

IMS and SOA High Availability and Workload Balancing

Suzie Wendler
IBM IMS ATS

March 14, 2012
11001



Abstract

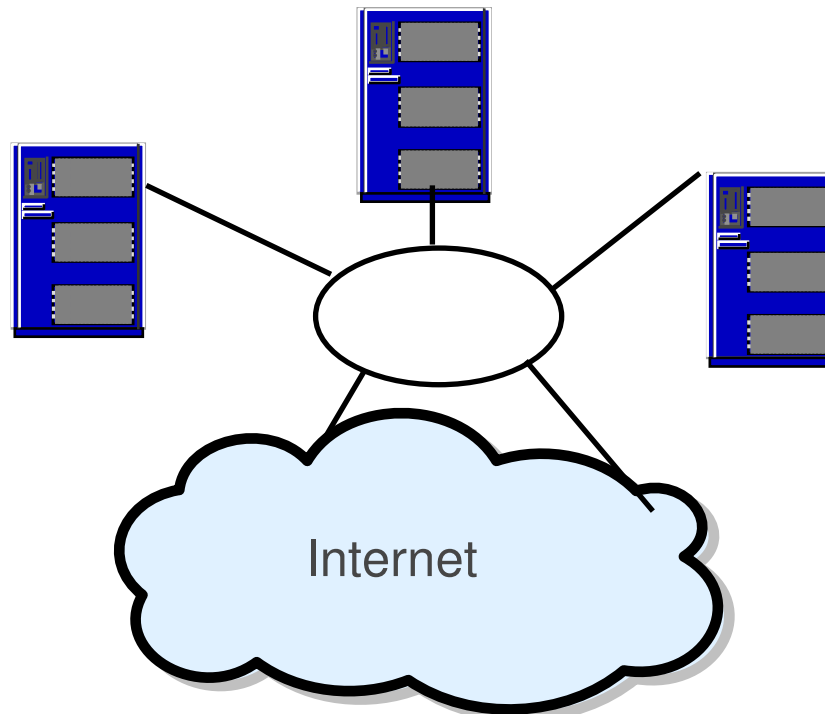
- With the ever-increasing options to integrate IMS into an SOA world, care must be taken to ensure that the proper decisions are made for scalability, backup, and failure scenarios. This session explains and compares different options including DVIPA, Sysplex Distributor, IMS Connect Datastore routing, the use of aliases with ODBM to manage workload requests for IMS resources, and even the considerations when using the new IMS callout capabilities.

Topics

- Workload Distribution, load balancing and failover concepts
 - DNS/WLM, VIPA, Sysplex Distributor
- Inbound:
 - IMS Connect Workload Distribution – Accessing IMS as a server
 - Transactions and Supermember support
 - Databases and ODBM aliases
- Outbound:
 - Calling out from IMS – Accessing external web services
 - Addressing single points of failure

Sysplex

- **Collection of connected system z processors**
 - Enables horizontal growth
 - **IMS: Shared Queues, Data Sharing, Multiple and/or Cloned Systems, MSC routing, ...**
 - Provides a single system image



Workload Distribution and Load Balancing Concepts

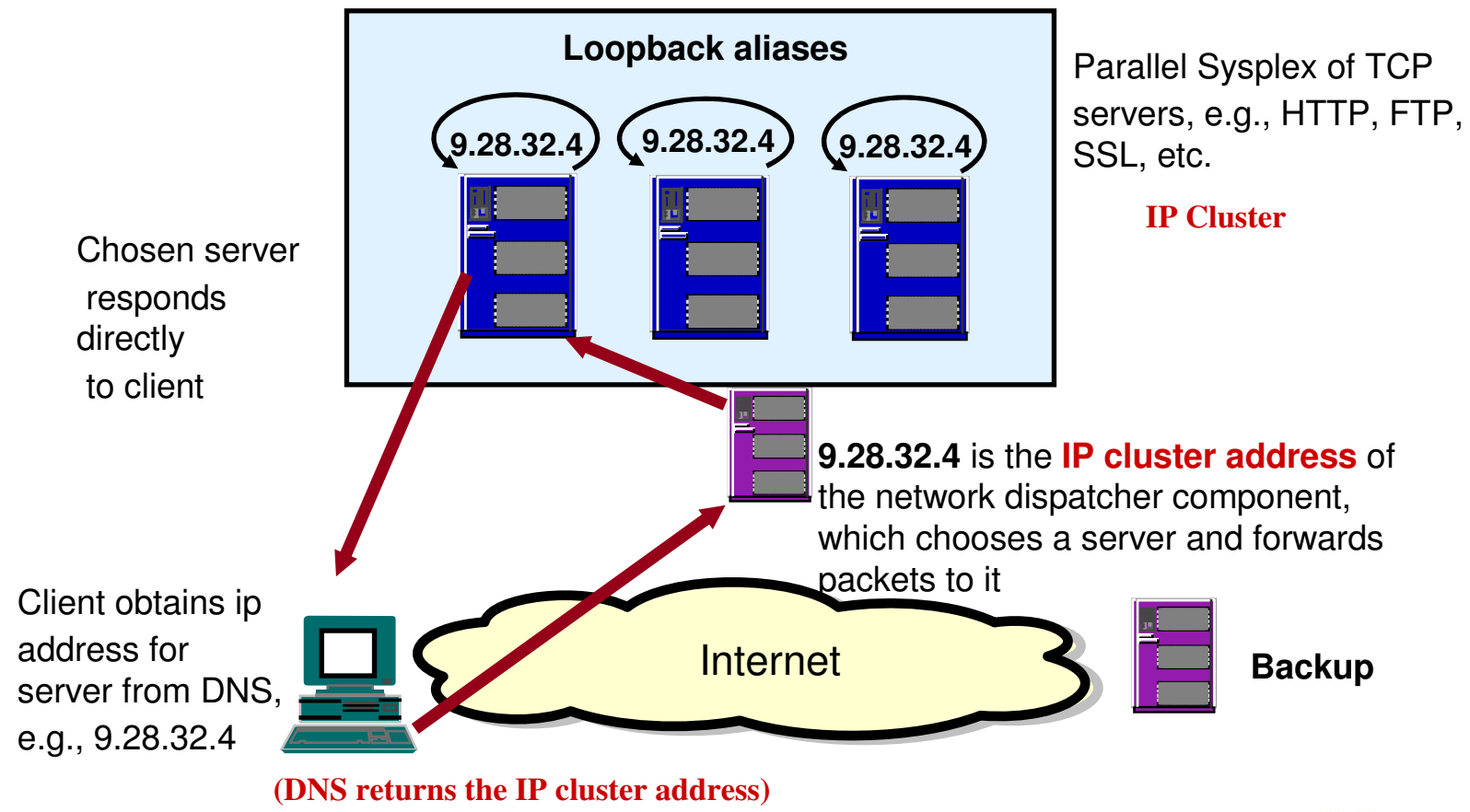


Load Balancing - e.g., WebSphere Edge Components

- Dispatcher support - IP Spraying
 - Intercepts connection requests
 - Attempts to balance traffic by choosing and then forwarding the request to a specific server, e.g., a specific instance of IMS Connect, in the sysplex.
- Establishes session with WLM if servers are z/OS
 - Balances workload based on workload goals
 - Never selects an unavailable server
- Provides scalability and failover
 - Detects new instances of a server as they are added
 - Detects server failure and prevents routing new connection requests to them
- Used for "short duration" applications like web traffic
 - Inbound data goes through the router
 - Outbound data goes directly to the client

Workload Distribution and Load Balancing ...

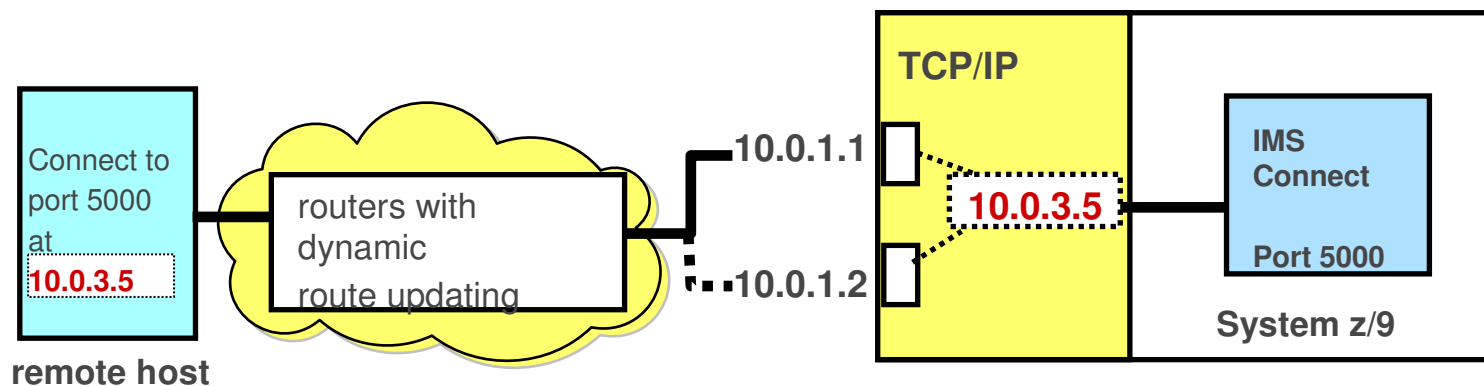
Load Balancing - e.g., WebSphere Edge Components



Failover

Static VIPA

- **Static Virtual IP Addressing (VIPA)**
 - First VIPA implementation
 - Eliminates an application's dependence on a particular network interface (IP address)
 - Non-disruptive rerouting of traffic in the event of failure
 - A defined VIPA does not relate to any physical network attachment
 - Multiple network interfaces on a single TCP/IP stack

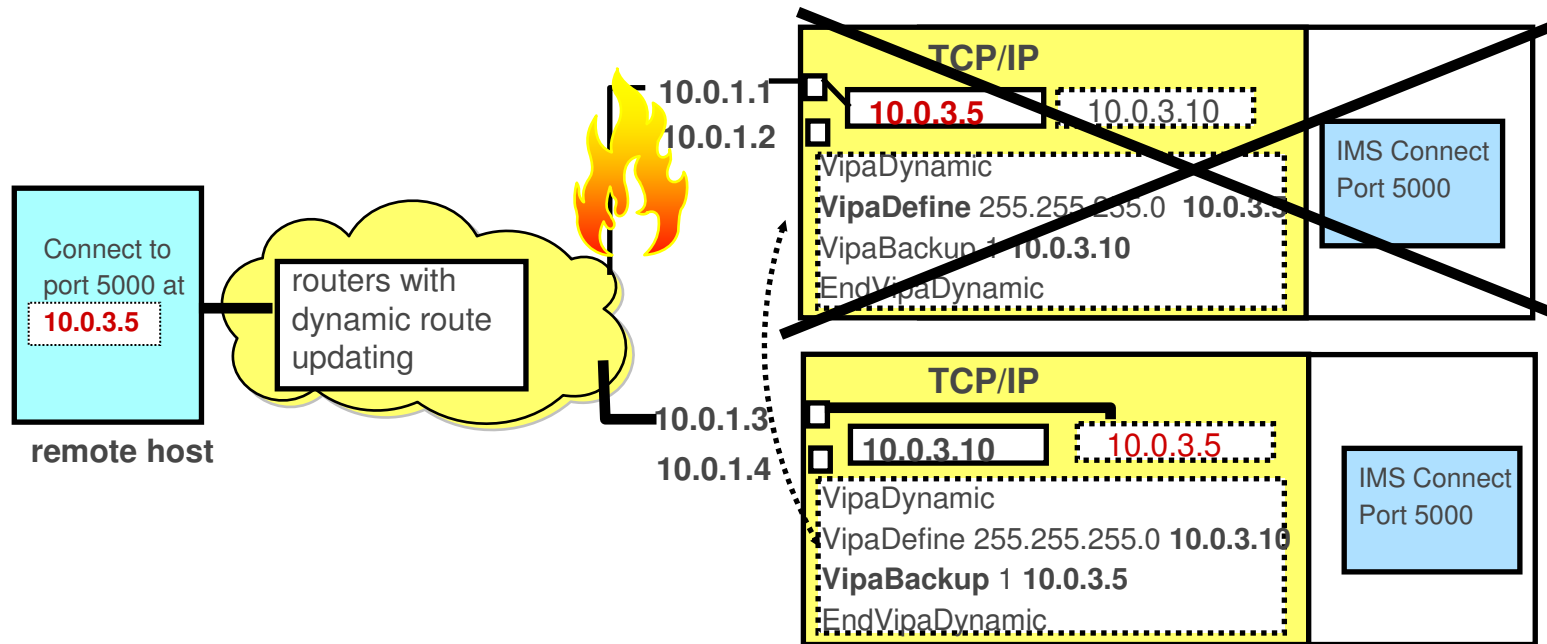


Note: The real network interfaces 10.0.1.1 and 10.0.1.2 appear to be intermediate hops
 10.0.3.5 is a virtual IP address and can never fail

Failover ...

Dynamic VIPA

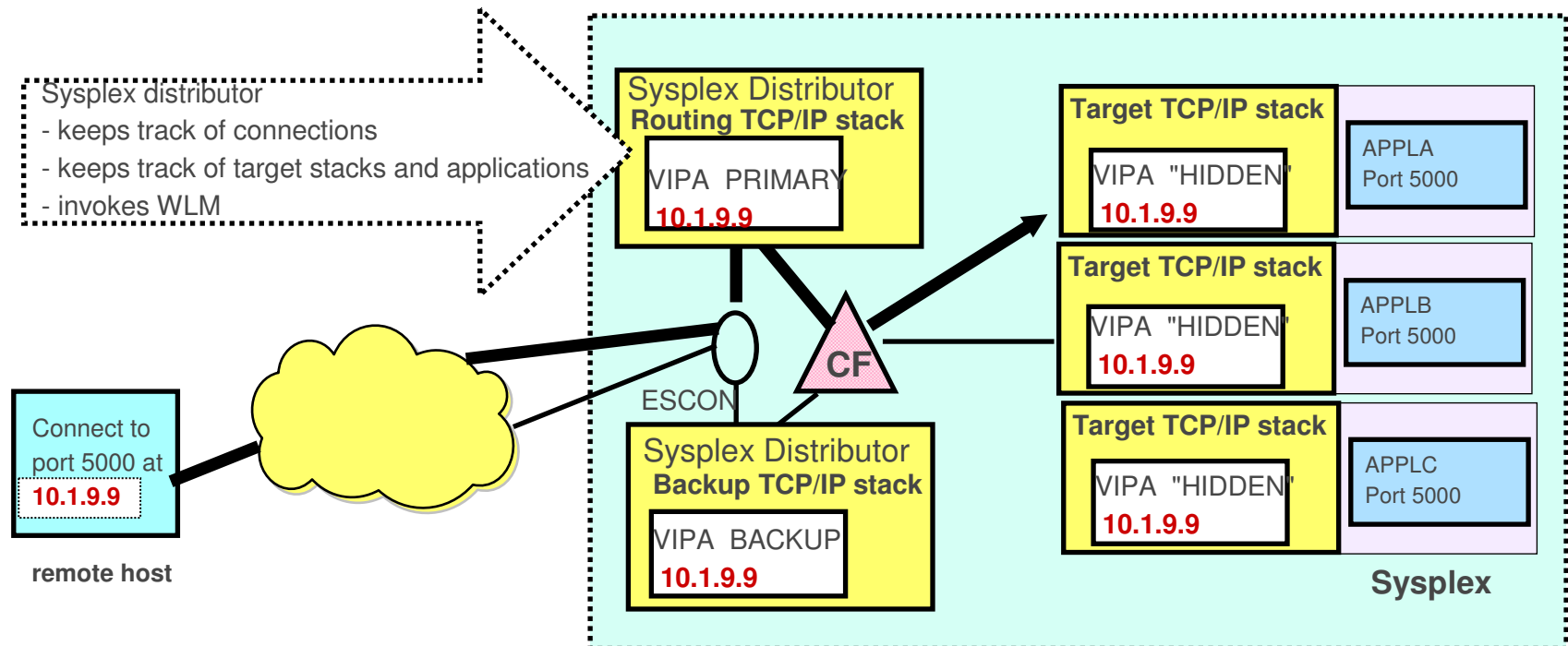
- **Automatic VIPA Takeover**
 - Support for other TCP/IP stacks to be backup VIPA address
 - Allows an active stack to assume the load of a failing stack
 - Stacks share information using z/OS XCF messaging



Note: Server application may need to be started on the same port on the backup host since the route to the primary host is unavailable, the backup host provides a backup for the virtual IP address of 10.0.3.5

Sysplex Distributor

- Sysplex function - Single IP address for a cluster of Hosts
 - Sysplex-wide VIPA
 - Workload balancing across multiple z/OS servers
 - Performs a Load Balancer type function on the System z environment
 - High availability
 - Enhanced Dynamic VIPA and Automatic Takeover
 - Allows movement of VIPAs without disrupting existing connections



Sysplex Distributor...

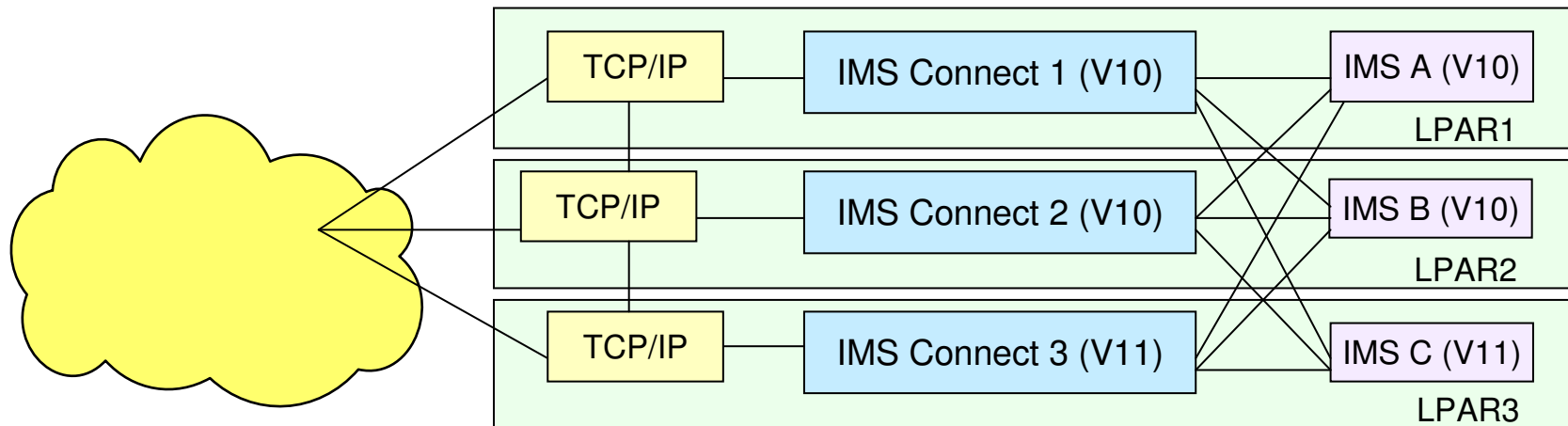
- z/OS Communications Server includes
 - **Server-Specific WLM for Sysplex Distributor**
 - Connections are distributed using a round-robin distribution
 - Or, based on a target server's (IMS Connect's ports) workload capacity
 - *Rather than the target system's (z/OS TCP/IP stack) capacity*
 - Balancing can also be done across a group of servers (multiple IMS Connect instances) sharing a port
 - **Sysplex autonomics health monitor for target stacks**
 - Provides a mechanism to monitor a target server's (IMS Connect's ports) connection setup responsiveness
 - *Approximately 1 minute intervals*
 - When connection responsiveness diminishes, new connection requests are diverted to alternate instances of the server

Portsharing

- A method to distribute workload for IP applications within a z/OS LPAR
 - Multiple instances of an application can listen on the same PORT number
 - Allows the workload to be distributed among the server applications listening on the same port
 - Simplifies the request for the remote client that does not know there might be multiple server application instances that can accept the message
 - Works with the Sysplex Distributor but does not require it
 - Can be implemented using round-robin distribution or with the workload manager

IMS Connect

- The use of mechanisms such as IP spraying, workload balancing and sysplex distribution
 - Allow a connection request to be routed to any of the available IMS Connect instances
 - Different versions of IMS Connect and IMS can coexist
 - *Mixed versions of IMS Connect and IMS are limited to the functionality of the lower release level*

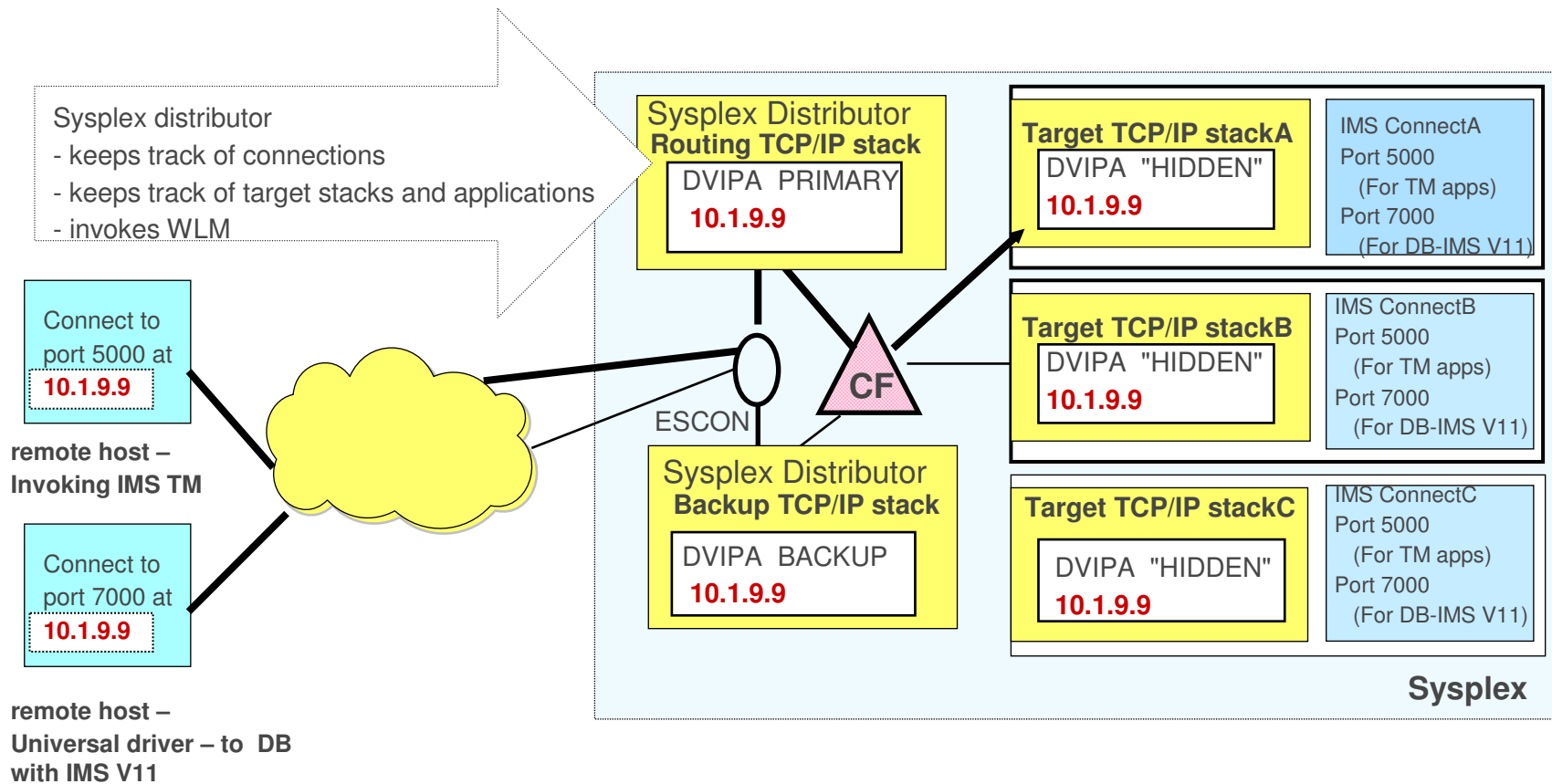


IMS Connect ...

- PORTS
 - Provide the mechanism to access different IMS resources
 - Specified in the HWSCFGxx configuration file
 - **PORTID** - defines the PORT numbers for access to IMS TM applications and commands
 - **PORT** - defines the PORT numbers, distinct from those defined in PORTID
 - Also for access to IMS TM applications and commands but allow an IMS Connect override of the TCP/IP Keepalive specification
 - **SSLPORT** – defines the Secure Socket Layer port
 - **DRDAPORT** (IMS V11)
 - Defines the PORT numbers, distinct from those defined in the PORT or PORTID statements, used specifically for access to IMS DB.

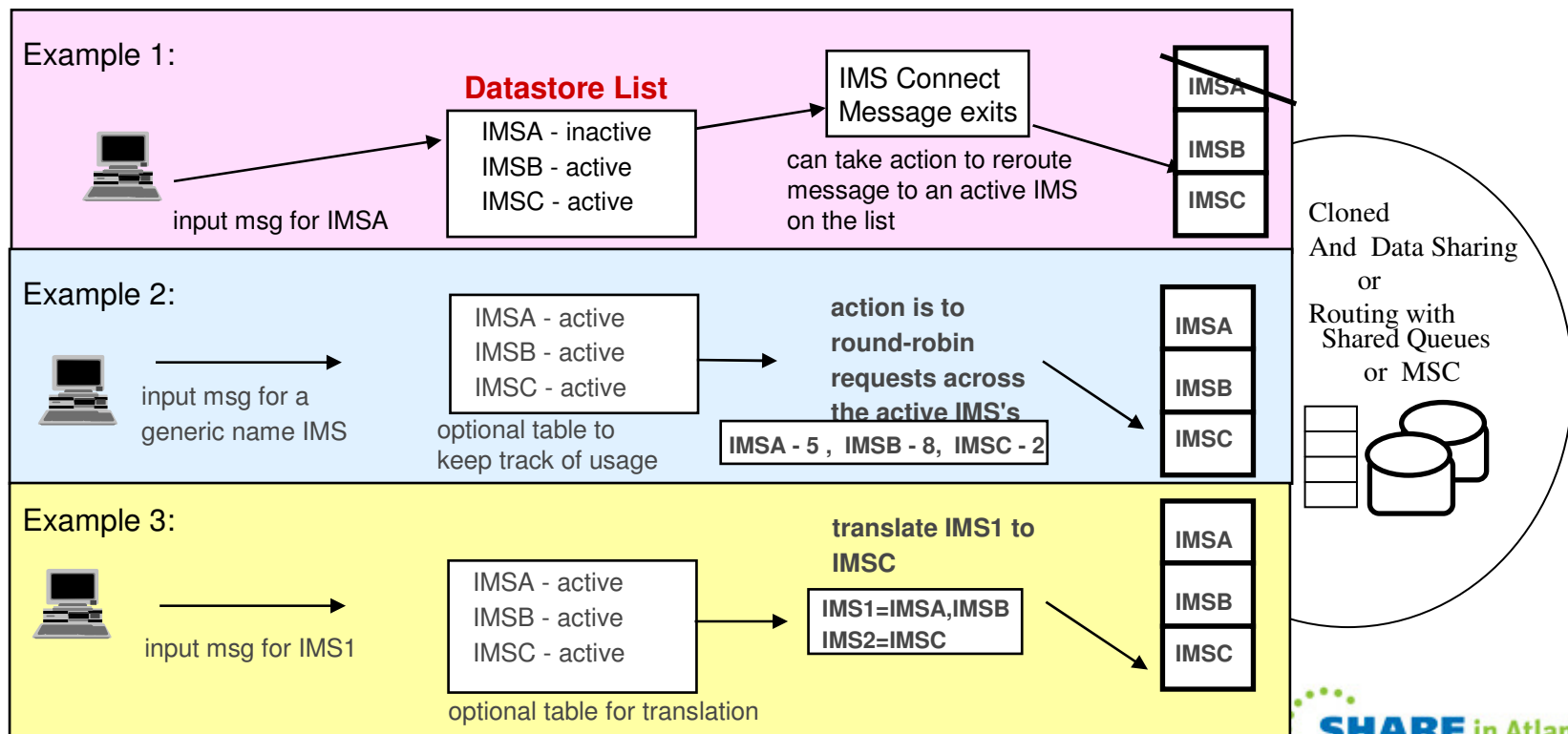
IMS Connect ...

- In a Sysplex Distributor environment



IMS Connect Workload Balancing and Failover

- Once a message destination is resolved to a particular z/OS host and IMS Connect system
 - IMS Connect can access multiple IMS Systems (datastores)
 - Message exits can reroute a message to a different target IMS
 - The Datastore table provides information as to which systems are active



Routing Examples

- Two tables available to IMS Connect Message exits
 - **INIT TABLE**
 - Points to the datastore table
 - Allows user data to be stored
 - **DATASTORE TABLE** (datastore = an IMS system)
 - Contains datastore id's, status (active or inactive) and optional user data
- User Initialization Exit Routine (HWSUINIT)
 - Driven during initialization and termination
 - Load user table(s) and obtain any needed storage
 - Add user data to INIT and DATASTORE tables
 - e.g., define alternate IMS systems
- IMS Connect provides the interface
 - IMS Connect message exits that are provided do not take advantage of the capability but can be enhanced to do so
 - Plug-ins such as **IMS Connect Extensions (CEX)** provide routing support


Note

- Although the Sysplex Distributor is an efficient mechanism for workload balancing
 - **Balancing only occurs when a new connection is being established**
 - If an IMS Connect region fails
 - *New connection requests are automatically routed to the remaining active IMS Connect regions and workload balancing occurs among the active regions*
 - When the failed IMS Connect is restarted
 - *Connections that are already active with other IMS Connect regions are NOT re-balanced*
 - *Only NEW connections are routed to the restarted region*
 - **Implication:**
 - Many of the connections with IMS Connect are persistent
 - *The environment after an IMS Connect has failed and been restarted, might be unbalanced until enough new connections have been requested.*

Considerations

- Connection failures – What happens to the output messages?
 - Commit mode
 - Send-then-Commit (CM1)
 - *Sync_level=none messages are discarded*
 - *119 abend only for IMS Connect failure*
 - *Sync_level=confirm messages result in 119 abends*
 - Commit-then-send (CM0)
 - *Always Sync_level=confirm*
 - *Undelivered messages stay queued in IMS*
 - *Retrieved with a ResumeTPIPE request from remote client*

Considerations ...

- Load Balancing --- vs --- CM0 messages on the IMS queue
 - Includes undelivered IOPCB messages along with all unsolicited ALTPCB messages 
 - Resume TPIPE request to retrieve the output message
 - Must be routed to IMS Connect that directly or indirectly triggered the output
 - On a new connection, the load balancing function may route the request to a different IMS Connect instance
 - In a Shared Queues environment
 - The IMS system that processes an ALTPCB message must also deliver it
 - *IMS Connect needs to route the Resume TPIPE request to that specific IMS instance*

Considerations ...

- Load balancing --- vs --- CM0 messages on the IMS queue ...

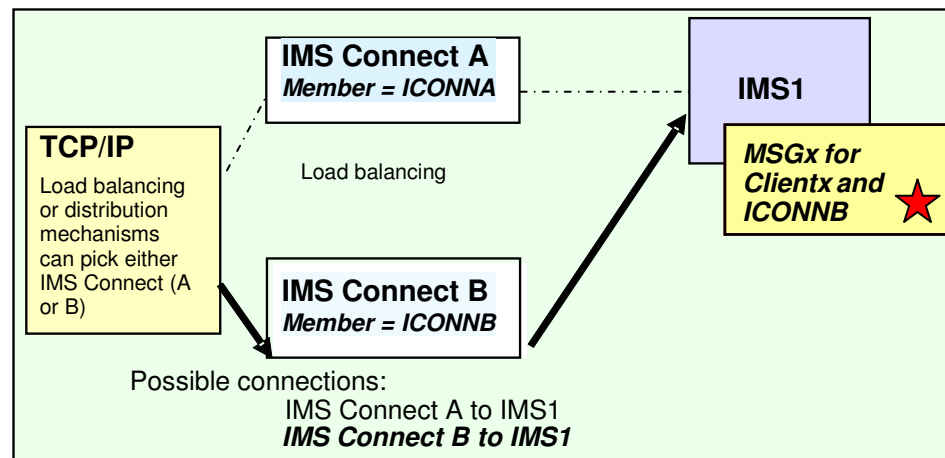
Clientx

Resume TPIPE
for clientx through
IMS Connect (TCP/IP
generic name)
and IMS

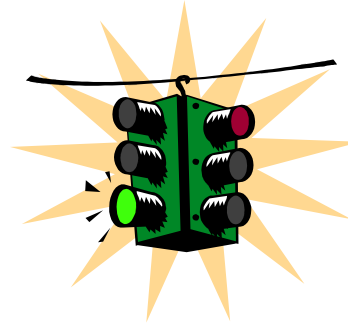
Will only retrieve MSGx if
the connection is correctly
established with IMS
Connect B and IMS1



If connection is established
through any other path, this
program will either
disconnect or timeout at
some point.



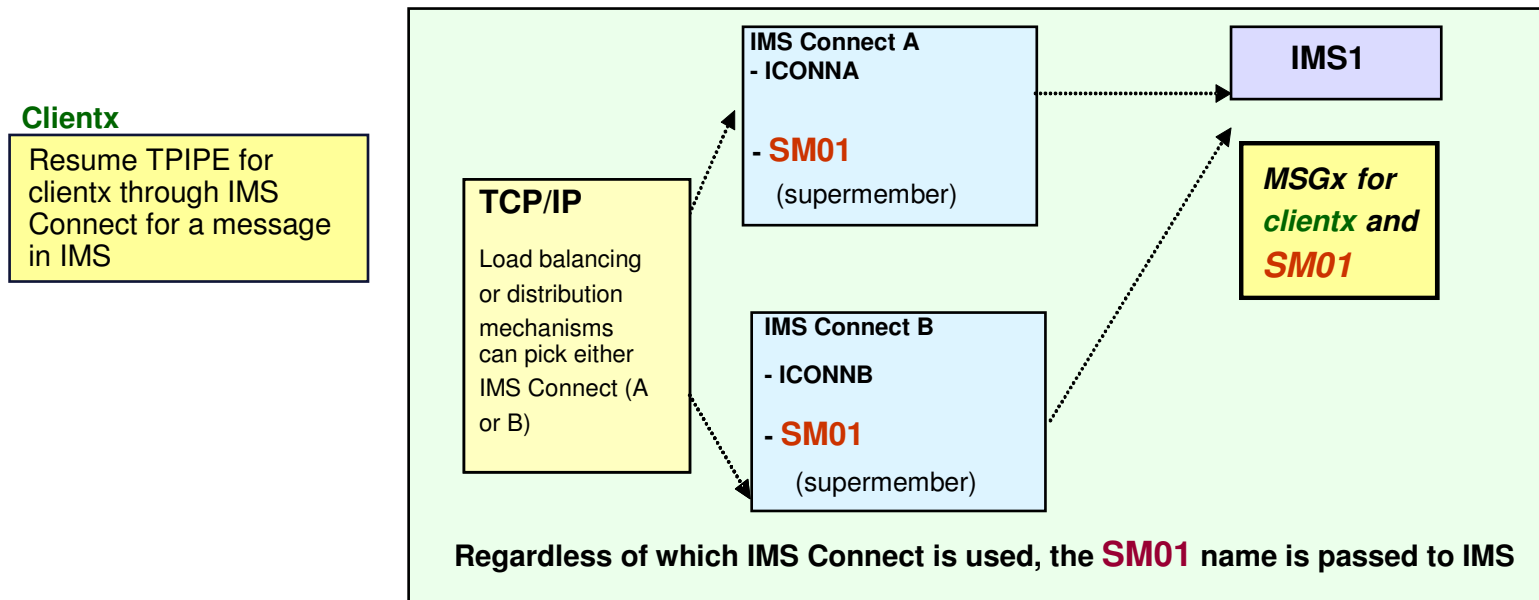
Supermember Support



- Solution
- **Supermember Support (OPTIONAL)**
 - A group name given to a set of IMS Connect instances
 - *Any IMS Connect can retrieve the message*
 - Generic structure name in IMS on which the messages are queued
 - *IMS Connect systems are recognized by IMS by both their specific name and supermember name*
 - *For Shared Queues, no affinity to an IMS system*
 - *Any IMS can deliver the message*
 - *Make sure you are current on maintenance !!!*

Supermember Support ...

- With supermember support, Resume TPIPE requests can retrieve any of the queued asynchronous messages regardless of the connection path



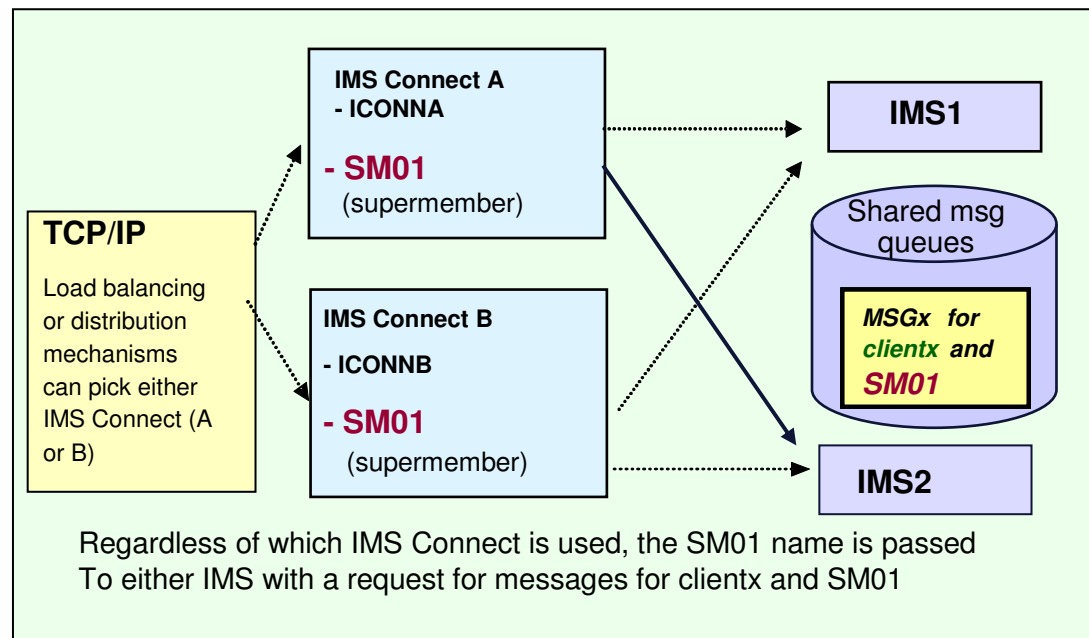
- Note: a single IMS with multiple IMS Connects does not require shared queues, data sharing, etc...

Supermember Support ...

- But... if multiple IMS systems are involved
 - Implement Shared Queues to allow any IMS to deliver the message even if the IMS that originally created the message is unavailable

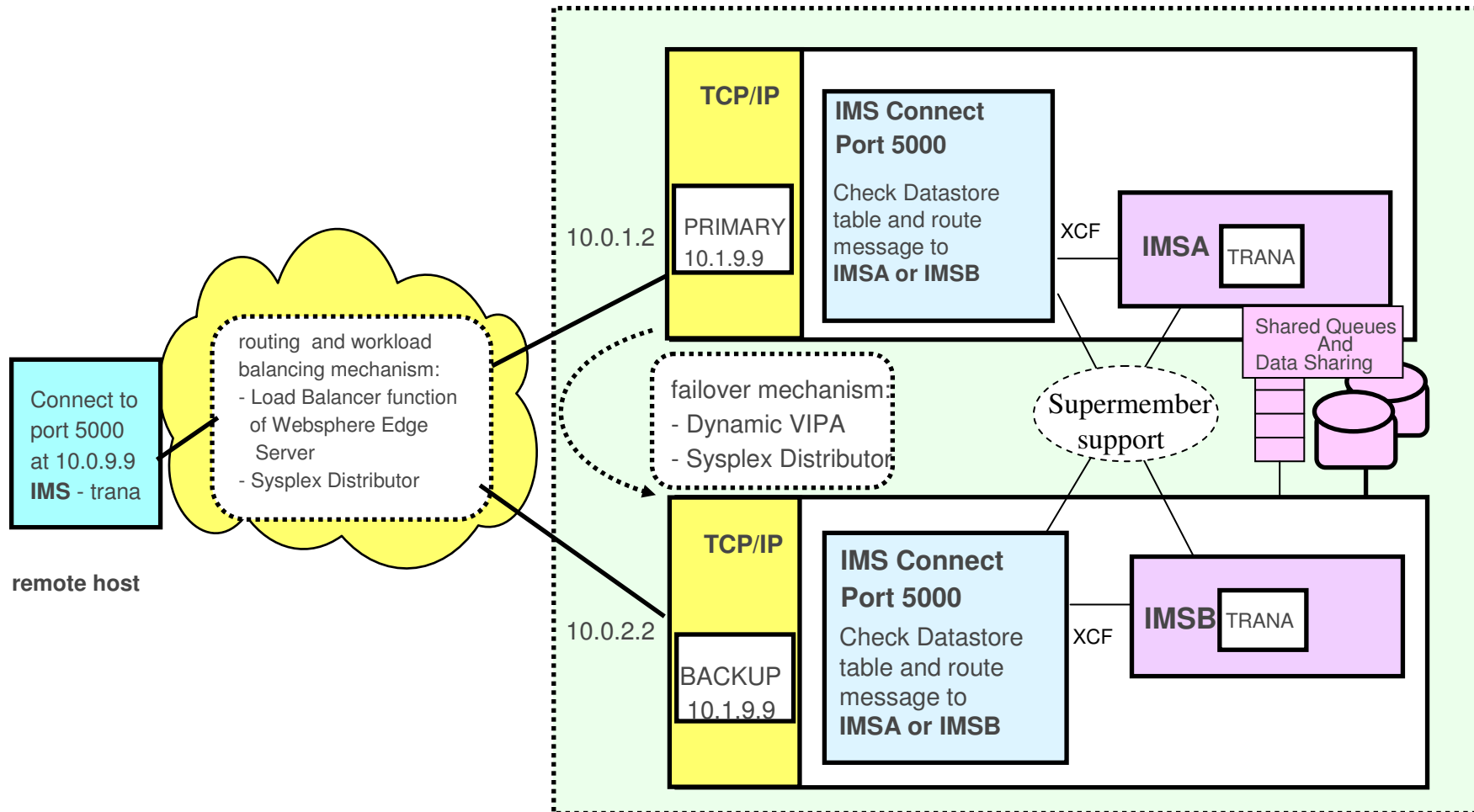
Clientx

Resume TPIPE for clientx through IMS Connect for a message in IMS



So... for Transaction Access

- Through sysplex distribution, and supermember support



Multiple IMS Systems and Shared Queues Support



- If the IMS systems are cloned
 - IMS Connect routing mechanisms can send messages to different IMS's for processing
- Or, with the use of IMS Shared Queues
 - CM0 (Commit-then-Send) messages can process in any back-end
 - CM1 (Send-then-Commit) synchronous processing in a back-end system
 - Uses zOS RRS (Resource Recovery Services)

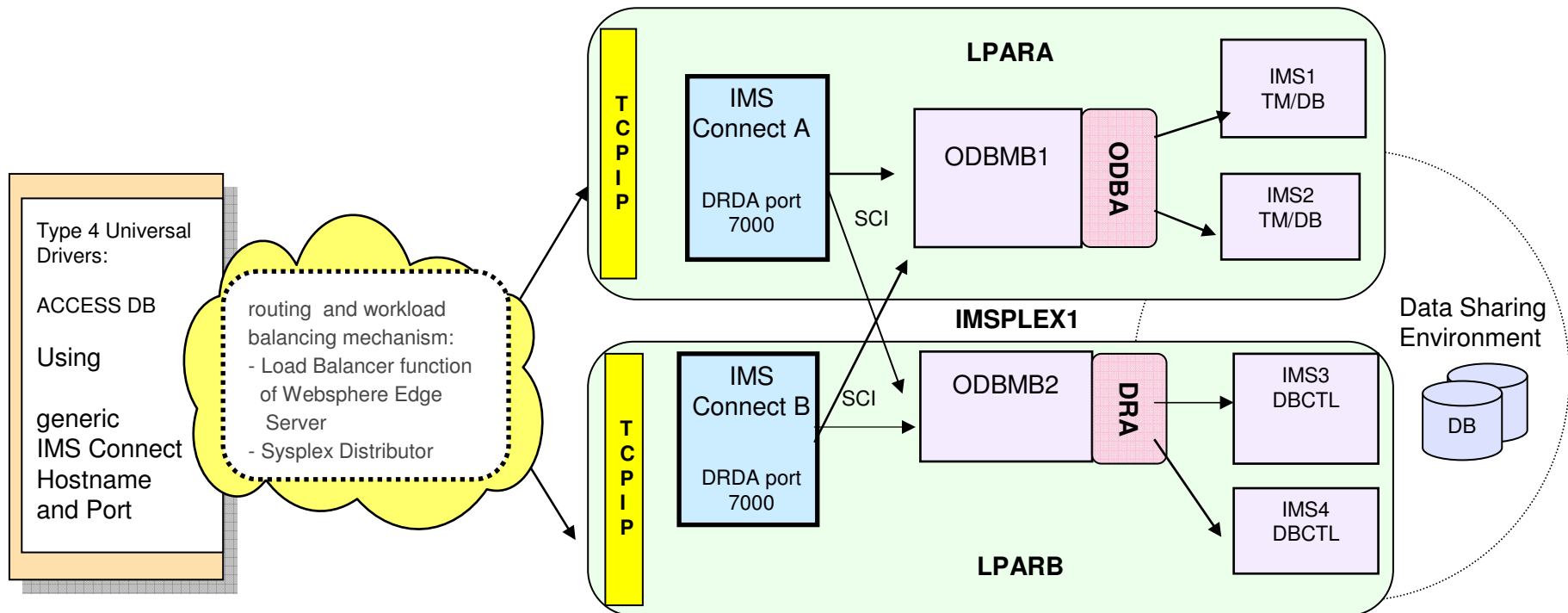
Multiple IMS Systems and Shared Queues Support ...



- Shared Queues ...
 - **IMS 11 (without RRS)**
 - No Synchronous processing in the back-end, **UNLESS**
 - *DFSMSCE0 specifies MSCEFL2 = MSCE2FAF*
 - *Requests CM1 Sync (NONE|CONFIRM) to process in the IMS B-E*
 - *IMS maintains connection to the client on the front-end system while processing the message as a non-APPC/OTMA message within the IMSplex environment*
 - *Response message is converted back to APPC/OTMA synchronous mode, and the response is queued to the client.*
 - *MSC must be defined (although it is not used)*
 - **IMS 12**
 - New DFSDCxxx option to request synchronous Sync (NONE | CONFIRM) requests be processed on a back-end system
 - *Using XCF services (not RRS)*

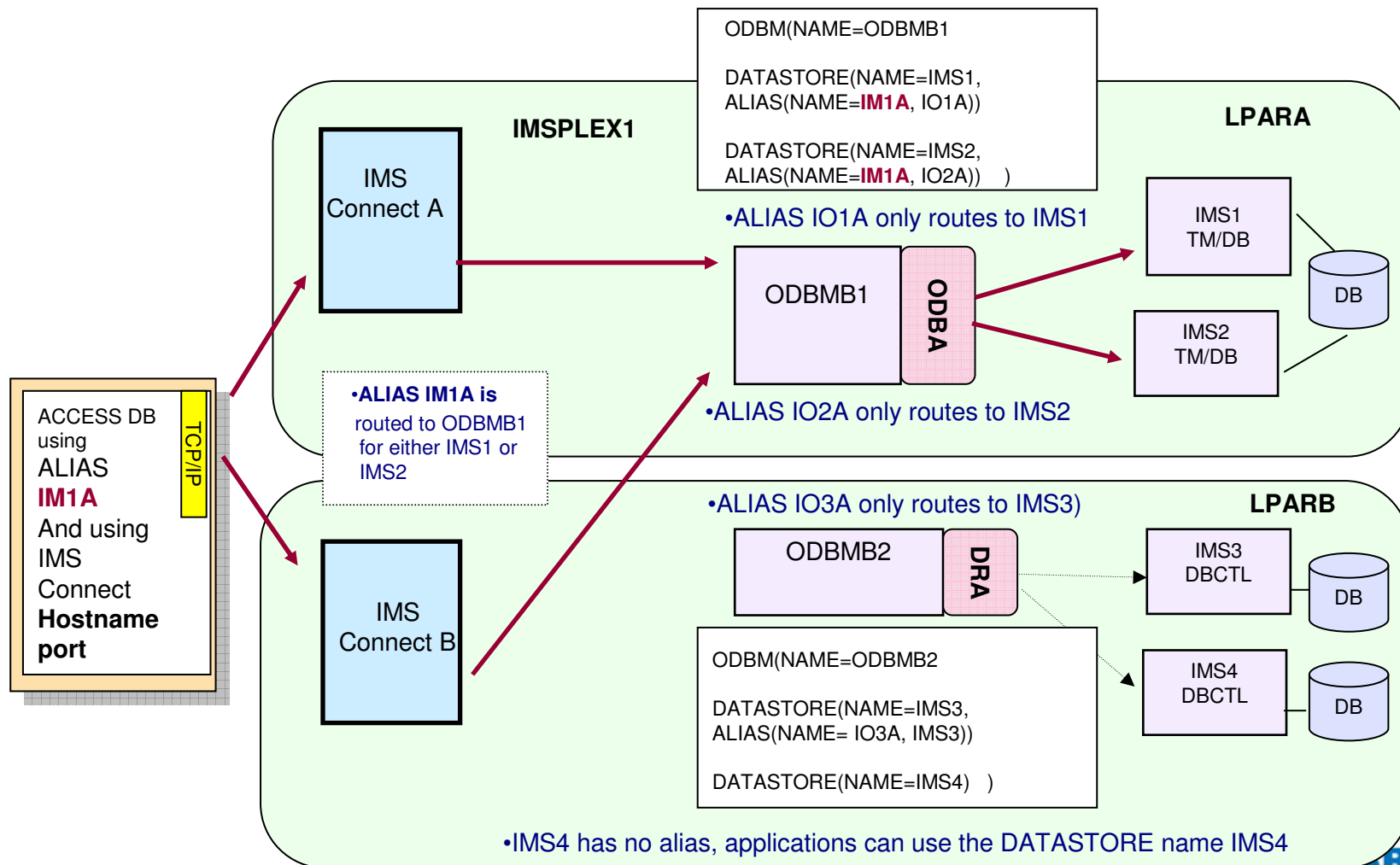
How about Database Access?

- With IMS 11, IMS Connect along with the Open Database Manager (ODBM) provide IMS database access
 - IMS Connect supports load balancing and sysplex distribution for this environment
 - The DRDA port number can also be shared across multiple instances



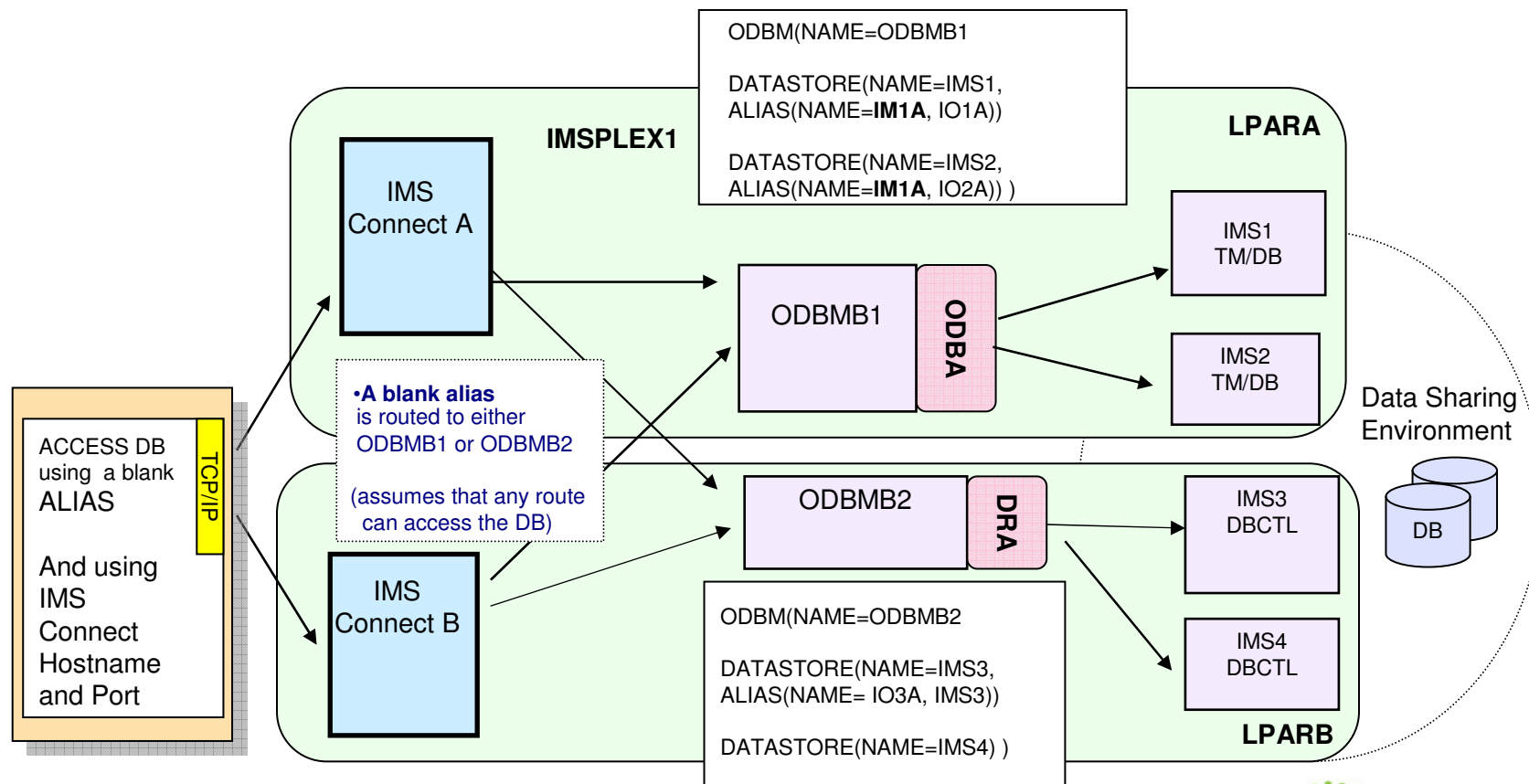
Database Access

- Calling programs can use an **alias** name to access IMS data stores without needing to know the actual IMSID of the IMS data store



Database Access ...

- Without an alias, IMS Connect routes incoming connections in round-robin fashion among all the instances of ODBM within the IMSplex and across the IMS systems within those ODBMs

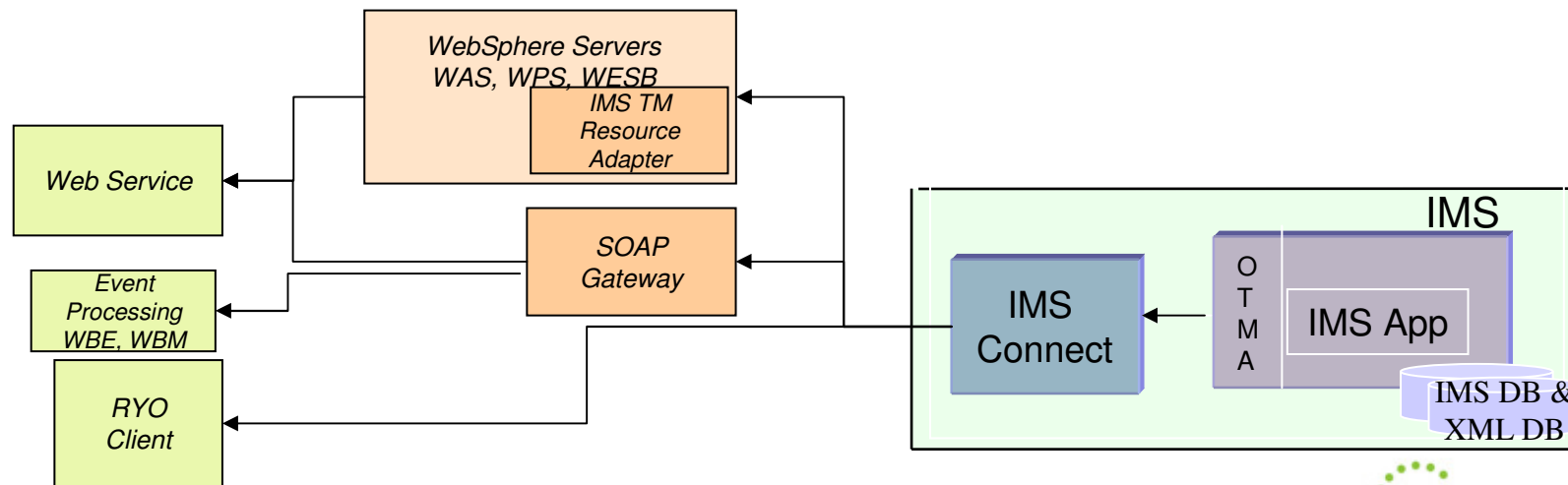


Database Access ...

- IMS Connect has an internal table
 - Keeps track of ODBMS and the IMS datastores and aliases
- IMS Connect DB Routing Exit routine (HWSROUT0)
 - Can determine or change the destination of a database access request
 - Select an ODBM by its name
 - Allow IMS Connect to select an ODBM instance
 - *By alias name*
 - *If only one ODBM has specified the alias, the request is routed to that ODBM*
 - *If the alias is associated with multiple ODBMs then IMS Connect uses a round-robin technique to route the request across those resources.*
 - *If the alias is blanks then IMS Connect round-robins the request across all the ODBM resources in the IMSplex.*
 - Override the alias provided in the incoming request

And for Outbound Requests

- IMS applications as clients to remote web services
 - IMS DL/I call “ICAL” sends messages through IMS Connect to:
 - *EJB/MDB on a JEE server, e.g., WebSphere Application Service*
 - *Soap clients through IMS ES Soap Gateway*
 - *RYO clients*
 - Remote systems need to issue
 - *Resume TPIPE to retrieve messages*
 - *IMS TM RA for JEE, IMS ES Soap Gateway, RYO*

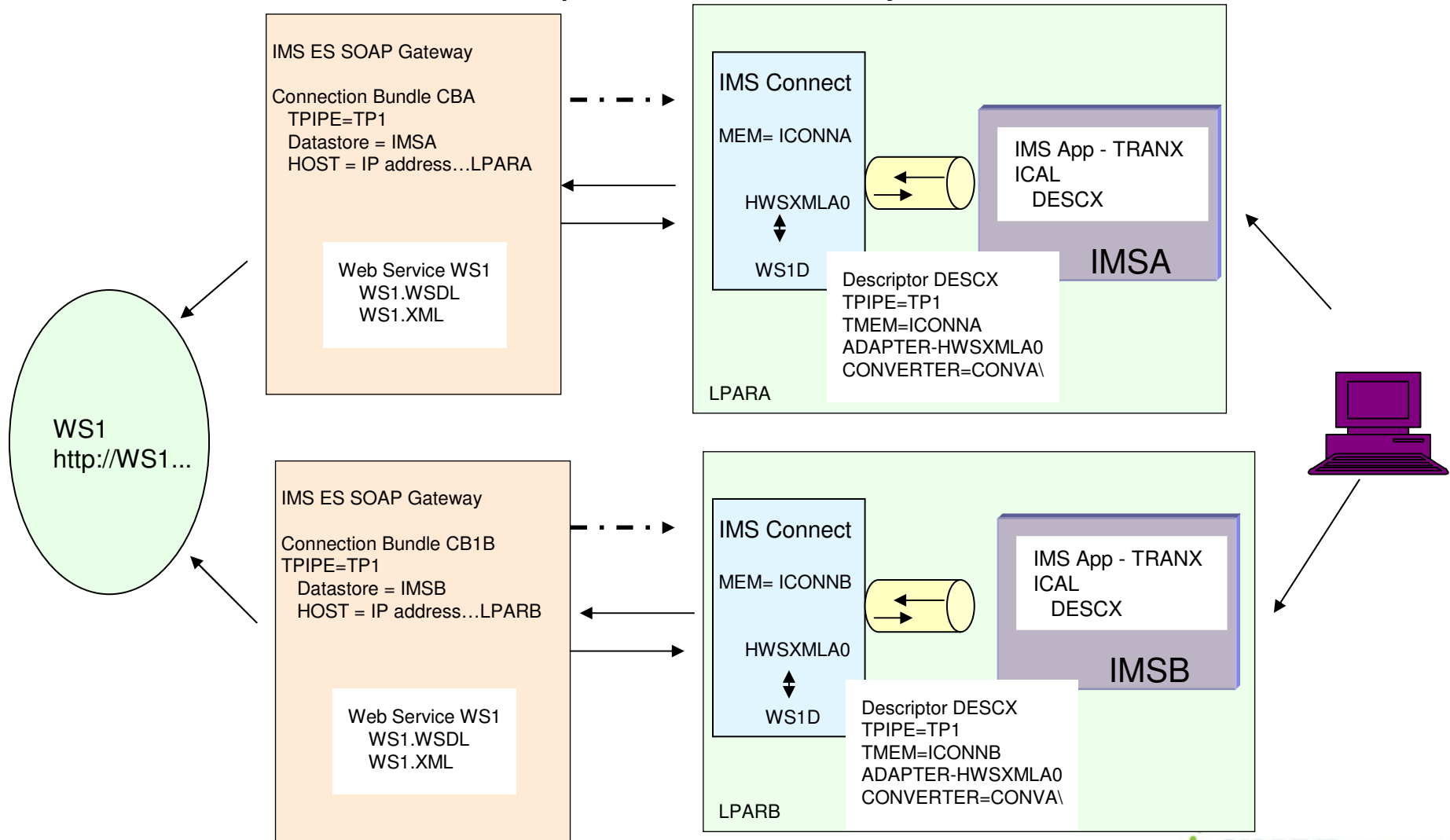


And for Outbound Requests ...

- Multiple applications/servers may be configured to pull IMS callout messages on the same destination for availability
 - If one server fails, the other server can continue to pull callout messages from IMS
 - Servers are pre-conditioned to issue Resume TPIPE requests
 - Failover support for OTMA allows queuing of Resume TPIPEs
 - IMS 10 APAR PK80758/ PTF UK50686
 - IMS V11 APAR PK91374/ PTF UK52368
 - First Resume TPIPE is active until satisfied and subsequent Resume TPIPEs are queued
 - IMS Command: /DISPLAY TMEMBER TPIPE SYNC
 - *Shows the number of queued “Resume Tpipe on Hold Queue” requests (RTQ) and the active Resume Tpipe status*

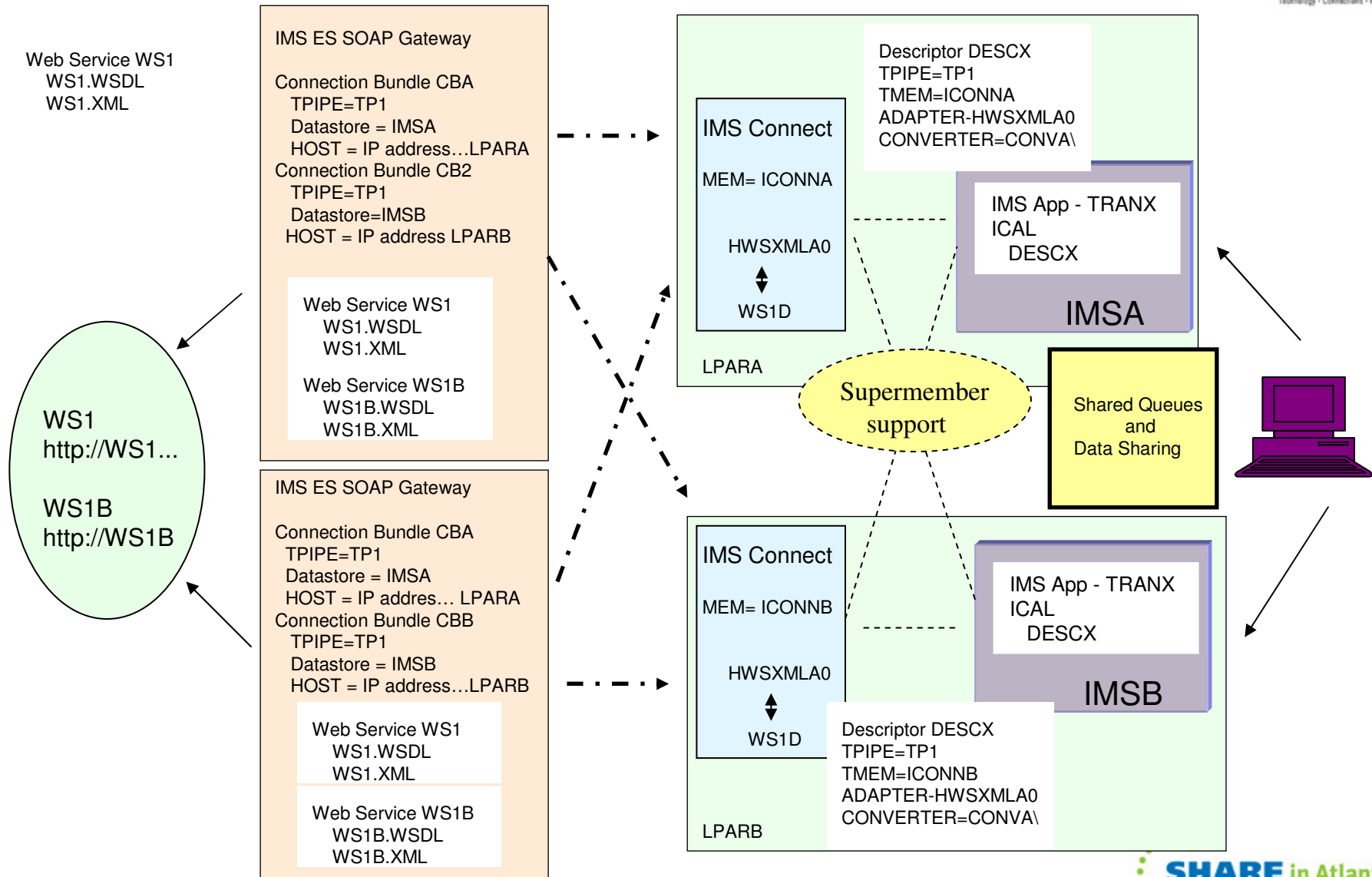
And for Outbound Requests ...

Step 1 - Redundancy



And for Outbound Requests ...

Target – Redundancy with Failover



In a Nutshell

- IMS Connect – as a standard TCP/IP server application
 - Fully participates in IP spraying and load balancing techniques
 - VIPA, Dynamic VIPA, Sysplex Distributor
 - Provides mechanisms to route requests to any IMS in the sysplex
 - Transaction resources
 - *Support for unsolicited messages with the supermember capability*
 - *Exit interfaces to route/reroute messages*
 - *Across LPAR boundaries*
 - Database resources
 - *Provides round-robin routing*
 - *Supports aliasing*
 - *Provides exit interfaces to impact routing decisions*