CS z/OS V1R7 Enhancements to IP Workload in a z/OS Sysplex: Sysplex Distributor

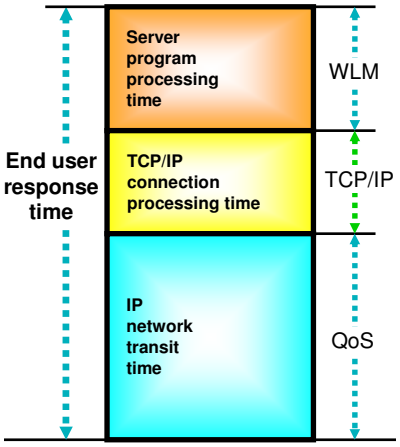
CS z/OS V1R7 Enhancements to IP Workload in a z/OS Sysplex

- **Sysplex Distributor (SD) load balancing decision enhancements**
- **Sysplex Distributor (SD) optimized forwarding of distributed workload to target TCP/IP stacks**

Sysplex Distributor load balancing decision enhancements

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**
  - Extracts WLM recommendations for each distributed server to determine which server(s) get new connections
  - More precise than existing WLM method, which uses recommendations based upon displaceable capacity of the system
- **Sysplex Distributor will detect target server unresponsiveness**
  - Target stacks push key TCP/IP "health" statistics for target application(s) to distributor, such as number of connections dropped due to backlog
  - When load balancing, the distributor uses these indicators along with values for WLM and QoS to determine which stack gets the connection
  - Strengthens overall evaluation of a server's health
- **Allow Sysplex Distributor to route over any available route**
  - Removes the need for Sysplex Distributor to route only over coupling facility links
  - Allows for use of high-speed links such as Ethernet and OSA Express QDIO interfaces
  - Can use any interface on the target except cannot specify a dynamic VIPA



Adresses some storm-drain scenarios, but not all.

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

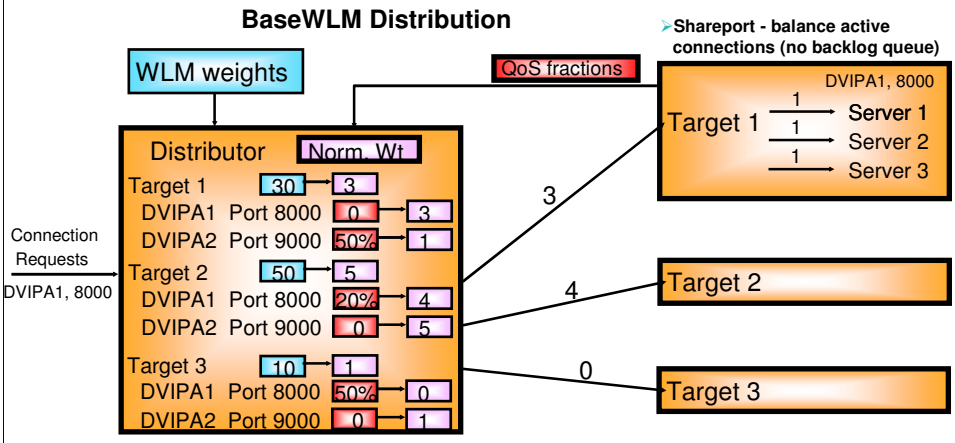
- **Workload Manager feedback has so far been a reflection of how much displaceable capacity the target LPARs have available at any point in time**
  - It has not been a reflection of how well the individual server address space meets its WLM performance goals
- **In z/OS V1R7, WLM provides new interfaces 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
    - The base weight is still LPAR displaceable capacity.
    - If server address space meets its WLM performance goal, WLM will report the LPAR displaceable capacity base weight
    - If server address space does not meet its WLM performance goals, WLM will augment the LPAR displaceable capacity base weight with a fraction that represents how much below the goal this address space currently performs
  - Sysplex Distributor in z/OS V1R7 makes use of these enhanced WLM interfaces to obtain server-specific WLM recommendations
- **Sysplex Distributor continues to support modification of the WLM recommendations based on feedback from the Policy Agent about QoS:**
  - Loss ratio
  - Time-out
  - Connection limit threshold

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

- **Sysplex Distributor in z/OS V1R7 factors 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 in z/OS V1R7 is enhanced to also factor in how well the individual server instances process new connections**

Background information

- **Using the Sysplex Distribution function, incoming connections for a DVIPA and port are distributed to multiple target stacks. Target selection is determined using**
  - RoundRobin
  - BaseWLM - capacity recommendations from WLM (weights) for each system
    - Weights are normalized - optionally modified with policy information (QoS fractions) from each target



- **Shareport - balance active connections (no backlog queue)**

Background notes

NOTES

- When determining a BaseWLM weight, WLM assigns a relative weight to each system in the sysplex with the highest weight going to the system with the most available CPU capacity. The weights range between 0 & 64. If all systems in the Sysplex are running at or near 100% utilization, WLM will assign the highest weights to the systems with the largest amounts of lower importance work. In this way, new connection requests will be distributed to the systems with the highest displaceable capacity.

- Normalizing and determining the QoS modified WLM weight
  - WLM weights are normalized - the WLM weights range in value from 1 to 64. These returned system weights are divided by the smallest system weight. For example, if BaseWLM system weights of 50, 30, and 10 are returned, the normalized weights are be 5, 3, and 1.
  - A QoS Service level fraction is received from the target for each group of connections that map to a DVIPA/PORT for that service level. The fraction represents the performance of this group of connections. This is based on maximum connection limit for the service level, the target-to-client performance (ratio of retransmits and timeouts to number of packets sent, overall throughput and throughput/connection against desired values) - the lower the fraction, the better the performance.
  - The normalized WLM weight is reduced by the QoS Fraction percentage. For example if the normalized WLM weight is 5 and the QoS Fraction is 20%, the modified weight is 4 (5 - (5 * 20%)).

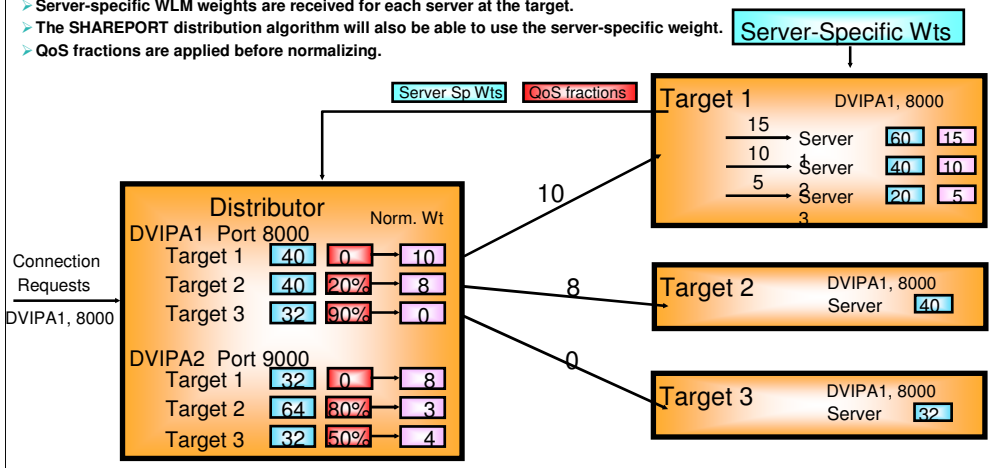
- Distribution of connections to DVIPA1, Port 8000
  - Connections come in destined for DVIPA1, Port 8000.
  - Based on the QoS modifed WLM weights for this DVIPA/Port and service level, as 7 connection requests are received, 3 connections are distributed to Target 1 and 4 connections to Target 2.
  - Target 1 is configured with SHAREPORT for Port 8000. Connections are evenly distributed among the servers that have no backlog queue such that each server has the same number of active connections.

Potential distribution problems

- **The WLM weight is based on a comparison of the available capacity for new work on each system, not how well each server is meeting the goals of its service class.**
- **If all systems are using close to 100% of capacity, then the WLM weight is based on a comparison of displaceable capacity - the amount of lower importance work on each system, but if the service class of the server is of low importance then it may not be able to displace this work.**
- **If a target stack or a server on the target stack is not responsive, new connection requests will continue to be routed to the target - the stack may even appear to be lightly loaded since applications are not processing any new work.**
- **QoS fractions monitor target-to-client performance, but do not monitor the path to the target.**
- **SHAREPRT ensures an even distribution of active connections among servers that have not exceeded their backlog queue limit.**
  - However it does not account for how well each server is meeting the goals of its service class.

Enhanced workload distribution based on server-specific WLM weights

- **WLM will assign a weight based on:**
  - How well a server is meeting the goals of its service class.
  - The displaceable capacity for new work based on the importance of its service class.
- **Server-specific WLM weights are received for each server at the target.**
- **The SHAREPORT distribution algorithm will also be able to use the server-specific weight.**
- **QoS fractions are applied before normalizing.**



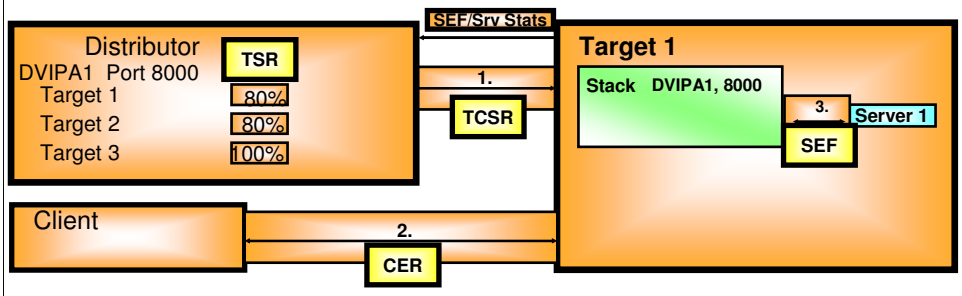
Determining the normalized WLM weight

NOTES

- A Server-specific weight is sent from the target to the distributor for each DVIPA/Port. In the case of multiple Shareport Servers, an average weight is sent to the distributor.
- Determining the QoS modified WLM weight and normalizing - to preserve more of the distinctions between different weights, the QoS fraction is applied to the raw WLM weight before normalizing, and the normalization algorithm is changed. From the previous chart, the weight for Target 2's DVIPA1, Port 8000 Server is calculated as follows:
  - The QoS Service level fraction is applied against the raw WLM weight before the WLM weight is normalized. The WLM weight is 40 and the QoS fraction is 20%, so the QoS modified WLM weight is 32 (40 - (40 * 20%))
  - The normalized weight is 8 - determined by dividing by 4.
  - The exception to this would be if all of the received WLM weights associated with a DVIPA/Port were less than or equal to 16. In that case normalization is not done. After the QoS fraction is applied against the raw weight, the weights are left unchanged.
  - To change a server weight, WLM depends on the server receiving work. So even if a server weight is zero, a connection request will still be forwarded infrequently to that server to generate new WLM values.
- Distribution of connections to DVIPA1, Port 8000
  - Connections come in destined for DVIPA1, Port 8000.
  - Based on the QoS modifed WLM weights for this DVIPA/Port and service level, as 18 connection requests are received, 10 connections are distributed to Target 1 and 8 connections to Target 2.
  - Target 1 is configured with SHAREPORT for Port 8000. As 30 connections are received, 15 will be distributed to server 1, 10 to server 2, and 5 to server 3.

Sysplex autonomics health monitor for target stacks

- **TCSR - Target Connectivity Success Rate**
  - Monitoring connectivity between the distributing stack and the target stack - are the new connection requests reaching the target?
- **CER - Connection Establishment Rate**
  - Monitor network connectivity between server and client - are new connections being established?
- **SEF - Server accept Efficiency Fraction**
  - Monitor Target Server responsiveness - is the server accepting new work?
- **TSR - Target Server Responsive fraction**
  - The target sends SEF values and server statistics to the distributor which creates a Target Server Responsiveness Fraction (TSR) based on the TCSR and SEF (which includes CER).
- **All values are expected to be 100 unless there is a problem.**
  - TCSR dropping to 25 or lower will drive optimized routing function to do a new route lookup.



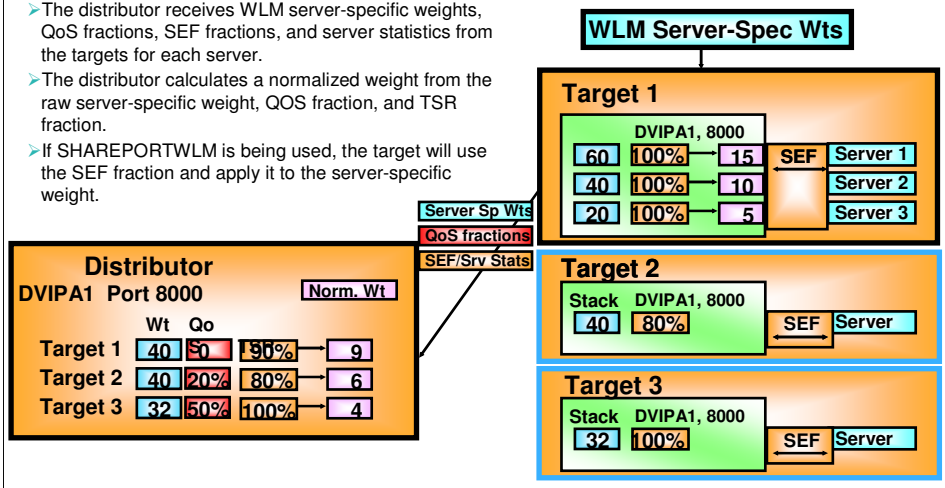
TSR calculations

NOTES

- TCSR - The distributor determines this value from the number of SYN segments it has sent to the target and the statistics returned from the target
- SEF - This value is based on whether the server is processing new connections
  - New connections are being established - Connection Establishment Rate (CER)
  - The server is accepting the new connections
- TSR - Based on SEF value (which includes CER) and TCSR value

Server-specific WLM with Sysplex autonomics health monitor for target stacks

- The distributor receives WLM server-specific weights, QoS fractions, SEF fractions, and server statistics from the targets for each server.
- The distributor calculates a normalized weight from the raw server-specific weight, QOS fraction, and TSR fraction.
- If SHAREPORTWLM is being used, the target will use the SEF fraction and apply it to the server-specific weight.



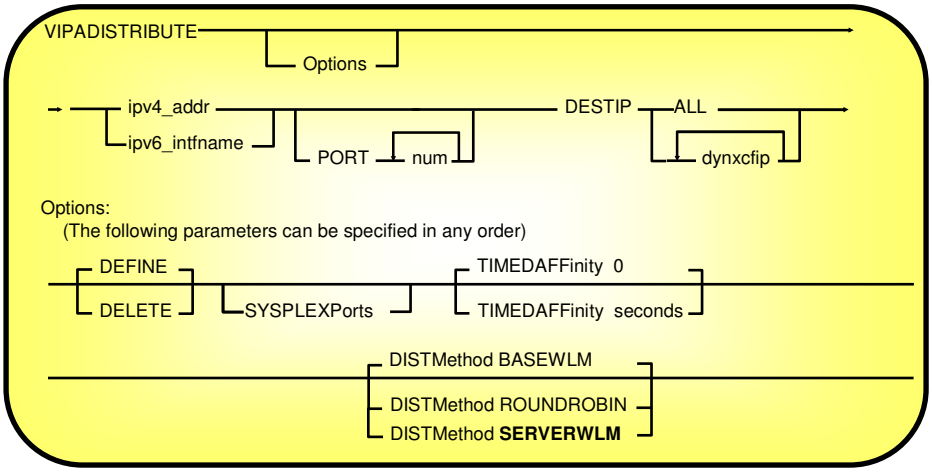
Determining the normalized WLM weight

NOTES

- From the previous chart, the weight for Target 2's DVIPA1, Port 8000 Server is calculated as follows:
  - The QoS Service level fraction is applied against the raw WLM Server weight. For example if the WLM weight is 40 and the QoS fraction is 20%, the QoS modified WLM weight is 32 (40 - (40 * 20%)).
  - A TSR fraction is calculated from the SEF value and server statistics that are received from the target and from information that the distributor keeps for each server. A higher fraction means a healthier server. So if the QoS modified WLM weight is 32, and the Server fraction is 80%, the new modified weight is 25 (32 * 80%).
  - Finally the normalized weight of 6 is determined by dividing by 4.
- SHAREPORTWLM distribution
  - The target calculates an SEF from the statistics for each SHAREPORT server. At the target, the fractions are applied against the raw server weights and normalized (as above).
  - The average of the SEF values and statistics is sent to the distributor.
  - The average of the raw server weights is sent to the distributor.
- SHAREPORT distribution - if the existing SHAREPORT parameter is used, distribution is changed to use the SEF value alone. The SEF value is applied against an assumed raw weight of 64 (the highest weight) and a normalized weight is calculated as above. It is no longer based on balancing the number of active connections among the servers.

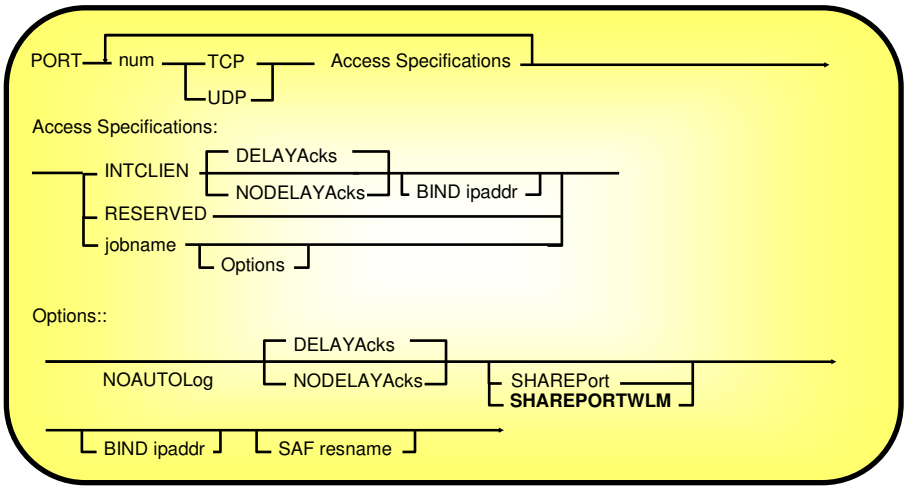
Configuring Sysplex Distribution using server-specific weights

- **Configure the existing parameter, SYSPLEXROUTING, on the IPCONFIG statement on the Distributor stack and each target stack**
- **Configure a new parameter, SERVERWLM, on the VIPADISTRIBUTE statement**



Configuring SHAREPORT distribution to use server-specific weights

- **Configure a new parameter, SHAREPORTWLM, on the PORT statement**
- **SHAREPORTWLM is independent of SERVERWLM**



Netstat VIPA configuration report

- **Netstat VIPADCFG/-F report - display the configured distribution method**

Long Format:

```
VIPA Distribute:
    Dest:        201.2.10.11..8000
      DestXCF:   ALL
        SysPt:   No  TimAff: No     Flg: ServerWLM
    Dest:        201.2.10.12..4011
      DestXCF:   ALL
        SysPt:   No   TimAff: No    Flg: BaseWLM
    Dest:        201.2.10.13..243
      DestXCF:   ALL
        SysPt:   No   TimAff: No    Flg: RoundRobin

Short Format:
VIPA Distribute:
IP Address       Port   XCF Address      SysPt  TimAff  Flg
----------       ----   -----------      -----  ------  ----
201.2.10.11      8000   ALL              Yes    No      S
201.2.10.12      4011   ALL              Yes    No      B
201.2.10.13      243    ALL              No     No      R
```

Netstat VIPA destination port report

- **Netstat VDPT/-O report - display distribution method, TSR values, WLM weights (after TSR adjustment)**

Long Format:

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS
Dynamic VIPA Destination Port Table:
Dest:        201.2.10.11..8000
  DestXCF:   193.9.200.1
  TotalConn: 0000000050  Rdy: 001  WLM: 15  TSR: 100
  Flg: ServerWLM
Dest:        201.2.10.11..8000
  DestXCF:   193.9.200.2
  TotalConn: 0000000050  Rdy: 001  WLM: 15  TSR: 100
  Flg: ServerWLM

Short Format:
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS           15:35:26
Dynamic VIPA Destination Port Table:
Dest IPaddr     DPort DestXCF Addr    Rdy TotalConn  WLM TSR Flg
-----------     ----- ------------    --- ---------  --- --- ---
201.2.10.11     08000 193.9.200.1     001 0000000050 15  100 S
201.2.10.11     08000 193.9.200.2     001 0000000050 15  100 S
```

Netstat VIPA destination port report with the detail option

- **Netstat VDPT/-O DETAIL - display TCSR, CER, and SEF values**

Long Format:

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS
Dynamic VIPA Destination Port Table:
Dest:        201.2.10.11..8000
  DestXCF:   193.9.200.1
  TotalConn: 0000000050  Rdy: 001  WLM: 15  TSR: 100
  Flg: ServerWLM
  TCSR: 100 CER: 100 SEF: 100
  QoSPlcAct:    *DEFAULT*
    W/Q: 15
  QosPlcAct:    Gold-Service
    W/Q: 10
Dest:        201.2.10.11..8000
  DestXCF:   193.9.200.2
  TotalConn: 0000000050  Rdy: 001  WLM: 15  TSR: 100
  Flg: ServerWLM
  TCSR: 100 CER: 100 SEF: 100
  QoSPlcAct:    *DEFAULT*
    W/Q: 15
  QoSPlcAct:    Gold-Service
    W/Q: 15
```

Netstat port list report

- **Netstat PORTList/-o report - An existing flag is used to indicate that port sharing is being used (S). An additional new flag will be used to indicate that port sharing with server-specific weights is being used (W).**

Long Format:

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS           08:58:11
Port# Prot User     Flags    Range
----- ---- ----     -----    -----
08000 TCP  CICS1    DASW
```

Netstat all report

- **Netstat ALL/-A report - If socket is in Listening state**
  - SEF value is displayed
  - If port sharing is being used, then shareport information will show:
    - Type of distribution method being used
      - SHAREPORT - BASE (SEF only)
      - SHAREPORTWLM - WLM (Server-Specific weight and SEF)
    - Server-specific WLM weights shown regardless of distribution method

Long Format:

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP NAME: TCPCS            17:40:36
Client Name: CICS1                    Client Id: 0000004A
Local Socket: 201.2.10.11..8000
       ...
  Last Touched:       14:59:25          State:               Listen
       ...
  ConnectionsIn:      0000000000        ConnectionsDropped: 0000000000
  CurrentBacklog:     0000000000        MaximumBacklog:     0000000010
  CurrentConnections: 0000000300        SEF:                100
  SharePort: WLM
    RawWeight:        40                NormalizedWeight:   10
```

SNMP Dynamic VIPA MIB support for the new functions

NOTES

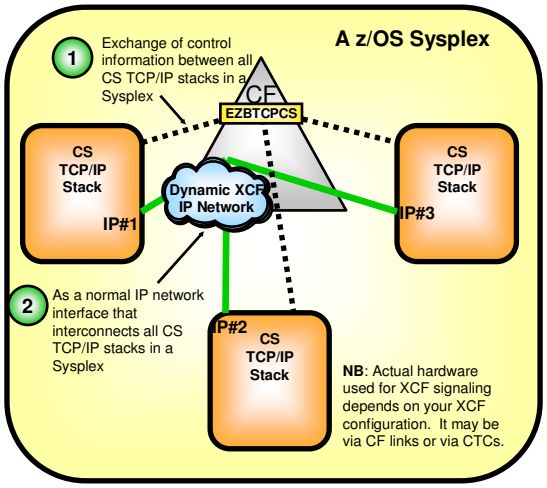
- New MIB objects provide distribution information for a DVIPA/Port
  - ibmMvsDVIPADistConfDistMethod, ibmMvsDVIPADistPortFlag - distribution method used
  - ibmMvsDVIPADistPortTsr - TSR value for a target server
  - ibmMvsDVIPADistPortTcsr - TCSR value for a target server
  - ibmMvsDVIPADistPortSef -SEF value for a target server
  - ibmMvsDVIPADistPortCer - CER value for a target server
- A new MIB object provides information about TCP servers using port sharing
  - ibmMvsPortSharePortWlm

Migration - things to think about

- **WLM server recommendations can only be used if the Sysplex Distributor and all target stacks for a distributed DVIPA port are V1R7 or later.**
  - If all targets for a DVIPA do not provide server-specific weights, then BASEWLM will be used.
- **Target server responsiveness values can only be used if the Sysplex Distributor and all target stacks for a distributed DVIPA port are V1R7 or later.**
  - TSR values can be used with BASEWLM or SERVERWLM weights.

Sysplex Distributor (SD) optimized forwarding of distributed workload to target TCP/IP stacks

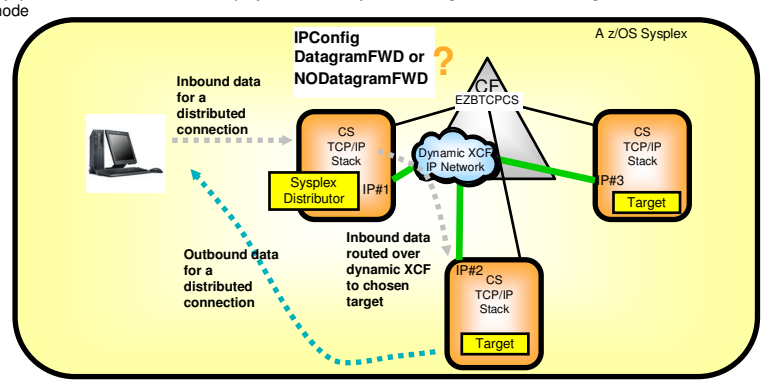
Background information: z/OS TCP/IP requires use of XCF signaling, but what is it used for?



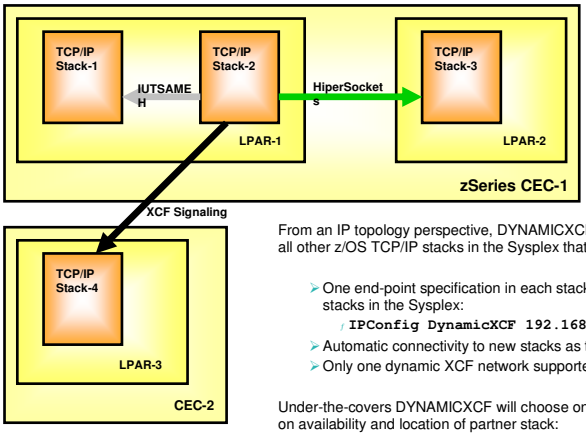
- **When a CS TCP/IP stack starts in a Sysplex, it always joins a predefined XCF group. This group is used by all CS TCP/IP stacks in the same Sysplex to exchange control information over, such as which IP addresses each stack has in its HOME list and event notification when an IP address is added or 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.**
- **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.**

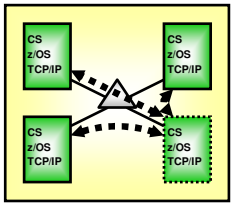
Background information: General IP forwarding no longer required for Sysplex Distributor traffic since z/OS V1R6

- **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



Background information: is XCF signaling always used for the DYNAMICXCF IP network?





From an IP topology perspective, DYNAMICXCF 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 stack:

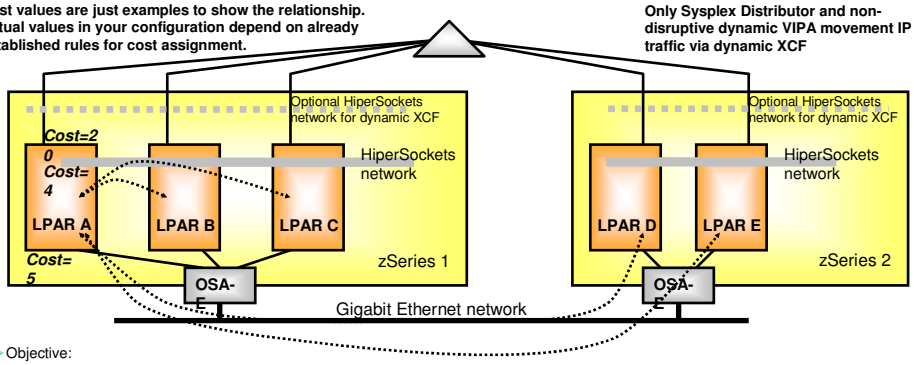
- Inside same LPAR: IUTSAMEH (memory-link inside a z/OS system)
- Inside same zSeries CEC: HiperSockets (if enabled for that purpose via the IQDCHPID VTAM start option)
- Outside CEC: XCF signaling

- **Dynamic XCF network usage:**
  - Sysplex Distributor and non-disruptive DVIPA movement forwarding of packets to target stack
  - General IP routing between the stacks in the Sysplex

Background information: Guidelines for how to control use of the DynamicXCF IP network for general IP routing

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 (recommended).
- 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 over DynamicXCF

- **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 IP address never changes
    - This is known as MAC-level forwarding, or dispatch mode balancing
    - The destination address (the DVIPA) resides 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

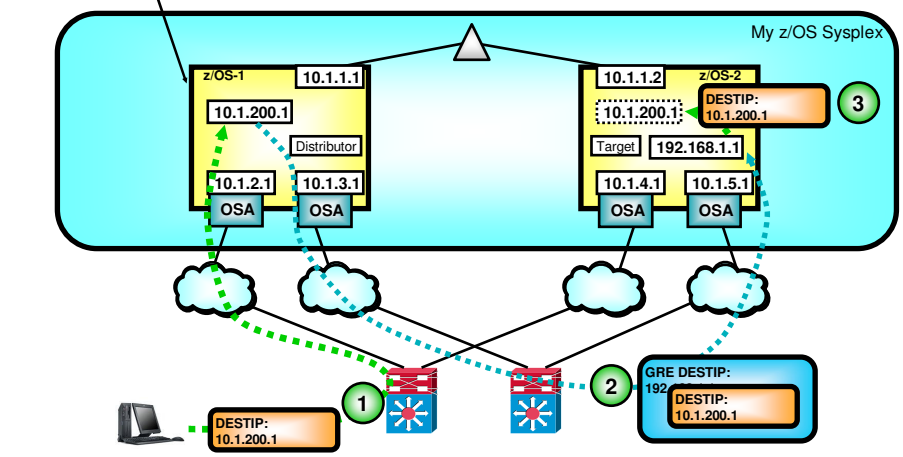
- **Removing the requirement for DynamicXCF means that we cannot guarantee that the target stack we're forwarding a packet to is exactly one hop away**
  - When DynamicXCF is not used, TCP/IP will use GRE (Generic Routing Encapsulation) to forward the packet to a unique IP address on the target stack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

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

PATHMTUDISCOVERY is in general recommended

- On remote nodes to learn max packet size (including the GRE hop)
- On z/OS if the directly connected network is a Gigabit Ethernet network that uses jumbo frames



Details on use of VIPAROUTE

- **Controlled by a new VIPAROUTE statement**
  - Indicates which IP address (target_ipaddr) on the target stack is to be used as the destination IP address during the route lookup selection
  - Used to select a route from a distributing stack to a target stack.
  - Used for distribution to all DVIPAs for which a matching dynamic XCF address, or ALL, was specified on a VIPADISTRIBUTE statement.
  - Used by TAKEOVER stack to forward packets for existing connections to a stack that previously owned the DVIPA (DVIPA in MOVING status).

- **Allows optimal interface to each target, for example:**
  - IUTSAMEH within same MVS image
  - HiperSockets within same CEC
  - OSA Express Gigabit Ethernet between CECs

- **Other considerations**
  - Multipath routing will be supported.
    - Multipath routes used on a per connection basis if the stack has been configured to use any kind of multipath (per connection or per packet).
  - If IP routing tables have changed or Target Connectivity Success Rate (TCSR) is low
    - Sysplex Distributor will perform a new route lookup to retrieve the current best route approximately every 60 seconds.
  - IPv4 uses GRE encapsulation
  - IPv6 uses an outer IP header

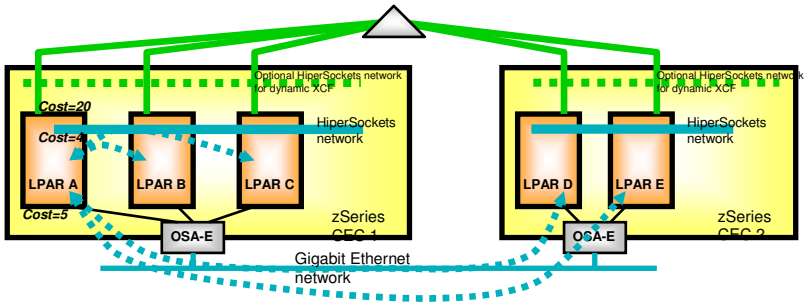
Notes on VIPAROUTE usage

NOTES

- When a connection from the client needs to be processed by Sysplex Distributor, it will determine if a matching VIPAROUTE statement has been specified or not. If it has, the best available route will be determined using the normal IP routing tables. If no matching VIPAROUTE statement exists for that target, IP packets distributed by Sysplex Distributor to that target will use Dynamic XCF interfaces.
- When a VIPAROUTE statement is in effect, packets are sent from the distributor to the target encapsulated in either a GRE wrapper (IPv4) or an IPv6 header. The outer IP header will contain the VIPAROUTE target IP address as its destination IP address and the distributor's dynamic XCF address as the source IP address.
- Generic Routing Encapsulation (GRE) is a standard protocol described by RFC1701. GRE allows a wrapper to be placed around a packet during transmission of the data. A receiving stack that supports GRE will remove the GRE wrapper, allowing the original packet to be processed by the receiving stack. This is often used to deliver a packet to a stack using an alternate destination IP address. For more information regarding GRE, please refer to RFC1701.
- For a pre-V1R7 backup stack, the stack will not be able to process the VIPAROUTE statement.
- For a pre-V1R7 target stack , all IP packets distributed from the routing stack have to be sent over the Dynamic XCF interfaces.
- It is therefore strongly recommended that all TCP/IP stacks participating in VIPAROUTE distribution must be at least z/OS V1R7.

When to use VIPAROUTE statements

- **On distributing stacks**
- **On stacks that are backups for distributing stacks**
- **On stacks that may be used for planned takeovers for non-distributed DVIPAs where the path for forwarding the packets would use XCF links.**
- **Not needed on stacks that are purely target stacks**



It may be desirable to have stacks within the same CEC share VIPAROUTE statements

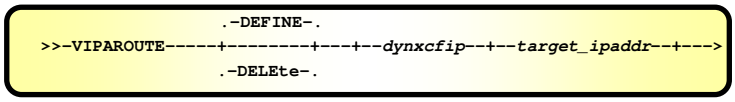
Notes on when to use VIPAROUTE statements

NOTES

- For example, using the previous illustration, if LPAR A is the distributing stack and LPAR B and LPAR D are the target stacks, it would be beneficial to define a VIPAROUTE statement on LPAR A for LPAR D but not LPAR B.
- If LPAR C was the backup for LPAR A, it would be beneficial to use a shared VIPAROUTE definition with LPAR A.
- If LPAR E was the backup for LPAR A, it would not be beneficial to share the VIPAROUTE definitions with LPAR A. LPAR E would want to define a VIPAROUTE statement to LPAR B but not LPAR D.

VIPAROUTE statement in VIPADYNAMIC/ENDVIPADYNAMIC block

- A VIPAROUTE statement is used to define a route from a distributing stack or a backup distributing stack to a target stack



Example:

```
VIPADYNAMIC
        VIPADEF.....
        VIPAROUTE   193.1.3.94    112.112.112.1
        VIPAROUTE   20EC::193:1:3:94   2001::1:2
ENDVIPADYNAMIC
```

Notes on VIPAROUTE statement in VIPADYNAMIC/ENDVIPADYNAMIC block

NOTES

- DEFINE
  - Specifies that the Sysplex Distributor should use the *target_ipaddr* to find the best available route to reach the target stack defined by the *dynxcfip*.
- DELEte
  - Specifies that a previously defined VIPAROUTE statement should be deleted. Sysplex Distributor processing for the target stack specified by the *dynxcfip* will revert to using Dynamic XCF interfaces for existing and new connections after approximately 60 seconds.
- *dynxcfip*
  - Specifies the IPv4 or IPv6 Dynamic XCF address that uniquely identifies a target stack. The address is defined with IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF of that target stack.
- *target_ipaddr*
  - Specifies any fully qualified IPv4 address (in dotted-decimal format) or fully qualified IPv6 address (in colon-hexadecimal format) in the HOME list of the target stack except for a Dynamic VIPA (DVIPA) or a loopback address. It is a static VIPA, a dynamic XCF address, or a real IPv4/IPv6 address associated with a physical interface. Static VIPA addresses are recommended.
- To change the current configured statement, you must specify the VIPAROUTE DELETE with the same *dynxcfip* and the same *target_ipaddr* first, and then specify the VIPAROUTE DEFINE with the same *dynxcfip* and the different *target_ipaddr* in a configuration data set on a VARY TCPIP,,OBEYFILE command.
- If the VIPAROUTE is changed, it will affect active as well as new connections.

Netstat VIPA configuration report (VIPADCFG)

- **Display additional information for VIPAROUTE statement**
- **Support the filter function (IPAddr/I) to allow users to display only the information related to a specific DVIPA or dynamic XCF address**

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS           15:51:43
Dynamic VIPA Information:
VIPA Define:
  IpAddr/PrefixLen: 103.1.1.94/24
.
VIPA Distribute:
  Dest:         103.1.1.94..701
    DestXCF:    ALL
.
VIPA Route:
  DestXCF:      193.1.3.94
    TargetIp:   9.33.113.3
  DestXCF:      193.1.4.94
    TargetIp:   9.44.114.4
  DestXCF:      2ec0::943:f003
    TargetIp:   2ec0::943:f113
  DestXCF:      2ec0::943:f004
    TargetIp:   2000::4:4
```

Netstat VIPA dynamic report (VIPADYN)

- **Display additional information for VIPAROUTE status**
- **Add optional modifier (DVIPA | VIPAROUTE) to display the current dynamic VIPA information only or the current VIPAROUTE information only.**
  - If no modifier is specified, both dynamic VIPA and VIPAROUTE information will be shown

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS        11:24:56
Dynamic VIPA:
  IpAddr/PrefixLen: 103.1.1.94/28
    Status: Active     Origin: VIPADefine       DistStat: Dist/Dest
.
.
VIPA Route:
  DestXCF: 193.38.2.2
    TargetIp: 213.38.1.2
    RtStatus: Active
  DestXCF: c1::38:2:2
    TargetIp: d5::38:1:2
    RtStatus: Active
```

Notes on Netstat VIPA dynamic report (VIPADYN)

NOTES

- **RtStatus: Active**
  - The local stack will forward DVIPA packets to the target stack using normal IP routing table to determine the best available route.
  - Active: indicates that the target stack that is identified by XCF Address or DestXCF is active and TargetIp is defined at that target stack, and at least one route is available to TargetIP. The local stack will forward DVIPA packets to the target stack using normal IP routing table to determine the best available route.
- **RtStatus: Defined**
  - The local stack will forward DVIPA packets to the target stack using dynamic XCF interfaces.
  - Defined: indicates that the target stack which is identified by XCF Address or DestXCF is not active.
- **RtStatus: Inactive**
  - The local stack cannot forward any DVIPA packets to the target stack.
  - Inactive: indicates that the target stack that is identified by XCF Address or DestXCF is active, and TargetIp is defined at that target stack, however no route is available to TargetIp. As a result, the local stack cannot forward any DVIPA packets to the target stack.
- **RtStatus: Unavail**
  - The local stack will forward DVIPA packets to the target stack using dynamic XCF interfaces.
  - Unavail: indicates that the target stack that is identified by XCF Address or DestXCF is active, but TargetIp is not defined at that target stack. The local stack will forward DVIPA packets to the target stack using dynamic XCF interfaces.

Netstat connection route table detail report (VCRT)

- **Display additional routing information when VIPAROUTE profile statements have been configured to the stack**

```
MVS TCP/IP NETSTAT CS V1R7       TCPIP Name: TCPCS           11:17:34
Dynamic VIPA Connection Routing Table:
Dest:      203.38.1.1..801
  Source:  192.168.2.76..1037
  DestXCF: 193.35.1.1
    PolicyRule:    *NONE*
    PolicyAction:  *NONE*
    Intf:  EZAXCFI3
      VipaRoute:  No        Gw:  0.0.0.0
Dest:      203.38.1.1..801
  Source:  192.168.2.76..1036
  DestXCF: 193.38.2.2
    PolicyRule:    *NONE*
    PolicyAction:  *NONE*
    Intf:  LTRLE1A
      VipaRoute:  Yes       Gw:  213.116.38.1
         .
         .
```

VIPAROUTE impact to Sysplex Sockets

- **What is "Sysplex Sockets"?**
  - TCP sockets applications may benefit from knowing when the partner is in either the same MVS image or the same Sysplex.
    - When partners are in the same MVS image, for example, they can share information, such as security contexts, that is otherwise costly to generate.
    - When both partners are in the same Sysplex and communication is through a link that is not exposed outside the Sysplex, applications can provide security without costly encryption or decryption of exchanged packets.
  - The socket option SO_CLUSTERCONNTYPE on getsockopt() allows sockets applications to interrogate the hosting stack about the partner application and to determine whether the partner is in the same Sysplex, the same MVS image or internal.
- **Internal indicator, requested using the SO_CLUSTERCONNTYPE option will no longer be set if the destination IP address (partner's IP address) for a connection is a Dynamic VIPA or Distributed Dynamic VIPA residing in the Sysplex.**
  - Traffic destined to these IP addresses can now be forwarded to the target TCP/IP stacks over links or interfaces that are external to the Sysplex.

Dynamic VIPA MIB enhancements

NOTES

- **Add a new MIB object in the existing ibmMvsDVIPAConnRoute**
  - Provides information about the routes used for distributed connections
- **Add a new MIB table, ibmMvsDVIPARouteTable**
  - Provides information about the VIPAROUTE information. Each entry in this table represents a VIPAROUTE Profile statement

Optimized inter-LPAR messaging for Sysplex Distributor

- **Batch XCF messages that are sent from the target to the distributor**
- **Significant improvement in CPU utilization for:**
  - Heavy transaction workloads
    - Large number of short-lived connections, such as web traffic
  - Distributor in particular, but also for target
- **Automatically enabled (no externals)**
- **Both target and distributor must be z/OS CS V1R7 to get this function**

Performance impacts of optimized Sysplex routing

- **Streams workload - remote get processing (getting a file from z/OS)**

Connectivity	Trans / Second	Trans/Sec Delta %	CPU / Tran (SysDist)	CPU/Tran Delta % (Sys Dist)	CPU / Tran (Targets)	CPU/Tran Delta % (Targets)
XCF	3.0191	Base	82410	Base	89100	Base
OSAE-GbE	2.9480	- 2.4 %	61190	- 25.7 %	75510	- 15.3 %
IQDIO	3.1650	+ 4.8 %	71790	- 12.9 %	86890	- 2.5 %

- **Streams workload - remote put processing (moving a file to z/OS)**

Connectivity	Trans / Second	Trans/Sec Delta %	CPU / Tran (SysDist)	CPU/Tran Delta % (Sys Dist)	CPU / Tran (Targets)	CPU/Tran Delta % (Targets)
XCF	0.9108	Base	305700	Base	267000	Base
OSAE-GbE	2.6358	+ 189.4 %	223000	- 27.1 %	142900	- 46.5 %
IQDIO	2.6505	+ 191.0 %	209500	- 31.5 %	144700	- 45.8 %

Performance impacts of optimized Sysplex routing *(continued)*

- **Transactional workload - connect, request, response, close (CRR)**

Connect -ivity	Trans / Second	Trans/Sec Delta %	CPU / Tran (SysDist)	CPU/Tran Delta % (Sys Dist)	CPU / Tran (Targets)	CPU/Tran Delta % (Targets)
XCF	0.9108	Base	305700	Base	267000	Base
OSAE-GbE	2.6358	+ 189.4 %	223000	- 27.1 %	142900	- 46.5 %
IQDIO	2.6505	+ 191.0 %	209500	- 31.5 %	144700	- 45.8 %

Migration - things to think about

- **DynamicXCF must still be defined in z/OS V1R7:**
  - Target address for VIPADISTRIBUTE definitions is dynamic XCF IP address of target stacks
  - Some workload will still be routed via DynamicXCF:
    - Sysplex Wide Security Association (IPSec) packets
    - Multi Level Security (MLS) tagged packets
    - Policy Agent QoS performance data collection
  - To minimize XCF signalling, use HiperSockets for same-CEC DynamicXCF
    - Actual XCF (CF-links) will only be used for cross-CEC communication
- **Applications using the SO_CLUSTERCONNTYPE option on the GETSOCKOPT socket API**
  - Applications exploiting this internal indicator should continue to function properly from a communications perspective, but they may no longer optimize their processing when the destination address being used is a Dynamic VIPA or a Distributed Dynamic VIPA.
  - If you have applications that exploit this socket option with Dynamic VIPAs or Distributed Dynamic VIPAs, you should consider modifying the configuration to use Static VIPAs as the destination addresses.

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/390	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.

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.