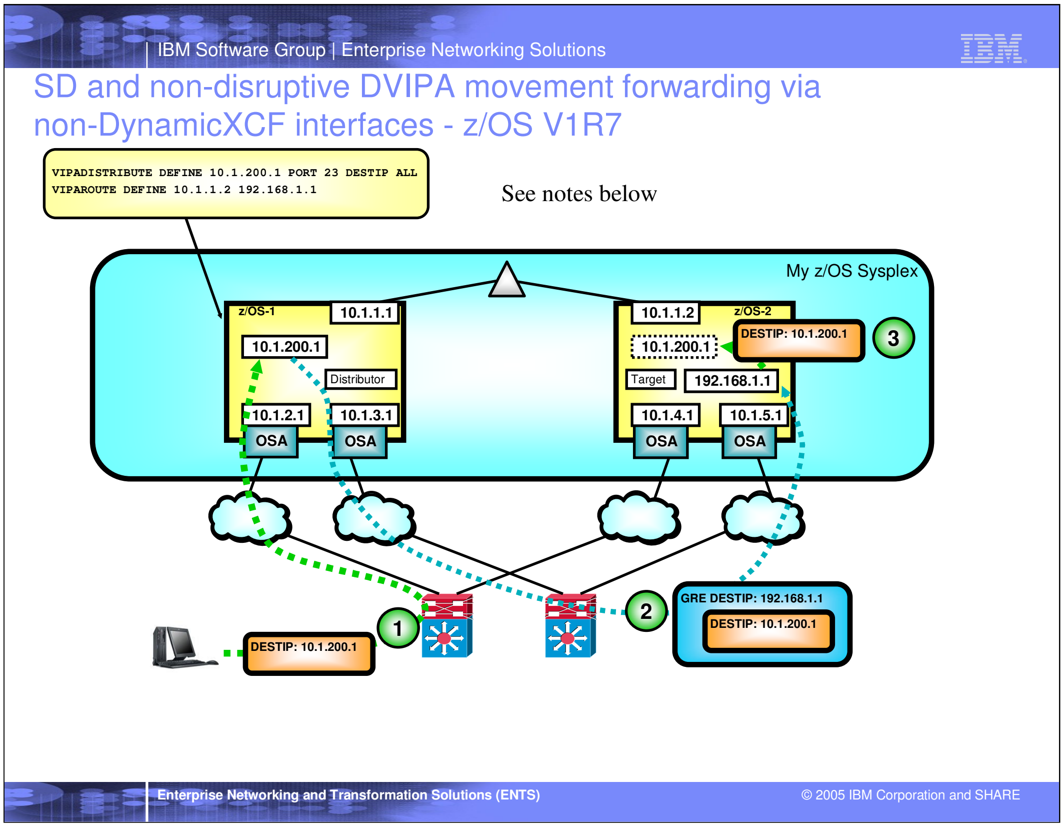
SD and non-disruptive DVIPA movement forwarding of IP packets

- **The reasons why SD and non-disruptive DVIPA movement initially required use of DynamicXCF were:**
  - The forwarding of packets is done without using NAT - the destination address never changes
    - This is known as MAC-level forwarding, or dispatch mode balancing
    - The destination address (the DVIPA) reside in the HOME lists of all stacks that are potential targets
  - This mode of forwarding requires that the destination host is exactly one hop away, or in other words that all members of the z/OS Sysplex are attached to a single shared IP network
    - DynamicXCF was a convenient way to ensure that this requirement was always met with minimal customer configuration requirements

- **Removing the requirement for DynamicXCF means that we cannot guarantee that the target stack we're forwarding a packet to is exactly one hop away**
  - When DynamicXCF is not used, TCP/IP will use GRE (Generic Routing Encapsulation) to forward the packet to a unique IP address on the target stack
  - The address to forward the packet to will be configured using a new configuration option in the VIPADYNAMIC block

      - VIPAROUTE DEFINE dynxcfIPaddress targetIPaddress

  - Whenever SD or non-disruptive DVIPA is to sent a packet to a given DynamicXCF IP address and a VIPAROUTE statement is configured with that DynamicXCF IP address, a GRE envelope will be wrapped around the original packet with the destination IP address from the VIPAROUTE statement and normal IP routing logic will forward that packet (DATAGRAMFWD is *not* required)
    - Path can change based on actual network availability
    - Multipathing is supported
    - High-speed network technologies are available for SD and non-disruptive DVIPA movement forwarding



SD and non-disruptive DVIPA movement forwarding via non-DynamicXCF interfaces - z/OS V1R7

```
VIPADISTRIBUTE DEFINE 10.1.200.1 PORT 23 DESTIP ALL
VIPAROUTE DEFINE 10.1.1.2 192.168.1.1
```

See notes below

PATHMTUDISCOVERY is in general recommended:

a) On remote nodes to learn max packet size (including the GRE hop)

b) On z/OS if the directly connected network is a Gigabit Ethernet network that uses jumbo frames.

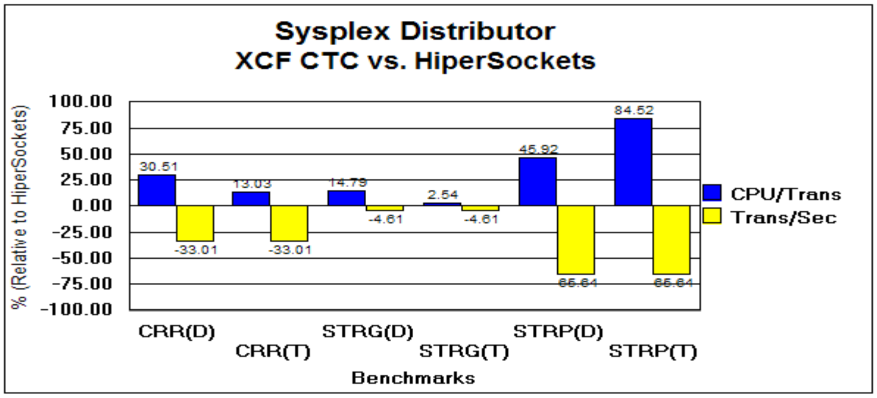
Static VIPA recommended as the target address allows for fault tolerance.

Dynamic XCF will still be used for:

a) Sysplex Wide Security Associations (SWSA)

b) MLS tagged traffic.

Sysplex Distributor XCF vs. HiperSockets

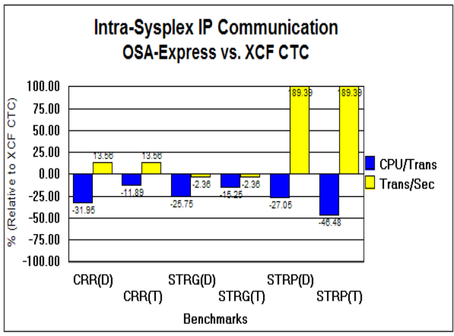


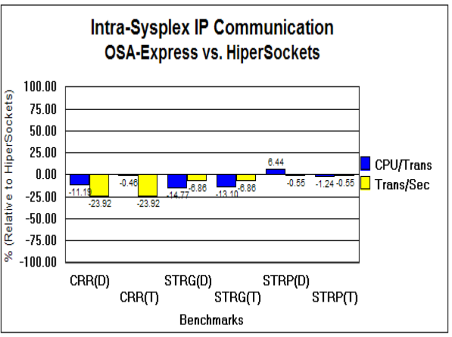
➢**Benchmark Descriptions**

- ƒCRR simulates a connect-request-response workload (i.e. Web traffic)
- ƒSTRG simulates a streaming outbound workload (i.e. FTP GET)
- ƒSTRP simulates a streaming inbound workload (i.e.. FTP PUT)

➢**Offloading IP traffic from XCF to HiperSockets lowers CPU cost and improves throughput**

Intra-Sysplex IP Communication (z/OS V1R7)



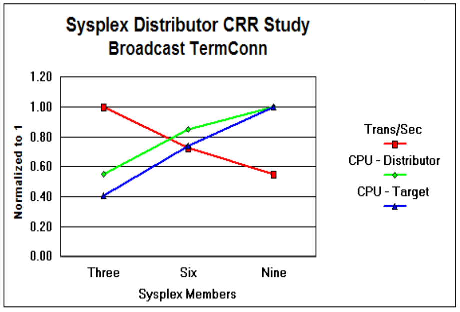


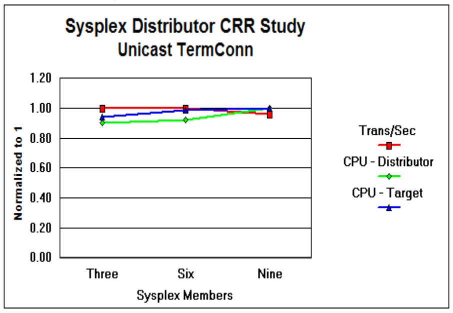
- **Benchmark Descriptions**
  - CRR simulates a connect-request-response workload (i.e.. Web traffic, CICS, IMS)
  - STRG simulates a streaming outbound workload (i.e. FTP GET)
  - STRP simulates a streaming inbound workload (i.e.. FTP PUT)

- **Offloading IP traffic from XCF to OSA-Express lowers CPU cost and generally increases thruput**

- **Offloading IP traffic from HiperSockets to OSA-Express generally lowers CPU cost but decreases thruput**

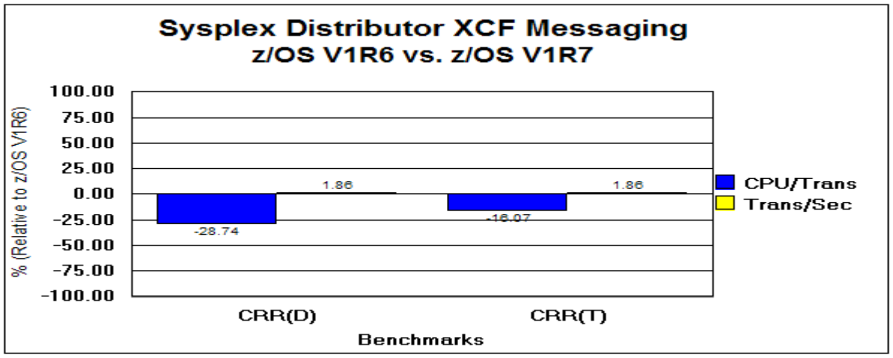
XCF Broadcast vs. Unicast (z/OS V1R6)





- **When connections terminated, a "TERMCONN" signal was broadcast from target TCP/IP stack to all other TCP/IP Sysplex members (allowed for recovery of sessions in cases of failures)**
  - CPU costs of sending/processing broadcast increased as number of Sysplex members increased
- **Enhanced "TERMCONN" signal to unicast directly from target TCP/IP stack to single distributor TCP/IP stack**
  - Alternate provisions made to handle recovery scenarios
- **CRR: Connect/Request/Response type of workloads (i.e. short lived TCP connections) benefit the most**
  - ✓Benefits are automatic! No configuration setup required.
  - ✓Especially for sysplex environments with many members

XCF Message blocking (z/OS V1R7)



- **Each new connection establishment and termination results in a "NEWCONN" signal and "TERMCONN" signal to be sent from the target TCP/IP stack to the distributor TCP/IP stack**
  - Blocking XCF messages on the target TCP/IP stack and sending them as a single "message" to the distributor TCP/IP stack significantly reduces CPU costs
- **Benchmark Descriptions**
  - CRR simulates a connect-request-response workload (i.e. short lived connections like Web, CICS, IMS traffic)
  - These type of workloads benefit most from this improvement (assumes relative high transaction rate)
  - ✓Benefits are automatic! No configuration needed.

Dynamic XCF connectivity for TCP/IP in a Subarea environment

- Current TCP/IP Support for Dynamic XCF Connectivity
  - Exploits VTAM's device layer XCF transport
  - Requires VTAM APPN XCF Connectivity to be configured!

- In z/OS V1R7, it will be possible to utilize the full range of TCP/IP sysplex functions (including XCF communications) without having to redefine the SNA network to use APPN communications, nor having to enable APPN XCF communications.

  - IP communications over XCF can now be enabled on:
    - APPN nodes without having to first establish APPN connections
    - pure Subarea nodes

  - XCFINIT now enabled for subarea nodes
    - special XCFINIT options that allow for IP communications only

Trademarks and notices

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

- AIX7
- AnyNet7
- AS/4007
- Candle7
- CICS7
- CICSPlex7
- CICS/ESA7
- DB27
- DB2 ConnectTM
- DPI7
- DRDA7
- e business(logo)7
- ESCON7
- eServerTM
- ECKDTM
- FFSTTM
- GDDM7
- GDPS7
- HiperSocketsTM
- IBM7
- Infoprint7
- IMSTM
- IP PrintWayTM
- iSeriesTM
- Language Environment7
- MQSeries7
- MVSTM
- MVS/ESATM
- NetView7
- OS/27
- OS/3907
- Parallel Sysplex7
- PrintWayTM
- PR/SMTM
- pSeries7
- RACF7
- RedbooksTM
- Redbooks (logo)TM
- S/3907
- System/3907
- ThinkPad7
- Tivoli7
- Tivoli (logo)7
- VM/ESA7
- VSE/ESATM
- VTAM7
- WebSphere7
- xSeries7
- z/ArchitectureTM
- z/OS7
- z/VM7
- zSeries7

Cisco, Cisco Systems, the Cisco Systems logo, Catalyst, and Cisco IOS are registered trademarks or trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.