

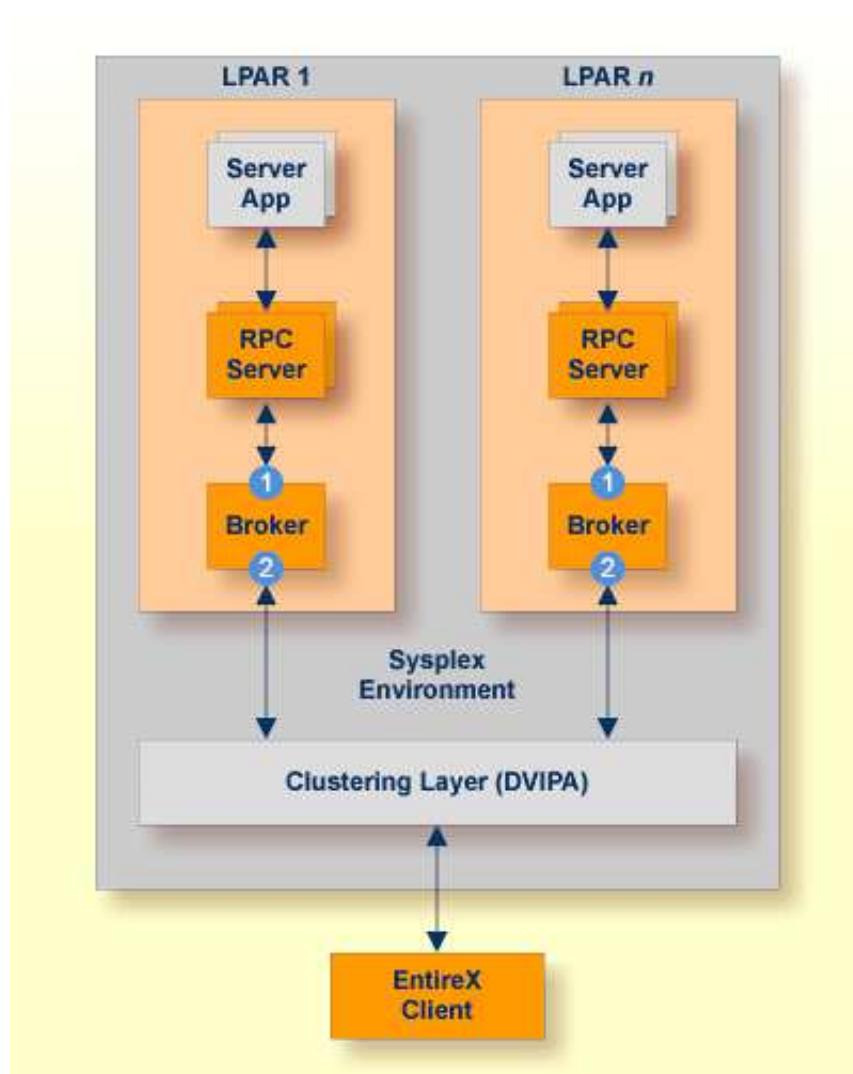
Setting up your EntireX Environment for Sysplex

Scenario: "I want to use Sysplex for my high availability cluster."

This chapter covers the following topics:

- Setting Broker Attributes
 - Broker Cluster Considerations
 - Configuring Redundant RPC Servers
 - Verifying your Configuration
 - Managing Brokers and RPC Servers
-

Setting Broker Attributes



Segmenting dynamic workload from static server and management topology is critically important. Using broker TCP/IP-specific attributes, define two separate connection points:

- One for RPC server-to-broker and admin connections. ①
- The second for client workload connections. ②

See *TCP/IP-specific Attributes* under *Broker Attributes*. Sample attribute file settings:

	Sample Attribute File Settings	Note
①	PORT=1972	In this example, the HOST is not defined, so the default setting will be used (localhost).
②	HOST=10.20.74.103 (or DNS) PORT=1811	In this example, the HOST stack is the virtual IP address. The PORT will be shared by other brokers in the cluster.

Broker Cluster Considerations

Platform-independent Considerations

We recommend the following:

- Share configurations - you will want to consolidate as many configuration parameters as possible in the attribute setting. Keep separate yet similar attribute files.
- Isolate workload listeners from management listeners.
- Monitor Brokers through SMH.

z/OS-specific Considerations

- Use Started Task names that match EntireX Broker naming conventions and have logical context.
- z/OS supports multiple TCP/IP stacks per LPAR. EntireX Broker supports up to eight separate listeners on the same or different stacks.

Configuring Redundant RPC Servers

In addition to broker redundancy, you also need to configure your RPC servers for redundant operations. We recommend the following best practices when setting up your RPC servers:

- **General Hints**
 - Make sure your definitions for `CLASS/SERVER/SERVICE` are identical across the clustered brokers. Using identical service names will allow the broker to round-robin messages to each of the connected RPC server instances.
 - For troubleshooting purposes, and if your site allows this, you can optionally use a different user ID for each RPC server.
 - RPC servers are typically monitored using SMH as services of a broker. Optionally, for example for troubleshooting purposes, the RPC servers can be configured with a unique TCP port number for SMH.

Note:

SMH port is not supported by the Natural RPC server.

- Establish the broker connection using the static `Broker name:port` definition.
- **CICS RPC Server**
 - CICS RPC Server instances can be configured through the *ERXMAIN Macro* that contains default settings and naming conventions. You can use the *RPC Online Maintenance Facility* to control and monitor these instances online.
 - Establish an unused port through the CICS default stack to start the SMH Listener. See *SMH*.

- If you are using server-side mapping files, share the server-side mapping container across RPC server instances. A server-side mapping file is an EntireX Workbench file with extension .svm. See *Server Mapping Files for COBOL*. Use VSAM RLS or simple share options to keep a single image of the server-side mapping container across all or groups of RPC servers (for example CICS, IMS, Batch). See *Job Replacement Parameters* under *Simplified z/OS Installation Method* in the z/OS Installation documentation.
- **Batch RPC Server**
 - Make sure logging is distinguishable through each RPC Server instance for troubleshooting purposes (e.g. JES SYSOUT).
- **Natural RPC Server**
 - Maintain separate parameter files for each Natural RPC Server instance.

Verifying your Configuration

Here are some sample commands for verifying your cluster environment:

➤ To display the Netstat Dynamic VIPA status

- Enter command

```
D TCPIP,TCPIPEXB,N,VIPADYN
```

Status must be "ACTIVE".

```
EZZ2500I NETSTAT CS V1R13 TCPIPEXB 187
DYNAMIC VIPA:
  IP ADDRESS      ADDRESSMASK      STATUS      ORIGINATION      DISTSTAT
  10.20.74.103    255.255.255.0   ACTIVE      VIPADEFINE        DIST/DEST
  ACTTIME:        11/17/2011 09:29:13
```

➤ To display the Netstat Dynamic VIPA info

- Enter command

```
D TCPIP,TCPIPEXB,N,VIPADCFG
```

This shows multiple static definitions to one dynamic definition:

```
EZZ2500I NETSTAT CS V1R13 TCPIPEXB 190
DYNAMIC VIPA INFORMATION:
VIPA DEFINE:
  IP ADDRESS      ADDRESSMASK      MOVEABLE      SRVMGR  FLG
  -----
  10.20.74.103    255.255.255.0   IMMEDIATE    NO
VIPA DISTRIBUTE:
  IP ADDRESS      PORT      XCF ADDRESS      SYSPT  TIMAFF  FLG
  -----
  10.20.74.103    18000     10.20.74.104     NO     NO
  10.20.74.103    18000     10.20.74.114     NO     NO
```

 **To display Sysplex VIPA Dynamic configuration**

- Enter command

```
D TCPIP,TCPIPEXB,SYS,VIPAD
```

This verifies that multiple LPARs have been defined (MVSNAMEs):

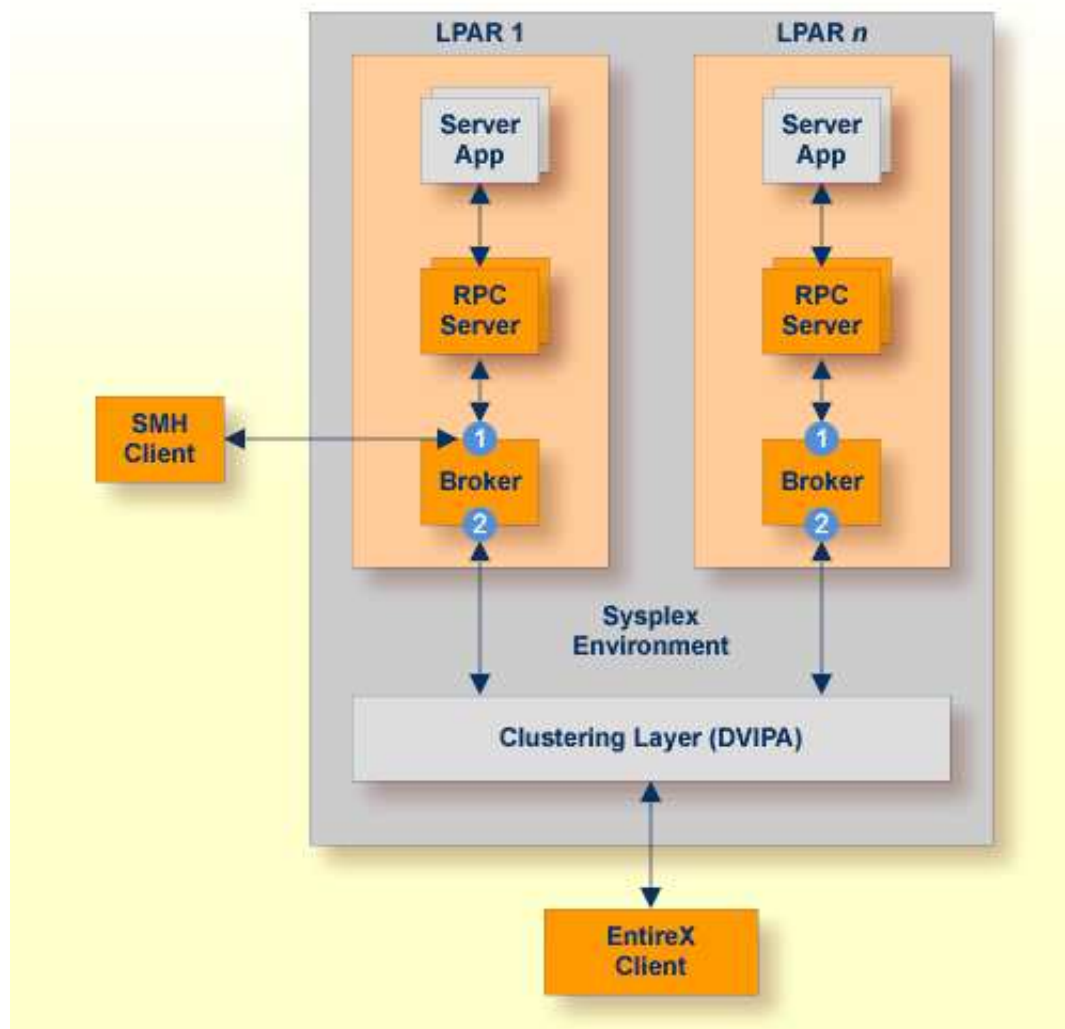
```
EZZ8260I SYSPLEX CS V1R13 166
VIPA DYNAMIC DISPLAY FROM TCPIPEXB AT AHST
IPADDR: 10.20.74.103 LINKNAME: VIPL0A144A67
ORIGIN: VIPADEFINE
TCPNAME MVSNAME STATUS RANK ADDRESS MASK NETWORK PREFIX
-----
TCPIPEXB AHST ACTIVE 255.255.255.0 10.20.74.0
TCPIPEXB BHST BACKUP 001
```

Managing Brokers and RPC Servers

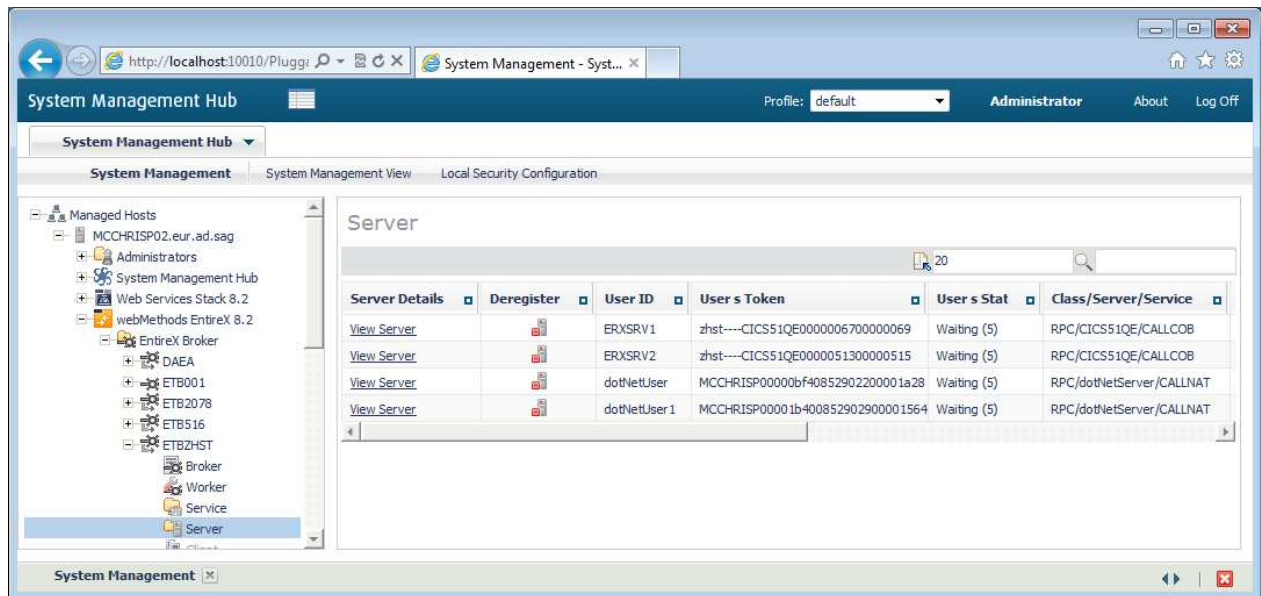
- Using System Management Hub
- Lifecycle Management of Brokers
- Lifecycle Management of RPC Servers

Using System Management Hub

Use the System Management Hub to monitor the status of the broker and RPC server instances using their respective `address:port` connections. Set up each connection with logical instance names.



The following screen shows two pairs of redundant RPC servers registered to the same broker from the Server view:



- Each broker requires a static TCP port for RPC server and management communications.
- This must be maintained separately from the broker's virtual IP address configuration.
- Onboard TCP-based RPC server connections *can* support HiperSockets requiring segregated static addresses (otherwise the address can be shared).

Lifecycle Management of Brokers

An important aspect of high availability is during planned maintenance events such as lifecycle management, applying software fixes, or modifying the number of runtime instances in the cluster. Using a virtual IP networking approach for broker clustering allows high availability to the overall working system while applying these tasks.

See *Starting and Stopping the Broker* in the z/OS administration documentation.

You can ping and stop a broker using the command-line utility *ETBCMD*.

Lifecycle Management of RPC Servers

Starting, pinging and stopping an RPC server is described in the EntireX documentation for CICS | Batch | IMS RPC servers.

See also *Operating a Natural RPC Environment* in the Natural documentation.