z/OS^7 CS - SHARE in Long Beach, CA - February 2004

Session 3936 - Networking - TCP/IP

Using Virtual IP Addressing for High Avilability and Application Workload Balancing on z/OS

Friday 27-Feb-2004 - 8:00 AM

Enterprise Networking Solutions, Raleigh
Alfred B Christensen - alfredch@us.ibm.com

Using Virtual IP Addressing for High Avilability and Application Workload Balancing on z/OS

Session Number:	3936
Date:	Friday, 27-Feb-2004
Time:	8:00 AM
Location:	Convention center, Concourse level, Seaside B
Speaker:	Alfred B Christensen, IBM
Chair:	Alfred B Christensen, IBM
Abstract:	In this session, we will focus on IP deployment issues within a z/OS Sysplex with special emphasis on IP traffic and application workload balancing and availability. The session will include such topics as intra-Sysplex traffic routing (Dynamic XCF, HiperSockets, or shared Gigabit Ethernet LAN), application availability issues (static VIPA, dynamic VIPA. and Distributed Dynamic VIPA), and IP traffic load-balancing (multipathing). Also discussed are application workload load-balancing technologies such as Sysplex Distributor which is internal to the Sysplex, and technologies that reside on switch equipment of various types outside the Sysplex.

Agenda

TCP/IP and use of Sysplex technologies

IP traffic balancing

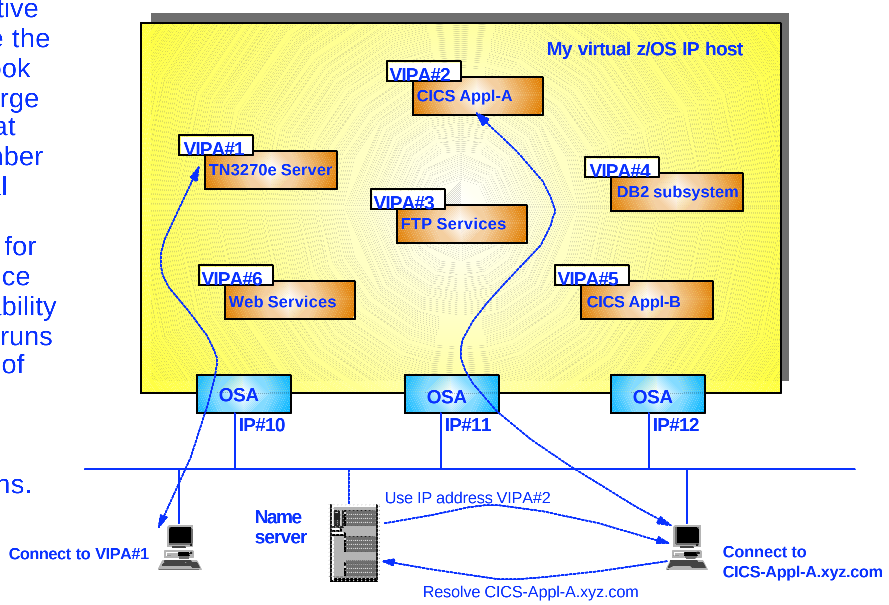
The Virtual IP Address concept

Incoming TCP workload balancing

Subsystem use of TCP/IP high-availability functions

The view of a typical (large) IP host
many network interfaces, many services

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

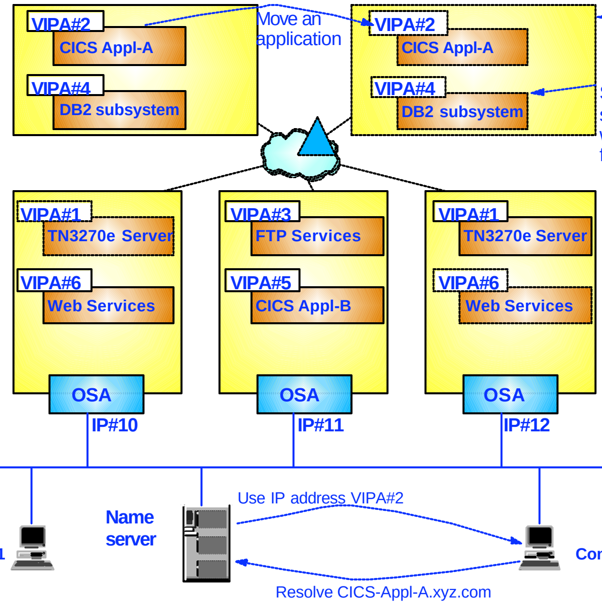


A typical (large) IP host = my Sysplex

e-business

Not all LPARs need to have a physical network interface (an OSA adapter). LPARs can communicate with each other using XCF or HiperSockets between LPARs inside the same z900 box.

The general recommendation is to have all LPARs share OSA adapters for performance reasons.



Add another LPAR to the Sysplex

Start a second DB2 subsystem to share the workload and back up the first DB2 subsystem

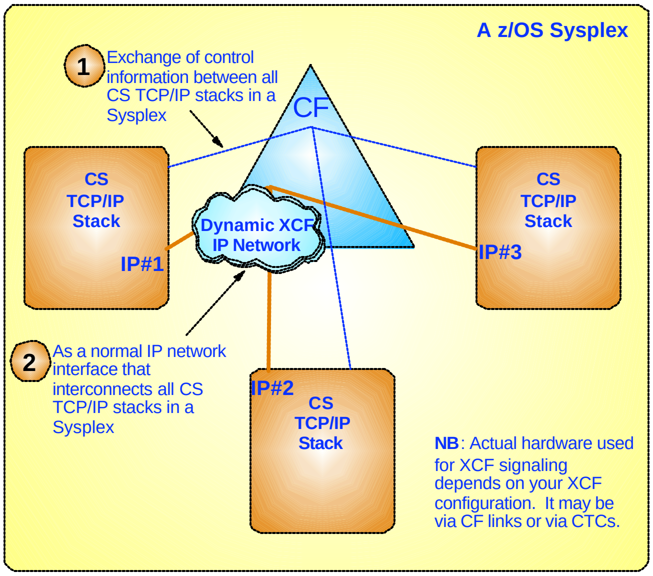
Some servers are duplicated for performance and availability (TN3270e and Web Services in this example), but that is transparent to client hosts.

Connect to VIPA#1

Connect to CICS-Appl-A.xyz.com

What does TCP/IP actually use XCF signaling for in the Sysplex?

e-business



Exchange of control information between all CS TCP/IP stacks in a Sysplex

As a normal IP network interface that interconnects all CS TCP/IP stacks in a Sysplex

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

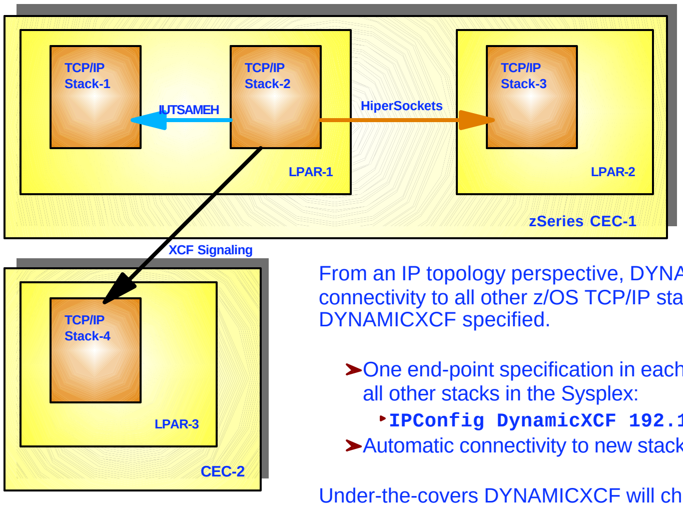
XCF signaling is used for two purposes:

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?

e-business



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

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

- Inside same LPAR: IUTSAMEH
- Inside same zSeries CEC: HiperSockets (if enabled for that purpose via the IQDCHPID VTAM start option)
- Outside CEC: XCF signaling

When do you need SYSPLEXROUTING and DYNAMICXCF in your CS z/OS TCP/IP configuration?

e-business

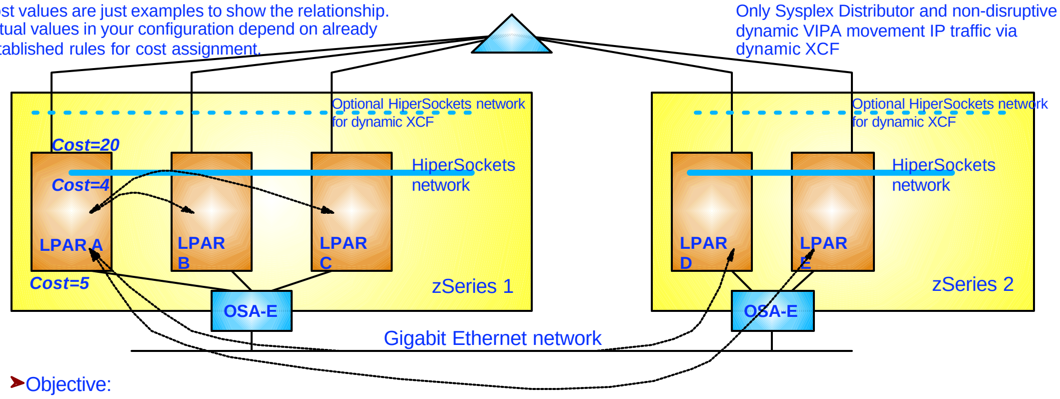
	SYSPLEXROUTING	DYNAMICXCF
What is it good for?	Enables the TCP/IP stack to interface with WLM	Enables the TCP/IP stack to use XCF signaling to transport IP packets to other TCP/IP stacks in a Sysplex.
DNS in WLM Mode on z/OS	Must be enabled	Optional
Dynamic VIPA functions	Optional	Optional (Must be enabled for non-disruptive movement of DVIPAs)
Sysplex Distributor functions	Optional (Must be enabled for WLM-based balancing)	Must be enabled

Sysplex Distributor in z/OS V1R5 adds a round-robin connection distribution method. If you need to disable WLM impact on SD's decision before then, you may try to not specify SYSPLEXROUTING on the SD stack.

Intra Sysplex IP communication guidelines

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 is required: Sysplex Distributor and non-disruptive dynamic VIPA movement
  - Use a HiperSockets network for IP communication between LPARs in the same "box"
  - Use a gigabit Ethernet infrastructure for IP communication between LPARs in different "boxes"
- 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 box a second HiperSockets network (through DEVICE/LINK definitions that interconnects all LPARs in that same box) - 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)

Network choice - notes

NOTES

- When the DynamicXCF interface isn't a preferred route due to a high cost or isn't defined as an interface OSPF knows about, it will not be chosen for normal transmission of IP packets between z/OS images that are connected to the DynamicXCF network, ***but it will still be used for the purposes for which it is required: Sysplex Distributor connection data forwarding and non-disruptive dynamic VIPA connection data forwarding.***
- DynamicXCF may span multiple physical zSeries boxes - but is limited to one z/OS Sysplex.
- HiperSockets may span multiple z/OS Sysplexes, Linux for zSeries, and zVM - but is limited to one physical zSeries box.
- A shared physical network, such as a Gigabit Ethernet network, may span everything that can be hooked up to it, including non-zSeries boxes and operating systems.
- You will most likely want to mix and match and you can rest assured that you will be making changes to this physical networking structure over time, which is one of the main reasons why you want to separate your application identities from the physical network structure by allocating virtual IP addresses to your applications.

Load-balancing outbound IP packets over multiple first-hop routers (MULTIPATH)

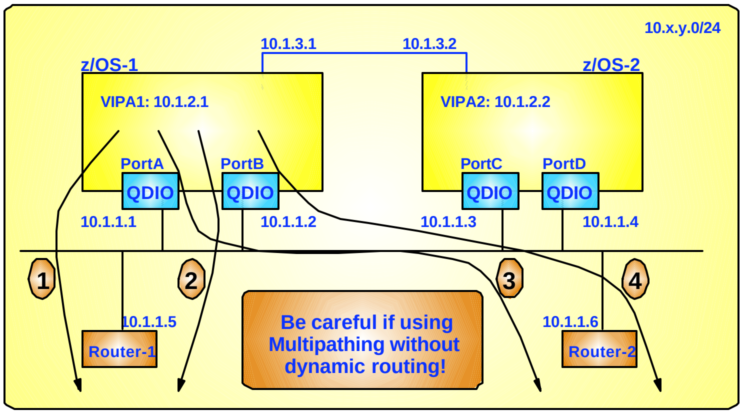
e-business

IPCONFIG MultiPath [PerConnection or PerPacket]

z/OS-1's IP Routing Table (extract)

Destination	Via
10.1.1.0/24	Direct delivery
Default	10.1.1.5 / PortA
Default	10.1.1.5 / PortB
Default	10.1.1.6 / Port A
Default	10.1.1.6 / Port B

z/OS V1R5 raises number of dynamic multipath routes from 4 to 16.



Be careful if using Multipathing without dynamic routing!

Static route definitions:

- If an adapter fails in such a way that z/OS TCP/IP gets informed, it will skip over the corresponding entries from the routing table
- If one of the first-hop routers loses its connection to the backbone network or if it "dies" - z/OS TCP/IP doesn't know anything about it since it doesn't participate in dynamic routing updates - and it will continue to attempt to use the corresponding routing table entries - connections will time out, UDP packets will be lost, etc.

Dynamic routing updates:

- z/OS TCP/IP will know both if the adapter itself fails or if the first-hop router fails - and dynamically update the routing table entries and recover from the router outage..

NB: Inbound load-balancing is the responsibility of Router-1 and Router-2 in this example. z/OS-1 will advertise dynamic routes to the routers, so they can use both the 10.1.1.1 and the 10.1.1.2 interface for sending IP packets to z/OS-1 - but it is a router responsibility to use that information for load-balancing inbound IP packets to z/OS-1 over the two interfaces.

Why do I need virtual IP addresses (VIPA)?

e-business

What does the virtual IP addressing (VIPA) technology promise?

Interface resilience:

- Communication with a server host is unaffected by server physical network interface failures. As long as just a single physical network interface is available and operational on a server host, communication with applications on the server host will persist.

Application access independent of network topology:

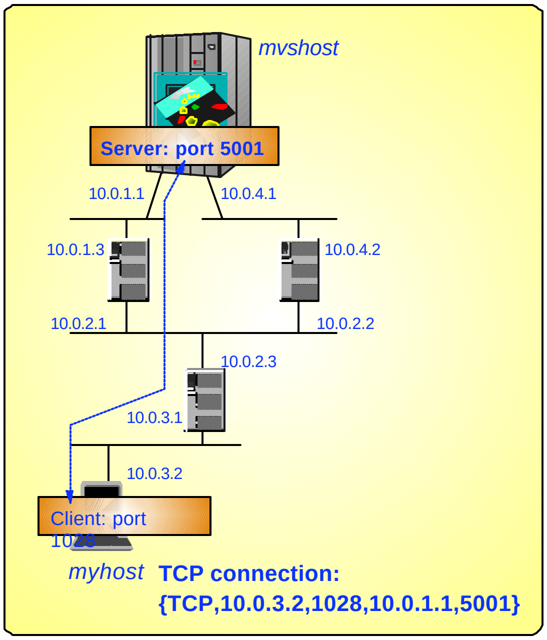
- Separates network topology from server application topology - a VIPA address can be used to identify a server application instead of a physical network interface.
- Allows network administrators to renumber physical network topology
  - no impact to end-user accessing server applications by IP address
  - no changes needed in DNS or hosts file configuration
  - no impact to firewall filtering rules

Single system image:

- Allows the Sysplex to be perceived as a single large server node, where VIPA addresses identify applications independently of which images in the Sysplex the server applications execute on.
- Applications retain their identity when moved between images in a Sysplex.
- Multiple instances of a server application can be accessed as one server.

Inbound TCP connection recovery without VIPA
Notes

NOTES



TCP connection:
{TCP,10.0.3.2,1028,10.0.1.1,5001}

Name server:
mvshost: 10.0.1.1 and 10.0.4.1

- 10.0.1.1 interface on *mvshost* fails:

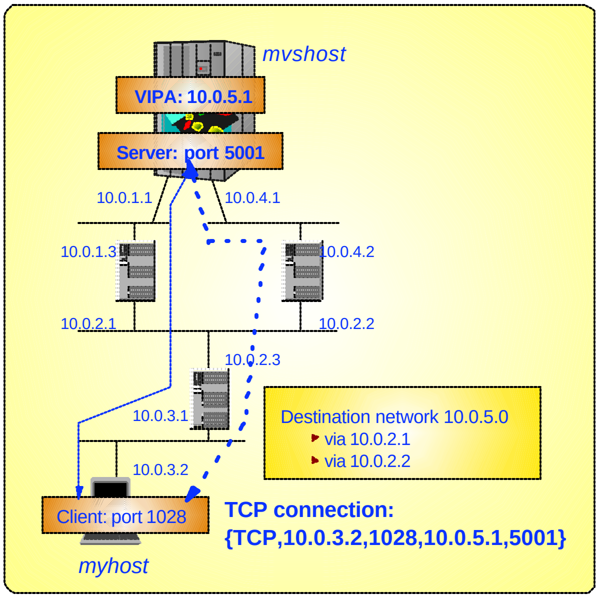
  1. TCP layer on *myhost* times out
  2. TCP layer on *myhost* retransmits
  3. All TCP retransmits fail
  4. TCP connection times out and breaks
  5. Client manually establishes new connection to 10.0.4.2

- The router at 10.0.3.1 does not update its routing table entry for the 10.0.1.0 subnet; that subnet is still reachable through the 10.0.2.1 router.

Subnet mask:
255.255.255.0

Inbound TCP connections using z/OS VIPAs survive a network interface failure on z/OS
Notes

NOTES



Destination network 10.0.5.0
- via 10.0.2.1
- via 10.0.2.2

TCP connection:
{TCP,10.0.3.2,1028,10.0.5.1,5001}

Name server:
mvshost: 10.0.5.1

- 10.0.1.1 interface on *mvshost* fails:

  1. TCP layer on *myhost* times out
  2. TCP layer on *myhost* retransmits
  3. Router at 10.0.3.1 accepts a new route to the 10.0.5.1 network via 10.0.2.2
  4. TCP layer retransmissions succeed

- Routing tables on myhost did not have to be updated.

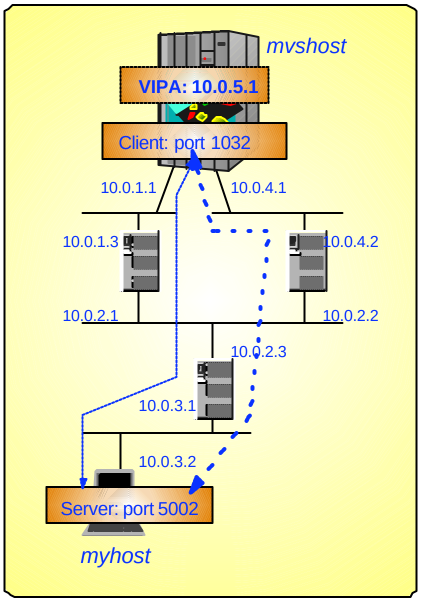
- Routers' routing tables must be updated before TCP times out the connection (can be a concern with RIP dynamic routing protocols).

Subnet mask:
255.255.255.0

Outbound TCP connections surviving a network interface failure on z/OS (SourceVIPA) Notes

e-business

NOTES



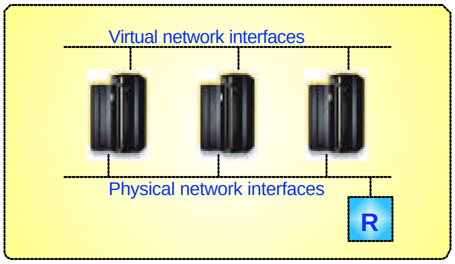
HOME LIST in mvshost:
10.0.5.1 VIPALINK
10.0.1.1 INTFA
10.0.4.1 INTFB

- Client application on *mvshost* connects to server application on *myhost*
- Default behavior:
  - Local IP address of socket is chosen based on the real interface over which the SYN segment is sent - in this case INTFA. The TCP connection will be based on **{TCP,10.0.1.1,1032,10.0.3.2,5002}** and is not recoverable.
- With **SOURCEVIPA** specified:
  - Local address of socket is chosen based on most recent VIPA link in the HOME list for the real interface over which the SYN segment is sent: **{TCP,10.0.5.1,1032,10.0.3.2,5002}** and the connection is recoverable.

Subnet mask:
255.255.255.0

Which IP addresses should I use as Virtual IP Addresses?

e-business



You can pictorially think of the VIPA addresses as addresses that belong to a non-existing network behind the z/OS images on which the VIPA addresses are defined.

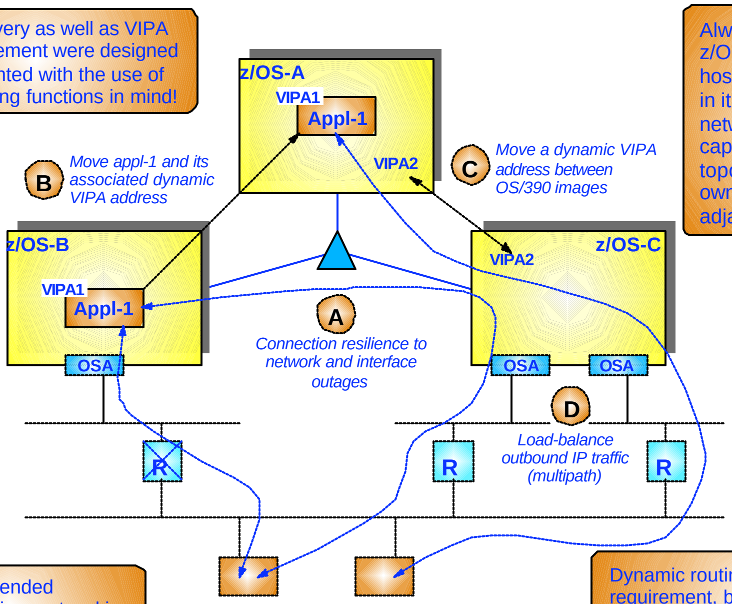
APAR PQ82792 will prevent OMPROUTE from including the VIPA subnets in OSPF advertisements! (PTFs for z/OS V1R4 and V1R5).

- A VIPA address is a normal IPv4 or IPv6 address. It has a link or interface name and it is included in the HOME list of a z/OS TCP/IP stack.
- A z/OS node may have multiple VIPA addresses defined.
- VIPA addresses in a Sysplex may all come out of one and the same subnet - or each member in the Sysplex may initially be assigned individual subnets
- When dynamic VIPA addresses are used in a Sysplex, don't expect to be able to maintain one subnet per z/OS image - since the addresses move around between z/OS images (unless you define one address per subnet)
- You may use multiple subnets for VIPA addresses in a Sysplex; they do not need all to come out of the same subnet.
- A physical network interface and a VIPA interface should not use the same subnet if dynamic IP routing is enabled.
- IPv4 VIPA addresses must be advertised by the z/OS routing daemons as 32-bit prefix destinations (host routes). RIP requires the -h flag to do so, OSPF always does so by default.

Is dynamic routing protocols required on z/OS in order to use VIPA?

Base IP recovery as well as VIPA address movement were designed and implemented with the use of dynamic routing functions in mind!

Always remember that a z/OS Sysplex is not a host, it is an IP network in itself and as any IP network it needs the capability to react to topology changes in its own network and in the adjacent networks.


z/OS-A
VIPA1
Appl-1
VIPA2
B
Move appl-1 and its associated dynamic VIPA address
C
Move a dynamic VIPA address between OS/390 images
z/OS-B
VIPA1
Appl-1
OSA
z/OS-C
VIPA2
OSA
OSA
A
Connection resilience to network and interface outages
D
Load-balance outbound IP traffic (multipath)
R
R
R

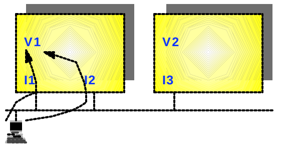
Connect to VIPA-1 (Appl-1)

The recommended dynamic routing protocol in the Sysplex is **OSPF**.

Dynamic routing is not an absolute requirement, but it is ***highly recommended*** when using VIPA addresses (it makes life a whole lot easier)!

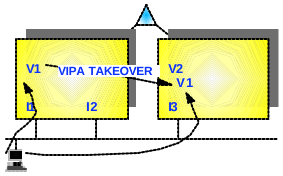
The evolution of the VIPA technology: static VIPA - dynamic VIPA

TCP/IP V3R2 Static VIPA Support



- VIPA addresses primarily used to represent the OS/390 host - some limited deployment of application-specific VIPA, but no specific support in place for that use.
- Static VIPA have no dependencies on Sysplex functions.
- Connection resilience to failure of network interface.
- If an application was to be moved from one OS/390 image to another, the DNS could be updated to point to V2 instead of V1.
- In TCP/IP V3R2, you could manually (through OBEYFILE commands) move a static VIPA address from one stack to another - the concepts were identical to dynamic VIPA, but the movement was completely manual.
- Static VIPA address usage has no requirements on Sysplex functions - static VIPAs can be defined and moved between LPARs that are not in a Sysplex.

CS for OS/390 V2R8 Dynamic VIPA Support



- A VIPA address can either represent an OS/390 host or an individual application where the name server is updated to include resource records that identify individual applications, such as, myCICS.xyz.com at IP address V1.
- VIPA still addresses connection resilience but now also addresses application recovery. If an OS/390 image is taken down, Dynamic VIPA backup policies can be used to define where the associated DVIPAs move within the Sysplex. Dynamic VIPA support also allows for manual movement of applications and associated DVIPA addresses.
- Dynamic VIPA functions require that the LPARs between which dynamic VIPAs may move are in a Sysplex (a base Sysplex is enough for most functions to work)

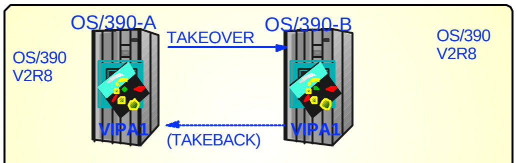
The concept of virtual IP addresses is spreading to other platforms, such as OS/400, AIX, Linux, zVM, and other vendors.

Dynamic VIPA usage - overview

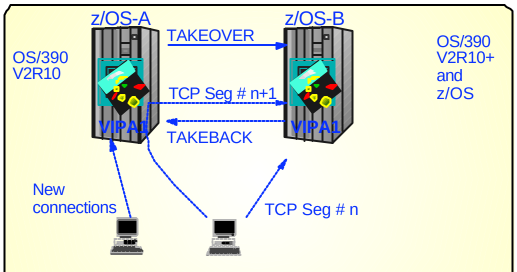
A dynamic VIPA address has all the attributes of a static VIPA address. In addition to those it has the ability to move between TCP/IP stacks in a Sysplex based on certain events - without operator intervention in terms of configuration changes.

When do you want the dynamic VIPA to move?	What's the type of DVIPA to use?	How do you define it?	Application requirements	Typical use
Move to a backup stack, when the currently owning stack goes down or is taken down.	**A stack- managed DVIPA**	VIPADEFINE on primary owner - VIPABACKUP on potential backup stacks.	Applications bind to INADDR_ANY.	Multiple instances of server runs on multiple stacks and can back each other up.
Move along with a specific server application that binds its listening socket to the dynamic VIPA address.	**An application-specific DVIPA**	VIPARANGE	Applications must bind to the specific dynamic VIPA address (alternatively use BIND specific on port reservation)	Single instance application that is moved between stacks - planned or unplanned.
Move when instructed to do so by executing a utility (moddvipa) or by an authorized application (using an ioctl call)	**A command-activated DVIPA**	VIPARANGE	No special requirements, but typically application binds to INADDR_ANY.	Single instance applications that cannot be controlled via bind specific functions.

Don't loose a connection just because we move a DVIPA from one LPAR to another!



1. B takes over VIPA1 when A goes down.
2. When A comes back up, a request for getting VIPA1 back is queued to B.
3. B doesn't give VIPA1 back until there are no more connections with VIPA1 on B - which may take some time.



1. When A comes back up again, it immediately takes VIPA1 back.
2. New connections to VIPA1 are accepted by A.
3. IP packets to old connections are routed via A to B for as long as such connections remain.

Gets workload back to where it rightfully belongs

- New connections using VIPA are handled by primary owner

Non-disruptive to connections established to backup

- Connection data forwarded to backup by primary owner
  - Uses internal Sysplex Distributor function
  - No additional configuration externals required
- Can be used in conjunction with Sysplex Distributor so that workload can be distributed to multiple backup servers during primary owner outage
  - Reduce impact of primary owner outage
  - Connection data forwarded to appropriate backup by primary owner
- Allows movement of application server without impacting existing workload
- Data for existing connections continue to be forwarded to old location

If a server instance goes or is taken down, all active connections with that server instance are lost - but if the DVIPA moves automatically to another LPAR, new connections can immediately be established

- Server or LPAR failure are disruptive to connections that were active with that server or LPAR

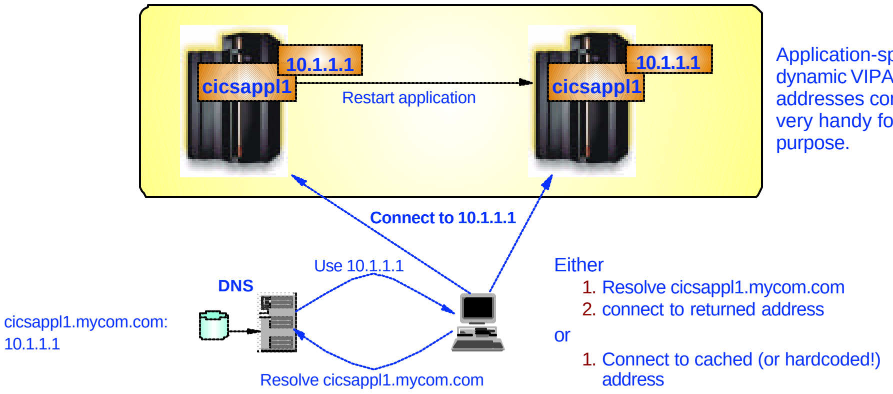
When is the DVIPA actually moved?

VIPADEFINE / VIPABACKUP	Initial activation on primary owner	Primary goes down, backup activates	Primary comes back up and tries to activate
Moveable IMMEDIATE	Successful	Successful	Successful - keep active on backup until connections terminate, then deactivate on backup
Moveable WHENIDLE	Successful	Successful	Delay activation until no more connections on the backup

VIPARANGE	Initial activation (not active elsewhere in sysplex) - bind activation or MODDVIPA activation	Application on other stack binds to address that is already active elsewhere in sysplex	IOCTL or MODDVIPA activation of address that is already active elsewhere in the sysplex
Moveable DISRUPTIVE	Successful	New bind fails.	1. Deactivate on current owning stack (break connections) 2. Successful activation on new stack
Moveable NONDISRUPTIVE	Successful	Successful - keep active on old stack until connections terminate	Successful - keep active on old stack until connections terminate

Basic principles for recovery of single-instance IP application in a Sysplex

- Single-instance applications are applications that only run in one instance in the Sysplex. Either because the application needs exclusive access to certain resources, or because there is no need to start it in more than one instance.
- Availability from an IP perspective then becomes an issue of being able to restart the application on the same LPAR or on another LPAR with as little impact to end-users as possible.
  - Speed of movement - ARM or automated operations procedures
  - Retain identity from a network perspective (its IP address) - application-specific DVIPAs



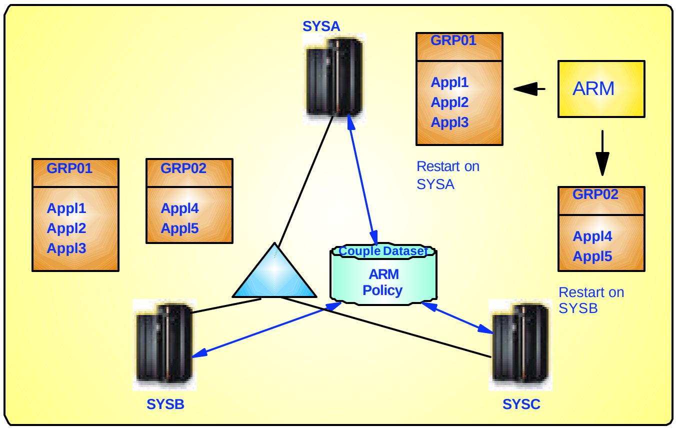
Application-specific dynamic VIPA addresses come in very handy for this purpose.

Either
1. Resolve cicsappl1.mycom.com
2. connect to returned address

or
1. Connect to cached (or hardcoded!) address

Automatic restart of application or subsystems using Automatic Restart Manager (ARM)

e-business



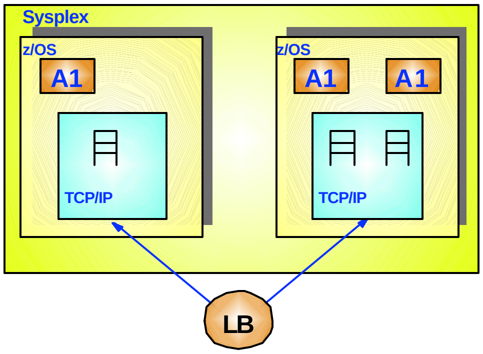
Exploit MVS Automatic Restart Manager (ARM)

- **Registered applications automatically restarted on failure**
  - ARM policy provides an ordered list for recovery
  - VTAM registers with ARM for restart
  - TCP/IP stack registers with ARM for in-place restart **(new in R8)**
- **ARM facility is open interface which can be exploited by any application**
  - Exploited by CICS, IMS, DB2

Or use your automated operations package to restart the application in another LPAR.

Multiple-instance capable server - connection load balancing and server availability

e-business



Connection load balancing technologies:

Between z/OS images:

a. DNS - DNS/WLM
b. NAT - CSS, many others
c. MAC forwarders - IND, MNLB, Sysplex Distributor
d. Contents-based - CSS, BigIP, etc.

Inside z/OS:

a. Port sharing

Application Characteristics:

- Multiple instances of the server are able to provide the exact same services to clients (will typically require data sharing)
- No state preserved at server between two connections (application protocol has to include support for such behavior or store state data in shared storage)

Benefits of Intelligent Load Balancing:

- ***Performance*** - improving response time
- ***Availability*** - If one instance goes down, connections with it break, but new connections can be established with remaining instance(s)
- ***Scalability*** - more server instances can be added on demand (horizontal growth)

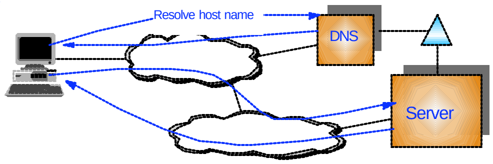
Examples:

- Web server
- TN3270 server
- Some CICS applications
- FTP server
- DB2
- MQ
- WAS

TCP connection load balancing technologies Notes

e-business

Generally not recommended - except for availability purposes!

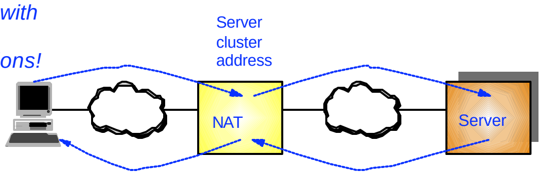


Good for long-duration connections and availability purposes, not good for short-duration connections. Mostly used for availability and geography selection.

DNS Based

- Server is selected at name resolution time
- Extra round-trip to resolve host name (does not fit well with short duration connections, such as HTTP requests)
- Relies on client hosts adhering to DNS TTL information of zero (not all do!)
- Inbound and outbound packets may flow through any available path between client and server
- Various implementations ranging from spraying (round-robin DNS) to more intelligent solutions where DNS knows if servers are running and what the workload of the target systems is (DNS/WLM on z/OS).

Restrictions on outbound routing and use with certain applications!



NAT is generally not a desirable technique due to various inherent problems with NAT, but it is widely used and cannot be ignored.

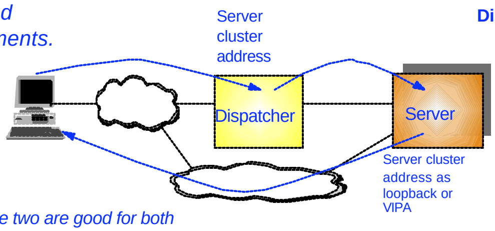
NAT (IP level forwarding)

- Server is selected at connect time
- Server can be on any host (one or more hops away)
- All inbound and outbound packets must flow through the NAT box
- Will not work well with IPSec
- Outbound routes from servers must point to NAT boxes: all outbound traffic goes through NAT boxes, not just output from balanced workload.
- Will not work for all application protocols that include IP addresses in data stream
  - private protocols will not work
  - encrypted well-known protocols will not work

TCP connection load balancing technologies Notes

e-business

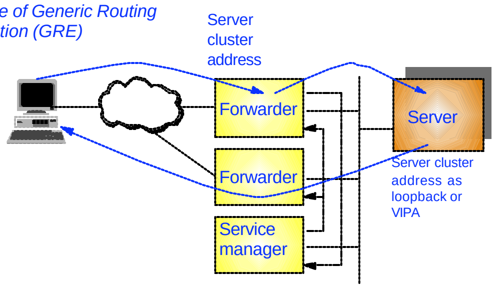
Works well in controlled environments.



Both these two are good for both short-duration and long-duration connections. The one-hop-away limitation can be addressed through use of Generic Routing Encapsulation (GRE)

Dispatcher (MAC level forwarding)

- Server is selected at connect time
- Server must be on a directly connected network (no routing)
- All inbound packets must flow through dispatcher
- Outbound packets may flow over any available path between the server and the client
- Doesn't work well with OSA adapters in shared mode
- IBM's Network Dispatcher works this way

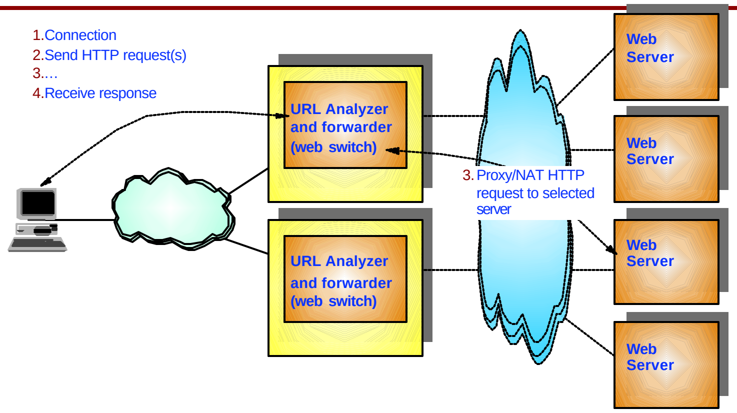


Multi-Node Dispatcher (MAC level forwarding)

- Basic concept same as Dispatcher, but implementation is distributed onto multiple nodes
- Server is selected at connect time
- When forwarders receive connection requests, they ask the service manager to make a server selection
- Routers that sit one hop away from servers, receive information from service manager and do forwarding of IP packets to selected servers
- Inbound and outbound may use any of the one-hop away routers in front of the servers
- Same problem with OSA adapters as Dispatcher solution
- Service manager may get workload information from server hosts
- Cisco's MNLB works this way

Contents-based load balancing for HTTP requests Notes

e-business


1.Connection
2.Send HTTP request(s)
3.…
4.Receive response
URL Analyzer and forwarder (web switch)
URL Analyzer and forwarder (web switch)
3.Proxy/NAT HTTP request to selected server
Web Server
Web Server
Web Server
Web Server

- Application-protocol specific solutions (typically limited to web traffic - HTTP and sometimes also HTTPS when combined wtih SSL/TLS offload capbilities on the switch)
- Connection load-balancing can be deployed in front of the web switches
- Each back-end server is specialized to service selected URIs
- IBM's WebSphere Edge Server (WSES) with its Contents Based Routing (CBR) module works this way using an HTTP proxy approach.
- Various switching vendors also support HTTP contents-based routing using a modified NAT approach.

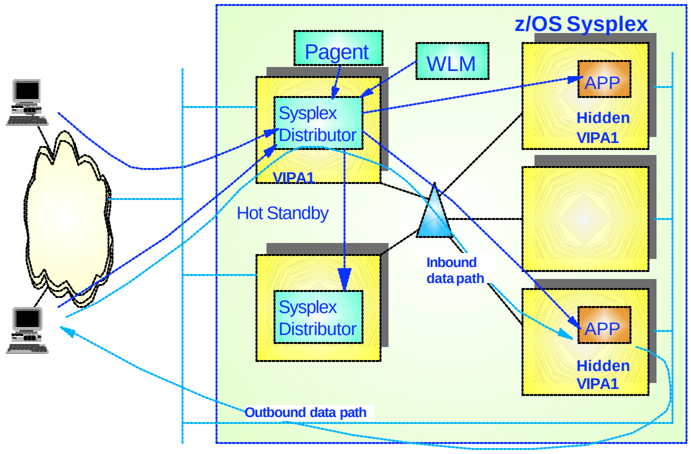
New, promising technology. Issues do exist with end-to-end security - IPsec and SSL/TLS connections must terminate on the load-balancing devices.

Some implementations now do support HTTPS, but not all.

Sysplex Distributor: z/OS-integrated intra-Sysplex workload balancing

e-business

- Independent of network attachment technology. Will work with both direct (including OSA Express) and channel-attached router network connections.
- All z/OS images communicate via XCF. Each TCP/IP stack has full knowledge of IP addresses and server availability in all stacks.
- A network-connected stack owns a given VIPA address and acts as the distributor of new connection requests to that VIPA address.

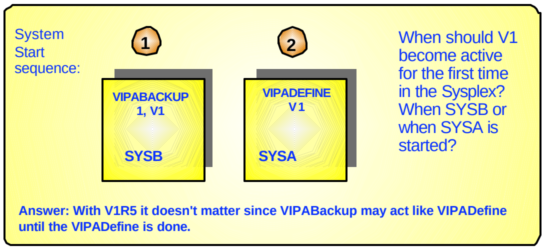


- Distribution of new connection requests is based on real-time consultation with WLM (if available, else round-robin), z/OS QoS policy agent and target stack for application availability.
- Connection information is stored in Sysplex distributor stack for routing of IP packets belonging to an existing connection to the appropriate target stack. Routing is based on the connection to which IP packets belong (connection-based routing).
- The hot standby stack is free to do other work, but will take over ownership of the VIPA address and the distributor function in case the primary stack fails. This takeover is non-disruptive to existing connections.

Overviev over TCP/IP Sysplex enhancements in z/OS V1R5

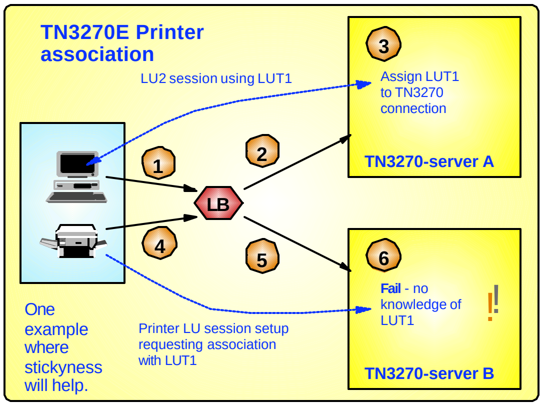
Sysplex Enhancements

- Increase ports on VIPADISTRIBUTE from 4 to 64
- Dynamic port definition for VIPADISTRIBUTE dynamic VIPA when server binds to dynamic VIPA
- Increase limit of DVIPAs per stack from 256 to 1024
- Support DVIPA activation based on VIPABACKUP before VIPADEFINE ever processed
- Sysplex Distributor affinity
  - Configurable timer-based stickyness per source IP address, server DVIPA and port
- New round-robin distribution method in Sysplex Distributor
  - Alternative to WLM-based distribution
  - Useful where availability is more important than capacity



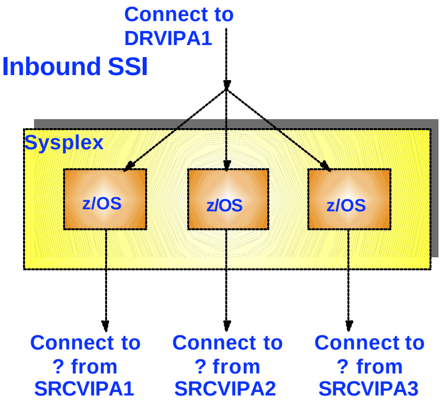
When should V1 become active for the first time in the Sysplex? When SYSB or when SYSA is started?

Answer: With V1R5 it doesn't matter since VIPABackup may act like VIPADefine until the VIPADefine is done.


TN3270E Printer association
LU2 session using LUT1
3
Assign LUT1 to TN3270 connection
TN3270-server A
1
2
LB
4
5
6
Fail - no knowledge of LUT1
!
TN3270-server B
Printer LU session setup requesting association with LUT1

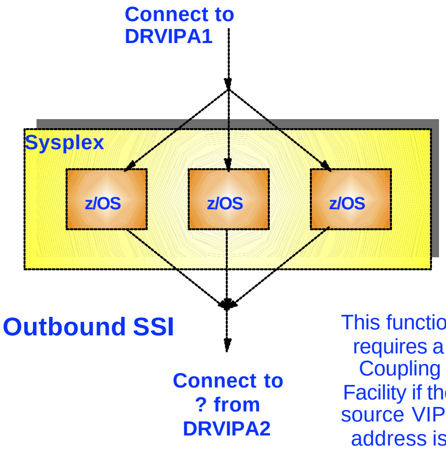
One example where stickyness will help.

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


Connect to DRVIPA1
Inbound SSI
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 conections 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.

- z/OS V1R4 introduced new capabilities 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.



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

When is a source VIPA address being used?

e-business

A source VIPA address is used when the following conditions are met:

- The SOURCEVIPA option has been enabled in the IPCONFIG statement

  AND

- An outbound connection or UDP association is being established from z/OS

  AND

- The application has not bound the local socket to a specific interface IP address before establishing an outbound connection or UDP association

  AND

- The application has not disabled the use of SOURCEVIPA through a setsockopt call

SOURCEVIPA is not being used when outbound data is sent on a connection that was established inbound to z/OS (such as data sent as a response on a TN3270(E) connection that was established from a remote TN3270(E) client to the TN3270(E) server on z/OS).

An FTP outbound data connection is always established from a socket that was bound to the same server IP address as to where the control connection was directed.

Which source VIPA address is being used?

e-business

- Two basic rules:
  - If TCPSTACKSOURCEVIPA ***is not enabled*** on the IPCONFIG statement, then the SOURCEVIPA address to use is selected based on the order of the HOME list

```
10.0.0.1    VIPALINK1
192.168.1.1 REALLINKA  ; Will use 10.0.0.1 as SOURCEVIPA
10.0.0.2    VIPALINK2
192.168.2.1 REALLINKB  ; Will use 10.0.0.2 as SOURCEVIPA
```

If the connection setup request is sent over REALLINKA, then 10.0.0.1 will be used as source VIPA. If the connection setup request is sent over REALLINKB, then 10.0.0.2 will be used as source VIPA.

  - If TCPSTACKSOURCEVIPA ***is enabled*** on the IPCONFIG statement, then the IP address specified as TCPSTACKSOURCEVIPA will be used as source VIPA address for all outbound TCP connections, while UDP and RAW associations will continue to use a source VIPA address based on the order of the HOME list

- If the IP address that is specified on the TCPSTACKSOURCEVIPA option is used on multiple LPARs in the Sysplex, then the address must be defined as a VIPADISTRIBUTE IP address and the SYSPLEXPORT option must be specified on the VIPADISTRIBUTE statement.

```
VIPADYNAMIC
  VIPADEFINE 255.255.255.192 201.2.10.11 201.2.10.12
  VIPADISTRIBUTE DEFINE SYSPLEXPORTS 201.2.10.11 PORT 9999
             DESTIP 201.3.10.10 201.3.10.11
  VIPABACKUP 100 201.2.10.13
ENDVIPADYNAMIC
IPCONFIG .... TCPSTACKSOURCEVIPA 201.2.10.11 ....
```

A summary of the different types of VIPA addresses

e-business

Static VIPA

Belongs to one TCP/IP stack. Manual configuration changes needed to move it.

- No dependencies on Sysplex functions

Dynamic VIPA

Stack-managed

- Belongs to one TCP/IP stack, but backup policies governs which TCP/IP stack in the sysplex takes it over if the primary TCP/IP stack leaves the sysplex

Application-specific

- Belongs to an application. Comes active on the TCP/IP stack in the sysplex where the application is started. Moves with the application.

Command-activated

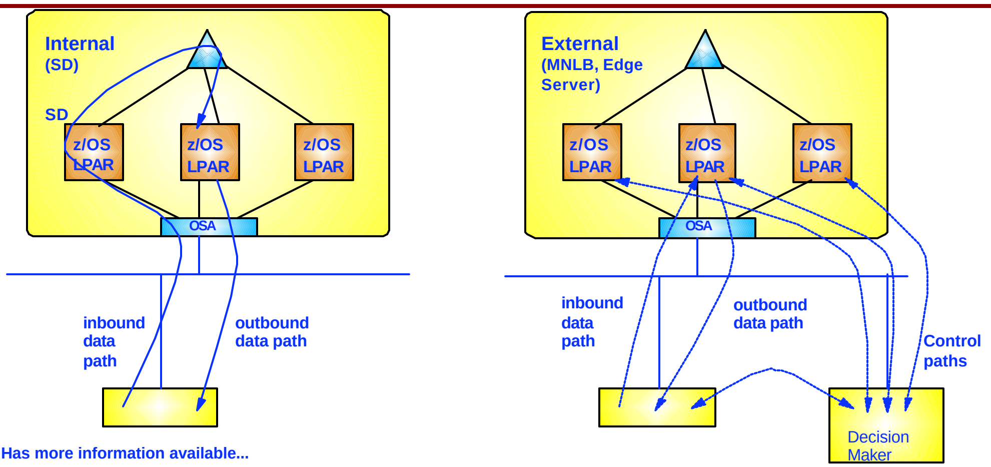
- Belongs to whatever TCP/IP stack in the sysplex on which a command to activate the address has been issued. Moves between TCP/IP stacks based on operator commands.

Distributed

- Used with Sysplex Distributor as a cluster IP address that represents a cluster of equal server instances in the sysplex. From a routing perspective it belongs to one TCP/IP stack. From an application perspective it is distributed among the TCP/IP stacks in the sysplex where an instance of the server application is executing.

Sysplex internal vs. external workload balancing - which technology is best for me?

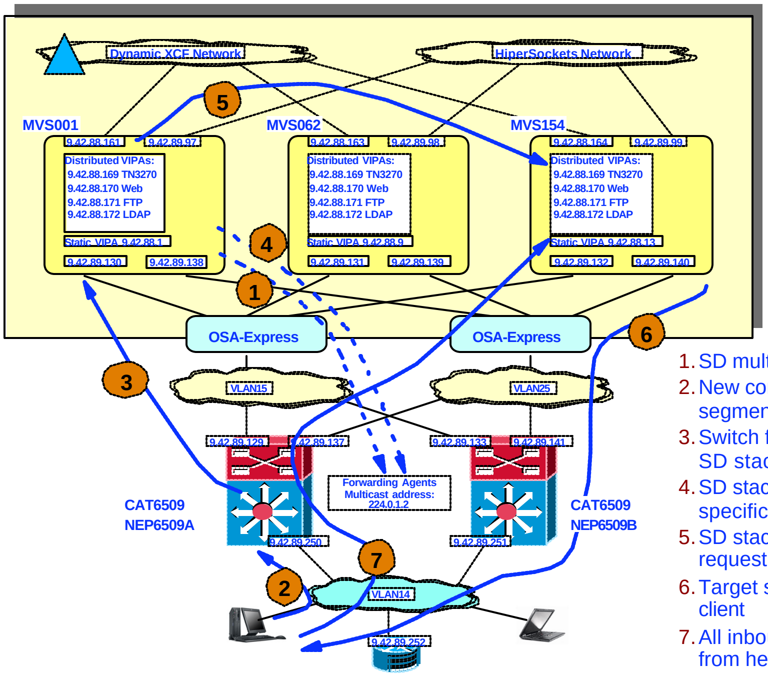
e-business



- Has more information available...
  - more timely capacity information
  - QoS from Service Policy Agent
  - application-independent server availability
- No problems with shared OSA adapters; no intermediate routers
- Uses expensive Sysplex resources for inbound traffic
  - Inbound traffic funneled through single point
  - Routing stack uses zSeries MIPs for inbound routing
  - Inbound traffic routed over XCF

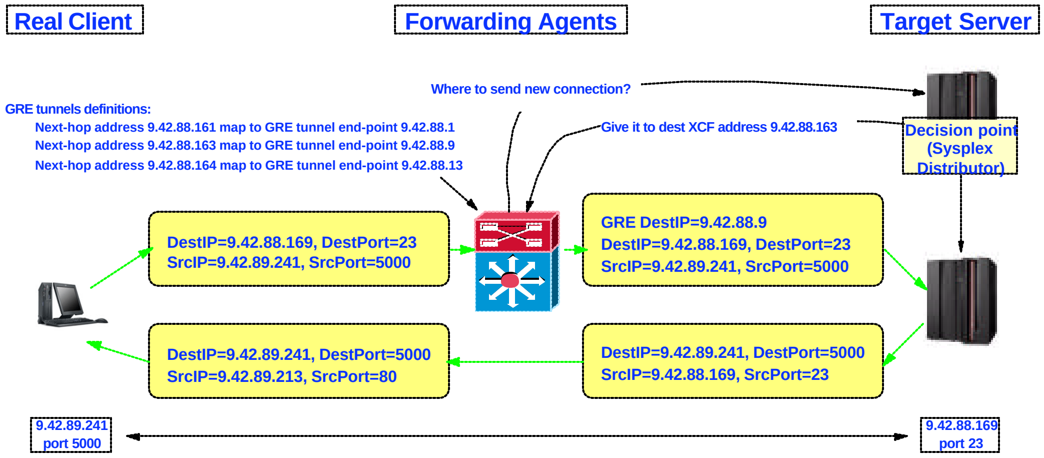
- Specialized routing hardware may be more cost-effective
- May be configured for no single point of traffic flow
- No general Sysplex-capacify feedback technology (some specialized attempts have been made)
- Requires application-specific health probes ("application ping")
- Problems with shared OSA adapters or other intermediate routers

Sysplex Distributor / Cisco MNLB forwarding agents



1. SD multicasts wildcard affinity
2. New connection request (TCP SYN segment) arrives in switch
3. Switch forwards connection request to SD stack
4. SD stack makes a decision and unicasts specific affinity back to switch
5. SD stack forwards the connection request over XCF to target stack
6. Target stack replies directly back to client
7. All inbound data for specific connection from here on flows directly from switch to target stack

Sysplex Distributor / MNLB Dispatch Mode forwarding



Real Client

Forwarding Agents

Target Server

Where to send new connection?

GRE tunnels definitions:

Next-hop address 9.42.88.161 map to GRE tunnel end-point 9.42.88.1
Next-hop address 9.42.88.163 map to GRE tunnel end-point 9.42.88.9
Next-hop address 9.42.88.164 map to GRE tunnel end-point 9.42.88.13

Give it to dest XCF address 9.42.88.163

Decision point (Sysplex Distributor)

DestIP=9.42.88.169, DestPort=23
SrcIP=9.42.89.241, SrcPort=5000

GRE DestIP=9.42.88.9
DestIP=9.42.88.169, DestPort=23
SrcIP=9.42.89.241, SrcPort=5000

DestIP=9.42.89.241, DestPort=5000
SrcIP=9.42.89.213, SrcPort=80

DestIP=9.42.89.241, DestPort=5000
SrcIP=9.42.88.169, SrcPort=23

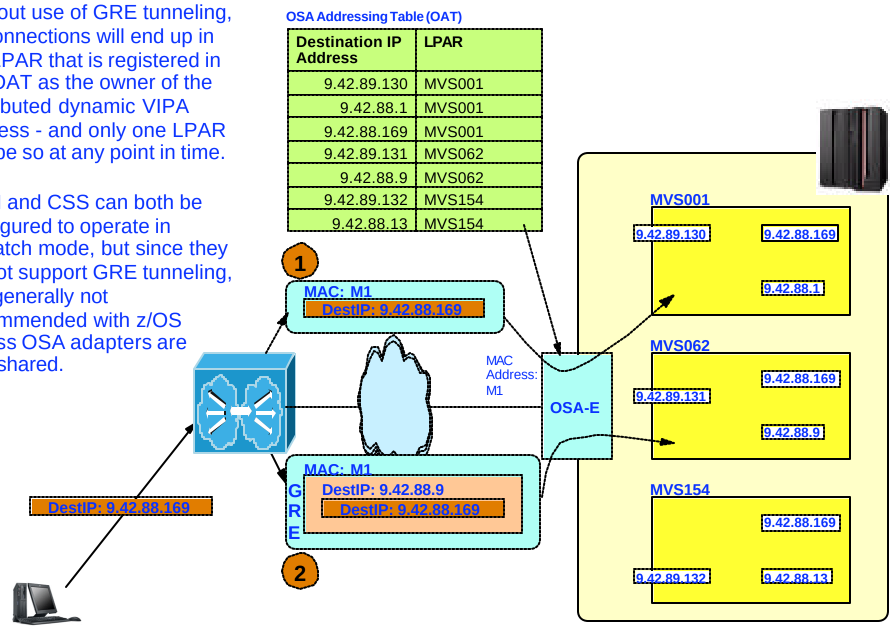
9.42.89.241
port 5000

9.42.88.169
port 23

- Load balancing decision point is inside the z/OS Sysplex and can take LPAR capacity into consideration
- Server IP address and client IP address are never NATed - there is no requirement for outbound packets to be routed via any specific path
- Must be combined with use of Generic Routing Encapsulation to overcome one-hop away and shared OSA limitations

Sysplex Distributor / MNLB
Dispatch Mode use of GRE tunneling w. shared OSA ports

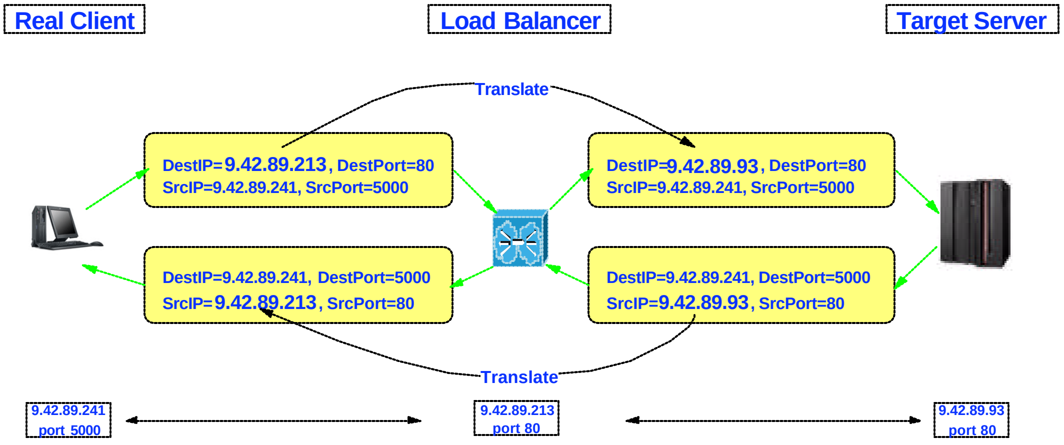
- Without use of GRE tunneling, all connections will end up in the LPAR that is registered in the OAT as the owner of the distributed dynamic VIPA address - and only one LPAR can be so at any point in time.
- CSM and CSS can both be configured to operate in dispatch mode, but since they do not support GRE tunneling, it is generally not recommended with z/OS unless OSA adapters are non-shared.



OSA Addressing Table (OAT)

Destination IP Address	LPAR
9.42.89.130	MVS001
9.42.88.1	MVS001
9.42.88.169	MVS001
9.42.89.131	MVS062
9.42.88.9	MVS062
9.42.89.132	MVS154
9.42.88.13	MVS154

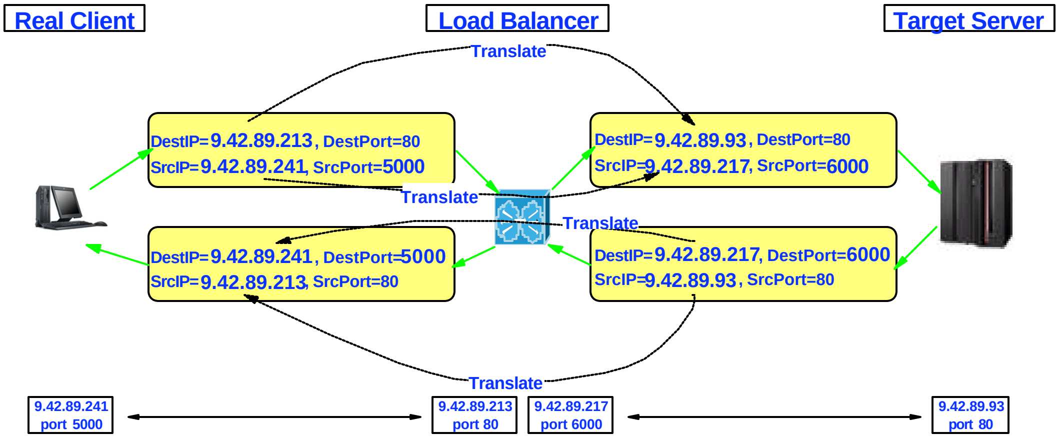
Cisco CSS/CSM - external decision point
Directed mode forwarding - Server NAT combined with Policy Based Routing (PBR)



Real Client
Load Balancer
Target Server
Translate
DestIP=9.42.89.213, DestPort=80
SrcIP=9.42.89.241, SrcPort=5000
DestIP=9.42.89.93, DestPort=80
SrcIP=9.42.89.241, SrcPort=5000
DestIP=9.42.89.241, DestPort=5000
SrcIP=9.42.89.213, SrcPort=80
DestIP=9.42.89.241, DestPort=5000
SrcIP=9.42.89.93, SrcPort=80
Translate
9.42.89.241
port 5000
9.42.89.213
port 80
9.42.89.93
port 80

- Only the server IP address is NATed (destination IP address on inbound and source IP address on outbound)
- Policy-based routing in routing infrastructure re-directs outbound IP packets from target servers to the load-balancer so it can NAT the source IP address in outbound packets
- Outbound packets that do not need NATing of the source IP address is routed using normal IP routing table processing
- Real client IP address information is available to target servers

Cisco CSS/CSM - external decision point Directed mode forwarding - Server NAT and Client NAT



Real Client

Load Balancer

Target Server

Translate

DestIP=9.42.89.213, DestPort=80
SrcIP=9.42.89.241, SrcPort=5000

Translate

DestIP=9.42.89.93, DestPort=80
SrcIP=9.42.89.217, SrcPort=6000

Translate

DestIP=9.42.89.241, DestPort=5000
SrcIP=9.42.89.213, SrcPort=80

DestIP=9.42.89.217, DestPort=6000
SrcIP=9.42.89.93, SrcPort=80

Translate

9.42.89.241 port 5000

9.42.89.213 port 80

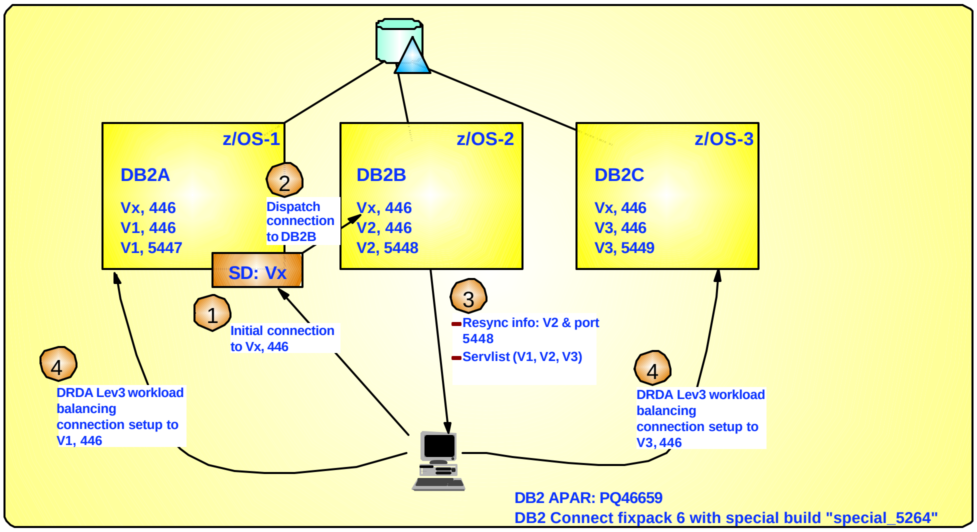
9.42.89.217 port 6000

9.42.89.93 port 80

- Both server IP address and client IP address are NATed by the load balancer - no need for use of Policy Based Routing since outbound packets from target servers are destined for a load balancer IP address
- Client IP addresses seen by target servers is an IP address on the load balancer and not the real client IP address
- Should be used with care if any of the following functions on server nodes are in use:
  - Networking policy conditions based on client IP address information
  - NETACCESS rules for access control and/or MLS label assignment
  - Server configuration options based on source IP address information, such as TN3270 server LU name assignment

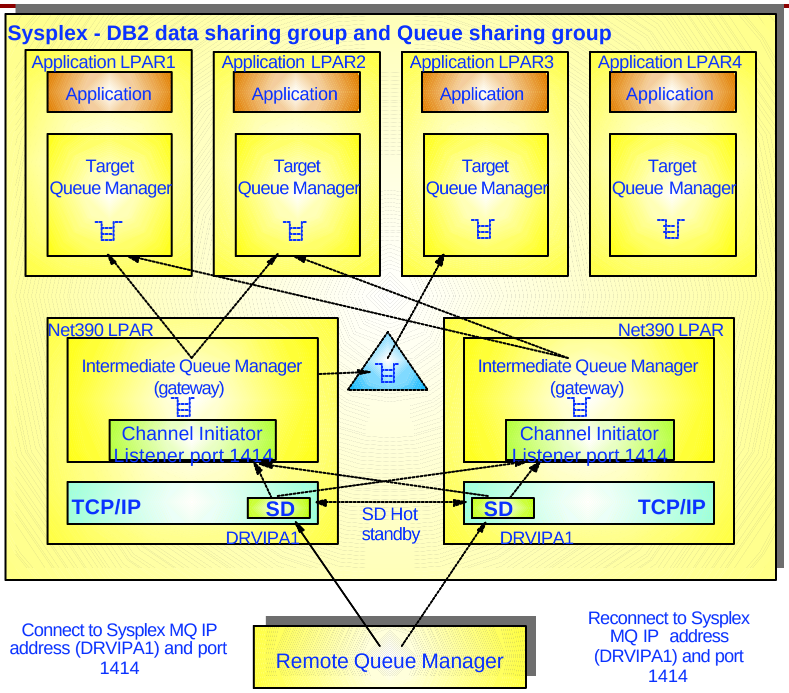
Appendix Subsystem Examples of DVIPA Usage

z/OS DB2 DRDA
Dynamic VIPAs and Sysplex Distributor

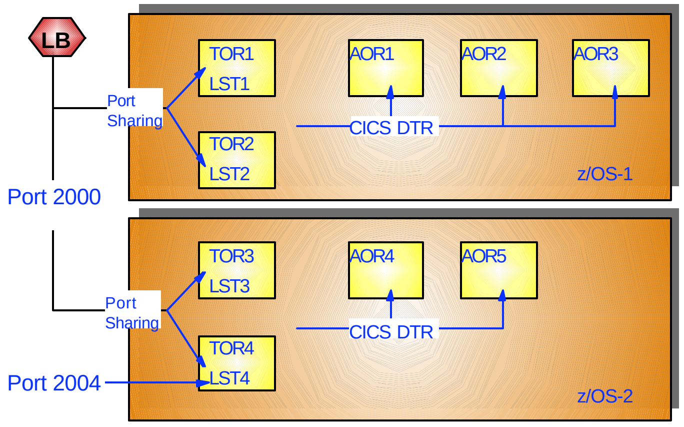

z/OS-1
DB2A
Vx, 446
V1, 446
V1, 5447
z/OS-2
DB2B
Vx, 446
V2, 446
V2, 5448
z/OS-3
DB2C
Vx, 446
V3, 446
V3, 5449
SD: Vx
2
Dispatch connection to DB2B
1
Initial connection to Vx, 446
3
Resync info: V2 & port 5448
Servlist (V1, V2, V3)
4
DRDA Lev3 workload balancing connection setup to V1, 446
4
DRDA Lev3 workload balancing connection setup to V3, 446
DB2 APAR: PQ46659
DB2 Connect fixpack 6 with special build "special_5264"

With the above listed APAR solution for DB2 V6 and V7, DB2 can use application-specific dynamic VIPA addresses and Sysplex Distributor for connection balancing. This allows for restart of a failed DB2 instance on another LPAR without manual movement of IP addresses.

Queue Manager gateway in a Sysplex using gateway nodes - one large Queue Sharing Group


Sysplex - DB2 data sharing group and Queue sharing group
Application LPAR1
Application
Target Queue Manager
Application LPAR2
Application
Target Queue Manager
Application LPAR3
Application
Target Queue Manager
Application LPAR4
Application
Target Queue Manager
Net390 LPAR
Intermediate Queue Manager (gateway)
Channel Initiator Listener port 1414
TCP/IP
SD
DRVIPA1
Net390 LPAR
Intermediate Queue Manager (gateway)
Channel Initiator Listener port 1414
SD
TCP/IP
DRVIPA1
SD Hot standby
Connect to Sysplex MQ IP address (DRVIPA1) and port 1414
Remote Queue Manager
Reconnect to Sysplex MQ IP address (DRVIPA1) and port 1414

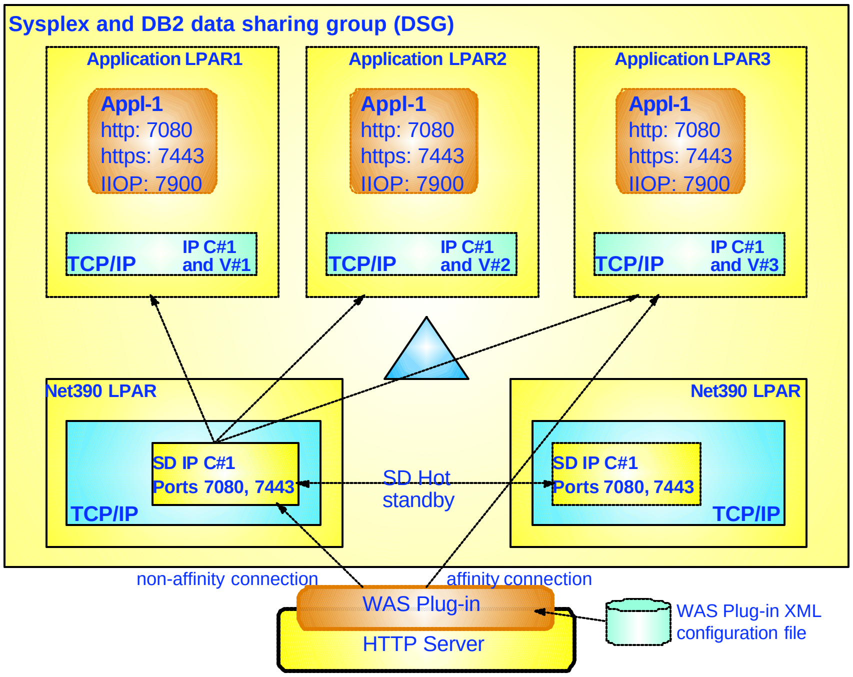
Workload balancing with CICS sockets



- Affinity to a certain TOR for multiple connections must be handled by the application protocol.
- An alternate port number could be defined for each TOR, and that port number could be returned to the client to use on succeeding connections that require affinity.

- CICS TS 1.3 and later supports Dynamic Transaction Routing for started transactions. Transaction must be defined as routable in region where start command is issued.
- DTRPGM=EYU9XLOP is the CICS Plex System Manager routine that will make a decision about which AOR to route to.
- The AOR must be on the same z/OS image as the TOR (givesocket/takesocket limited to same z/OS)

Sysplex Distributor as WAS application connection balancer in a Sysplex



The application servers must not be bind-specific - they must be able to respond to connections that arrive for both the cluster IP address (C#1) and server-specific IP address (V#1, V#2, or V#3).

For More Information....

e-business

URL	Content
http://www.ibm.com/servers/eserver/zseries	IBM eServer zSeries Mainframe Servers
http://www.ibm.com/servers/eserver/zseries/networking	Networking: IBM zSeries Servers
http://www.ibm.com/servers/eserver/zseries/networking/technology.html	IBM Enterprise Servers: Networking Technologies
http://www.ibm.com/software/network	Networking & communications software
http://www.ibm.com/software/network/commserver	Communications Server
http://www.ibm.com/software/network/commserver/library	Communications Server white papers, product documentation, etc.
http://www.redbooks.ibm.com	ITSO redbooks
http://www.ibm.com/software/network/commserver/support	Communications Server technical Support
http://www.ibm.com/support/techdocs/	Advanced technical support (flashes, presentations, white papers, etc.)
http://www.rfc-editor.org/rfcsearch.html	Request For Comments (RFC)