IBM WebSphere Software

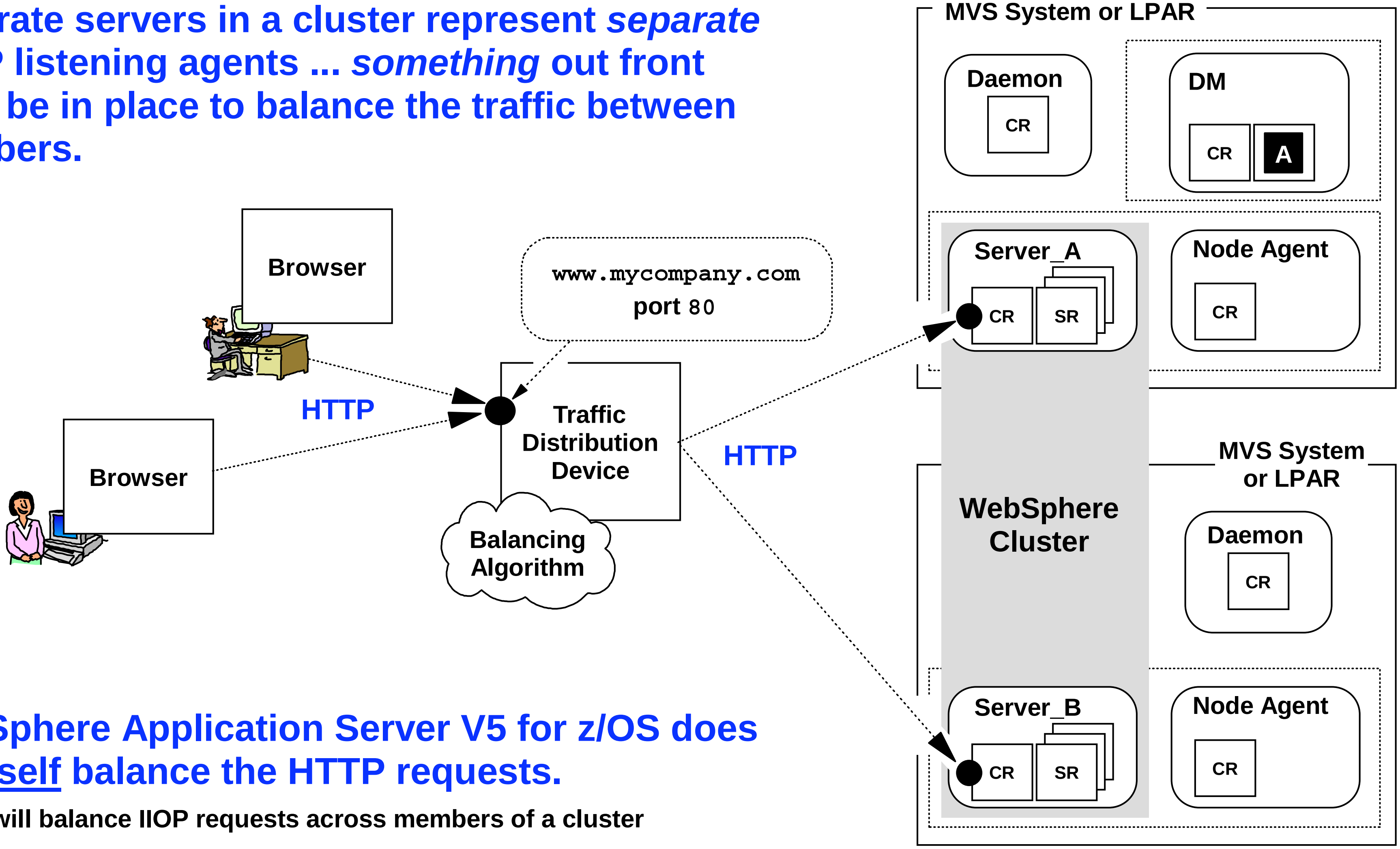
WebSphere Application Server for z/OS and OS/390

Configuring the WebSphere Plug-in with WebSphere V5 for z/OS

IBM Americas Advanced Technical Support -- Washington Systems Center
Gaithersburg, MD, USA

HTTP Requests

Separate servers in a cluster represent *separate* HTTP listening agents ... *something* out front must be in place to balance the traffic between members.



WebSphere Application Server V5 for z/OS does not <u>itself</u> balance the HTTP requests.

But it will balance IIOP requests across members of a cluster

Lots of different solutions to balance traffic.

Topic here: "WebSphere HTTP Plugin for z/OS"

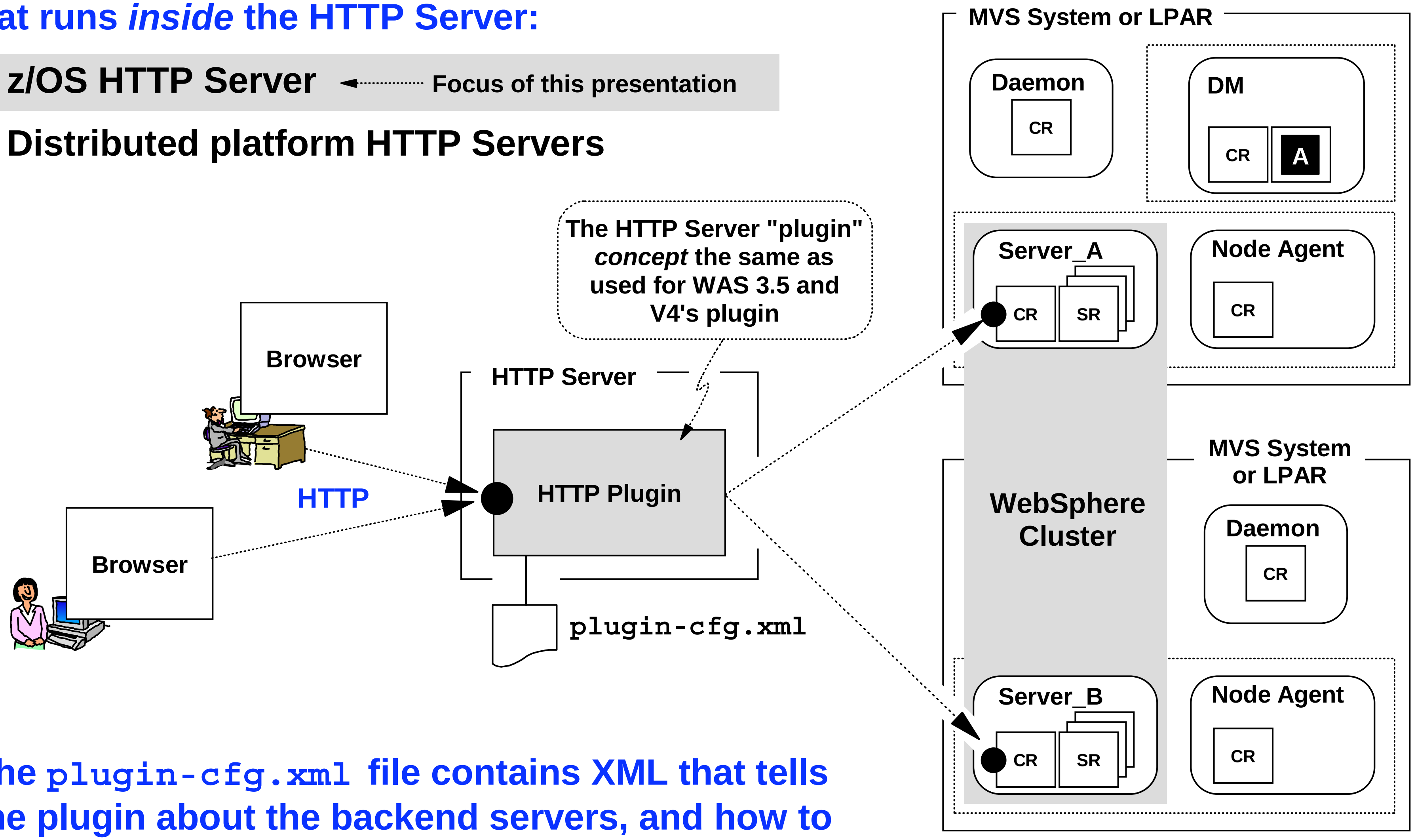
Agenda

- Answer some up-front questions about the "WebSphere HTTP Plugin for z/OS"
- Briefly discuss what "Session Affinity" is
- Show how the "HTTP Plugin" is configured in the HTTP Server
- Take a look at the contents of the `plugin-cfg.xml` file
- Show how WebSphere Application Server for z/OS Version 5 can automatically generate the `plugin-cfg.xml` file
- Review some troubleshooting and problem determination tips
- Finish up with a quick illustration of a blended configuration: "HTTP Plugin" + Sysplex Distributor

WebSphere HTTP Plugin for z/OS

Code provided with WebSphere for z/OS Version 5 that runs *inside* the HTTP Server:

- z/OS HTTP Server ← Focus of this presentation
- Distributed platform HTTP Servers

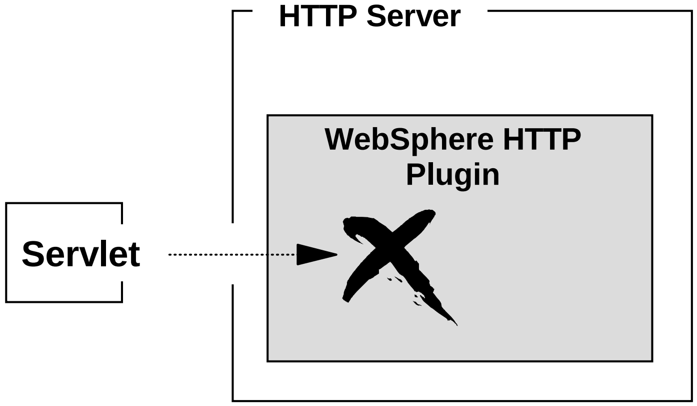


The HTTP Server "plugin" *concept* the same as used for WAS 3.5 and V4's plugin

The `plugin-cfg.xml` file contains XML that tells the plugin about the backend servers, and how to route requests to maintain "session affinity"

What the new HTTP Plugin is NOT

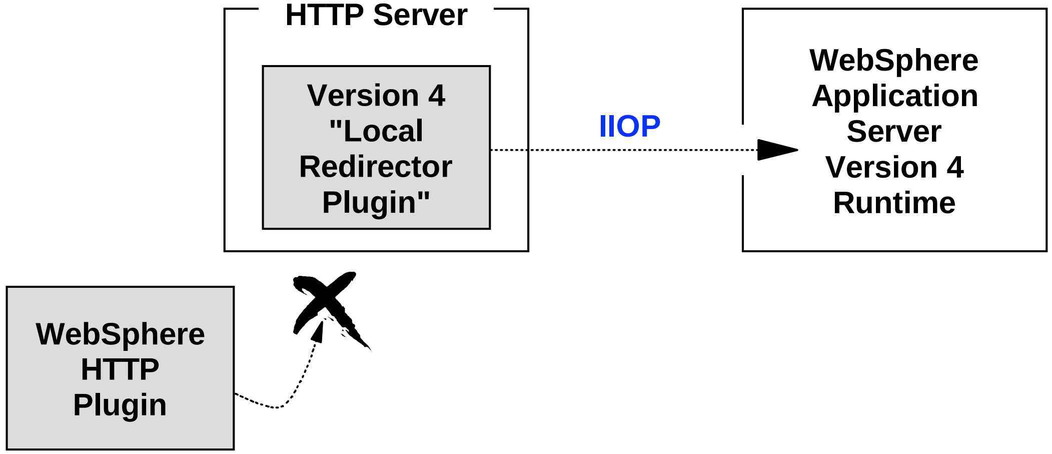
It is not a servlet execution environment



In this sense it is *different* from:

- WebSphere Application Server for OS/390 and z/OS Version 3.5
- "Local Redirector Plugin" that came with WebSphere Application Server for OS/390 and z/OS Version 4

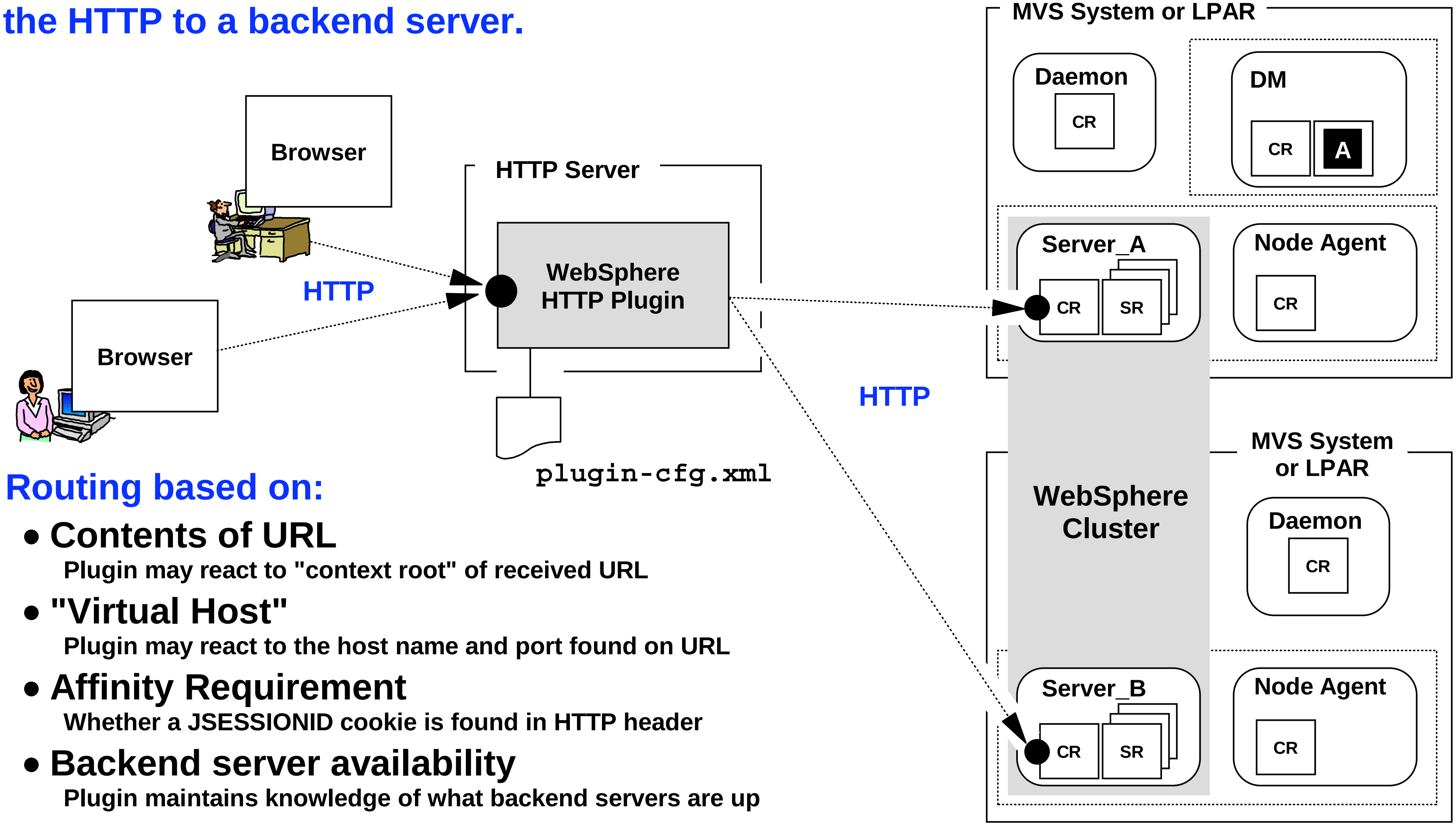
It is not a replacement of the "Local Redirector Plugin" used with WAS V4


HTTP Server
Version 4 "Local Redirector Plugin"
IIOP
WebSphere Application Server Version 4 Runtime
WebSphere HTTP Plugin

WebSphere HTTP Plugin flows HTTP, not IIOP.

What the WebSphere HTTP Plugin IS

A device that takes HTTP inbound and re-routes the HTTP to a backend server.

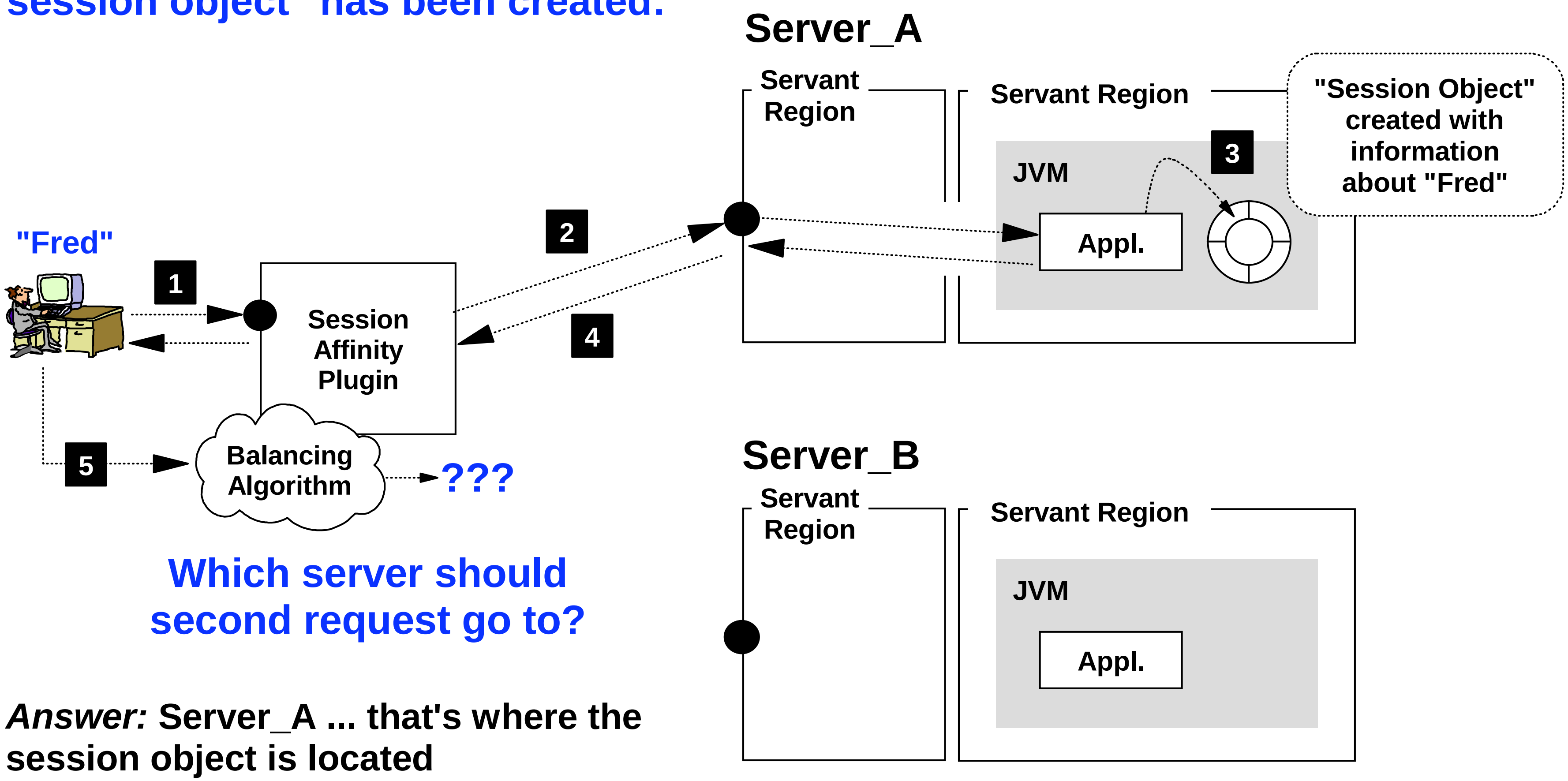


Routing based on:

- **Contents of URL**
  Plugin may react to "context root" of received URL
- **"Virtual Host"**
  Plugin may react to the host name and port found on URL
- **Affinity Requirement**
  Whether a JSESSIONID cookie is found in HTTP header
- **Backend server availability**
  Plugin maintains knowledge of what backend servers are up
- **"Weight" of each server in a WAS cluster**
  If "round-robin" distribution, then distribute based on defined "weight" of the servers in the cluster

What is "Session Affinity?"

"Session Affinity" is the routing of requests back to the server in which a client's "session object" has been created:



Which server should second request go to?

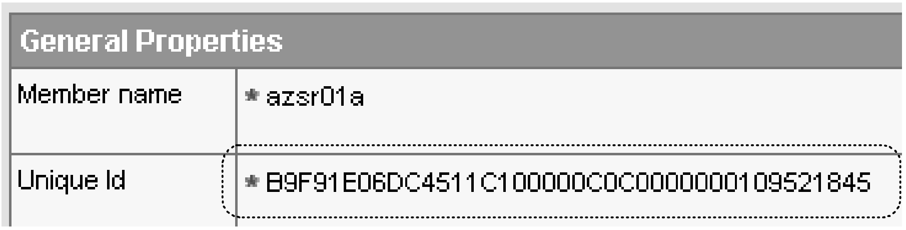
- *Answer:* Server_A ... that's where the session object is located
- *Question:* How does Plugin know which server to route second request to?
- *Answer:* based on information put in HTTP header ... the "Unique ID"

Each server in WebSphere is given a "Unique ID" ...

Cluster Members and "Unique ID"

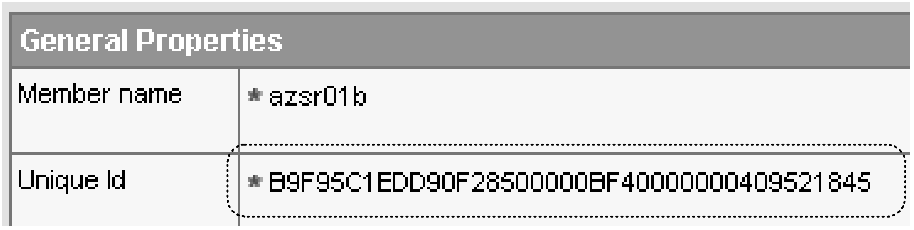
WebSphere assigns each server a "Unique ID", which can be seen in Admin Console:

Server: `azsr01a`



General Properties	
Member name	* azsr01a
Unique Id	* B9F91E06DC4511C100000C0C0000000109521845

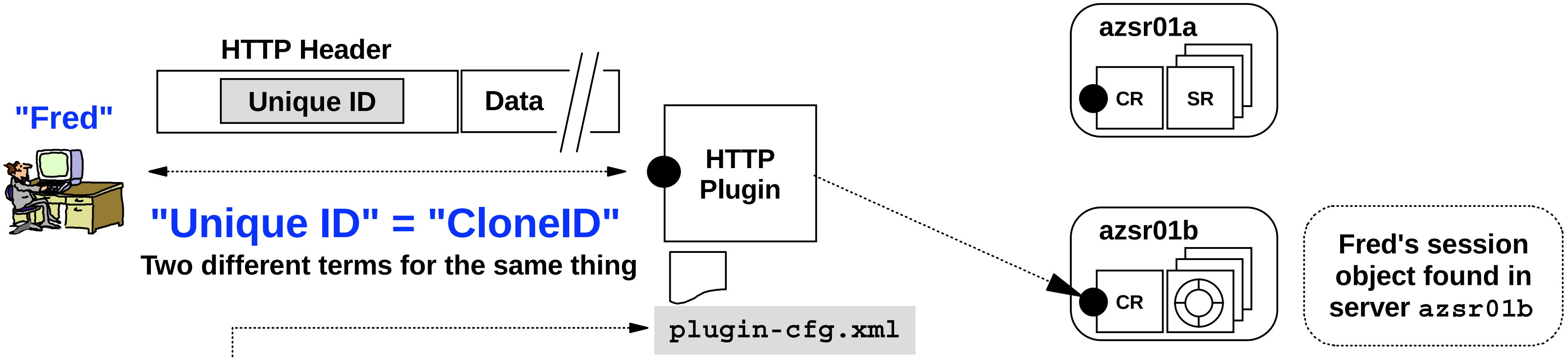
Server: `azsr01b`



General Properties	
Member name	* azsr01b
Unique Id	* B9F95C1EDD90F28500000BF40000000409521845

Unique ID's assigned to each

WebSphere will put "Unique ID" into HTTP header after a session object is created:



"Unique ID" = "CloneID"

Two different terms for the same thing

Fred's session object found in server `azsr01b`

File `plugin-cfg.xml` has information about where to route based on CloneID...

XML File Knows About "Unique IDs"

If Unique ID = B9F91E ...

Then send request to
wsc1.washington.ibm.com:9548

Server #1 in Cluster

Server #2 in Cluster

Else if Unique ID = ...

```
   :
   :
<ServerCluster Name="azsr01Cluster">

    <Server CloneID="B9F91E06DC4511C100000C0C0000000109521845"
        LoadBalanceWeight="2" Name="aznodea_azsr01a">
        <Transport Hostname="wsc1.washington.ibm.com" Port="9548" Protocol="http"/>
    </Server>

    <Server CloneID="B9F95C1EDD90F28500000BF40000000409521845"
        LoadBalanceWeight="2" Name="aznodeb_azsr01b">
        <Transport Hostname="wsc2.washington.ibm.com" Port="9548" Protocol="http"/>
    </Server>

    <PrimaryServers>
        <Server Name="aznodea_azsr01a"/>
        <Server Name="aznodeb_azsr01b"/>
    </PrimaryServers>
</ServerCluster>
    :
    :
```

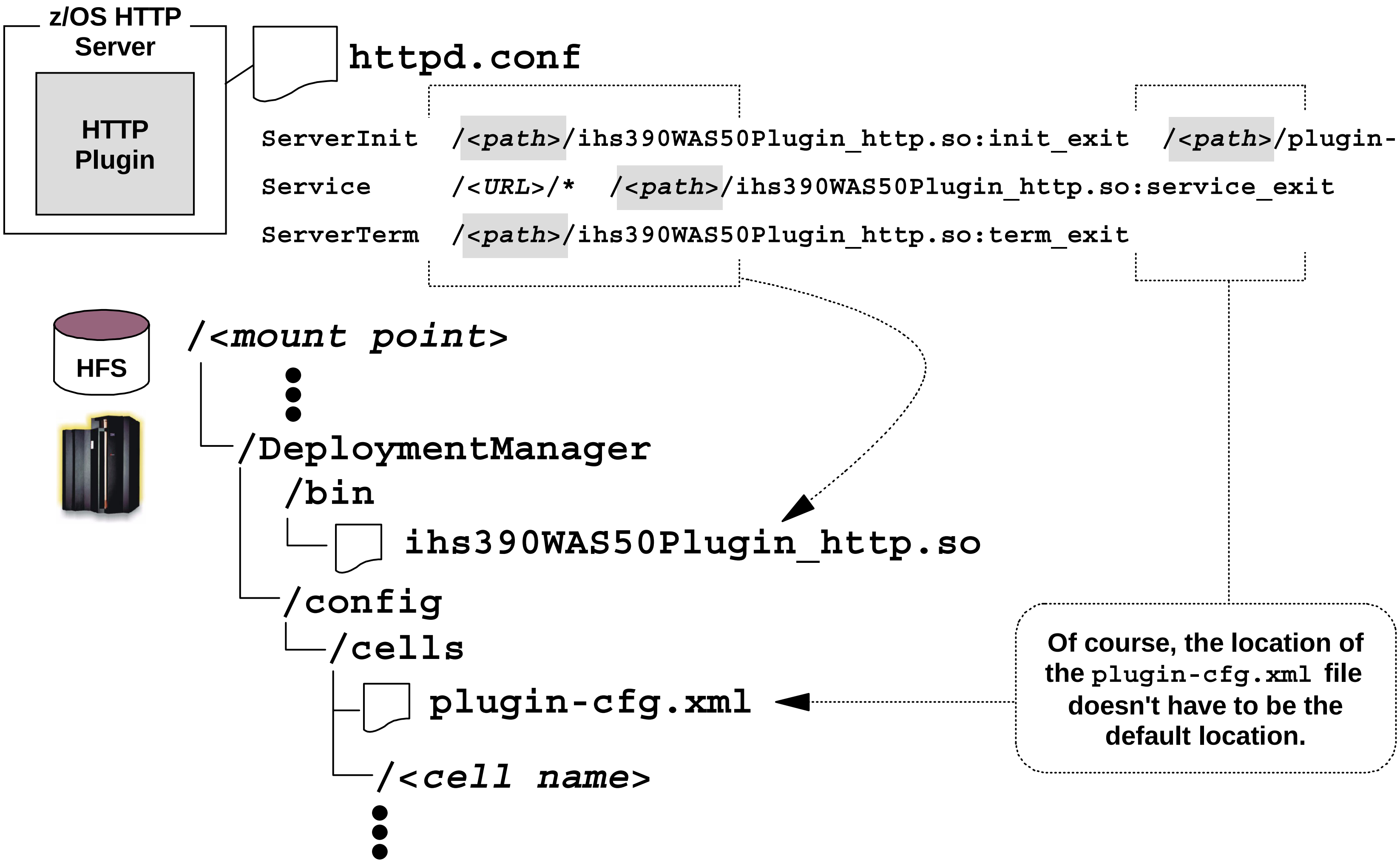
Much yet to be explained:

- Multiple server clusters
- How URL is routed to one or the other
- Other contents of this XML

First, let's look at how the Plugin is configured into the HTTP Server

How the HTTP Plugin is Configured

Very similar to how WebSphere for z/OS V3.5 was configured, and similar to how the V4 "Local Redirector Plugin" was configured:



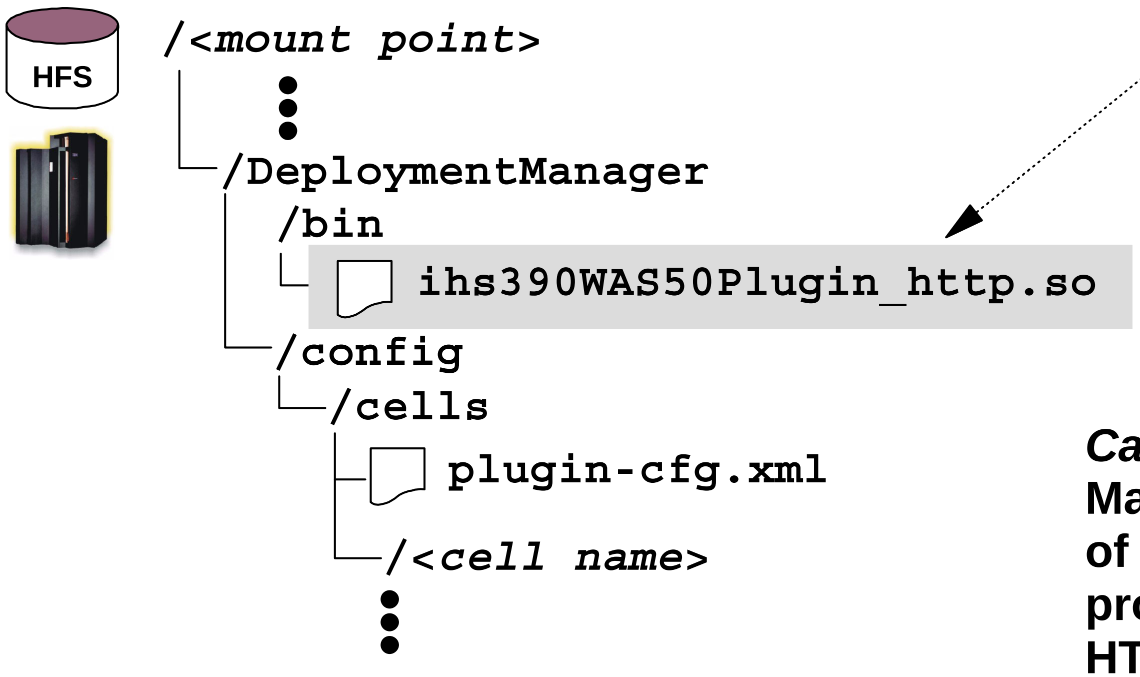
```
ServerInit   /<path>/ihs390WAS50Plugin_http.so:init_exit  /<path>/plugin-cfg.xml
Service      /<URL>/*  /<path>/ihs390WAS50Plugin_http.so:service_exit
ServerTerm   /<path>/ihs390WAS50Plugin_http.so:term_exit
```

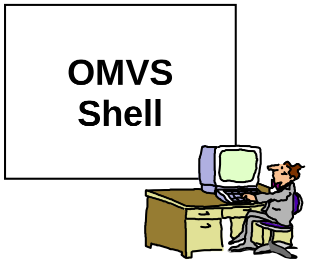
Of course, the location of the `plugin-cfg.xml` file doesn't have to be the default location.

Program Control ".so" File

Just like with WAS V3.5, the HTTP Plugin code has to be program controlled to load into HTTP Server and operate properly:

```
extattr +p ihs390WASPlugin_http.so
```

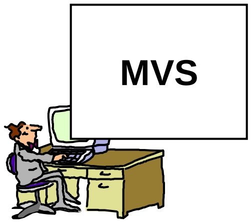




Caution: **each cell's Deployment Manager will have its own copy of this file. Make sure you're program controlling the one the HTTP Server is loading**

Starting the HTTP Server with this new Plugin is just like in the past ...

Starting the HTTP Server with Plugin



```
S <PROC>

  JESMSGLG JES2                  2 BBOWEB   S
  JESJCL   JES2                  3 BBOWEB   S
  JESYSMSG JES2                  4 BBOWEB   S
  SYSPRINT BBOWEB              101 BBOWEB   O
  SYSOUT   BBOWEB              105 BBOWEB   O
```

```
Licensed Material - Property of IBM
5655-I35 (C) Copyright IBM Corp. 2000, 2003

All Rights Reserved.

U.S. Government users - RESTRICTED RIGHTS - Use, Duplication, or
Disclosure restricted by GSA-ADP schedule contract with IBM Corp.
IBM is a registered trademark of the IBM Corp.

WebSphere HTTP Plug-in for z/OS and OS/390  Version 5.0 Service Level 0.0 is starting

WebSphere HTTP Plug-in for z/OS and OS/390
                        initializing with configuration file : /<path>/plugin-cfg.xml

WebSphere HTTP Plug-in for z/OS and OS/390  initialization went OK :-)
```

Quite a few things can go wrong ... we'll cover those later. Next let's look at the `plugin-cfg.xml` file, which is the configuration file used by the Plugin

Basic Layout of Configuration XML File

```
<Config>

 <Log LogLevel="Trace" Name="/etc/bboweb/http_plugin.log"/>

    <VirtualHostGroup Name="[VH_group_name]">
        <VirtualHost Name="[host]:[port]"/>
    </VirtualHostGroup>

    <ServerCluster Name="[name]">
        <Server CloneID="[Unique ID]"
            LoadBalanceWeight="2" Name="[node]_[server]">
            <Transport Hostname="[host]" Port="[port]" Protocol="http"/>
        </Server>
        <Server CloneID="[Unique ID]"
            LoadBalanceWeight="2" Name="[node]_[server]">
            <Transport Hostname="[host]" Port="[port]" Protocol="http"/>
        </Server>
        <PrimaryServers>
            <Server Name="[node]_[server]"/>
            <Server Name="[node]_[server]"/>
        </PrimaryServers>
    </ServerCluster>

    <UriGroup Name="[URI_group_name]">
        <Uri AffinityCookie="JSESSIONID"
            AffinityURLIdentifier="jsessionid" Name="/[context root]/*"/>
    </UriGroup>

    <Route ServerCluster="<ServerCluster name>"
        UriGroup="[VH group name]"
        VirtualHostGroup="VH group name"/>

</Config>
```

Location of logging file for plugin

Virtual Host Group (optional)

Information on a cluster and the server members in that cluster

One block of XML for each server cluster. *A single server is considered a cluster.*

URIs expected (optional)

Where to route URL ... this is the key to XML file

Let's see how this works ...

Multiple ServerClusters in XML

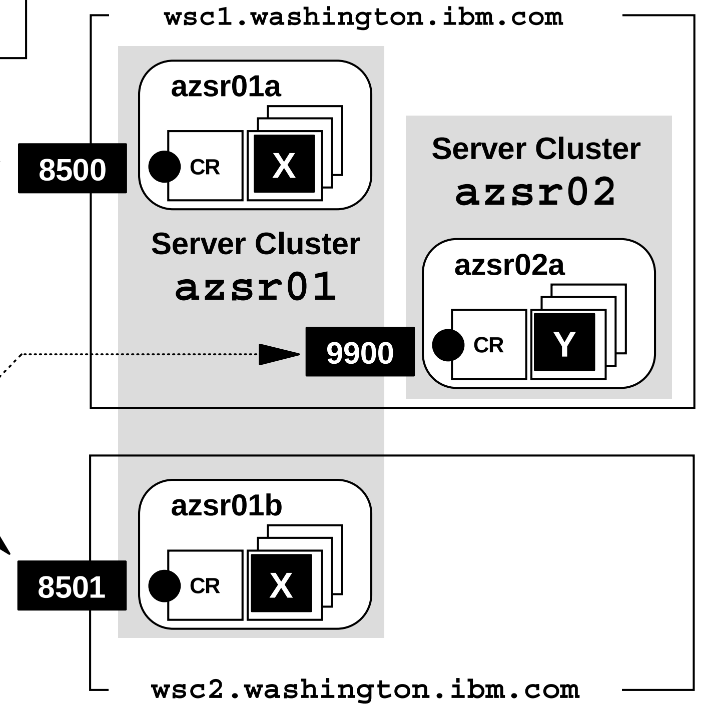
plugin-cfg.xml

```
<ServerCluster Name="azsr01">
    <Server CloneID="B9F0..."
        <Transport
          Hostname="wsc1.washington.ibm.com"
          Port="8500"/>
    </Server>
    <Server CloneID="A7FC..."
        <Transport
          Hostname="wsc2.washington.ibm.com"
          Port="8501"/>
    </Server>
</ServerCluster>

<ServerCluster Name="azsr02">
    <Server CloneID="C3FF..."
        <Transport
          Hostname="wsc1.washington.ibm.com"
          Port="9900"/>
    </Server>
</ServerCluster>
```

Note

This is not *exactly* how XML looks ... simplied here to save space on page

This illustrates how a single server is still considered part of a "ServerCluster"



Applications:

azsr01 cluster: "X" **Context root: /X**

azsr02 cluster: "Y" **Context root: /Y**

Next let's see how the Plugin knows to route a URL to one ServerCluster versus another ...

URIGroups and the <Route> Block

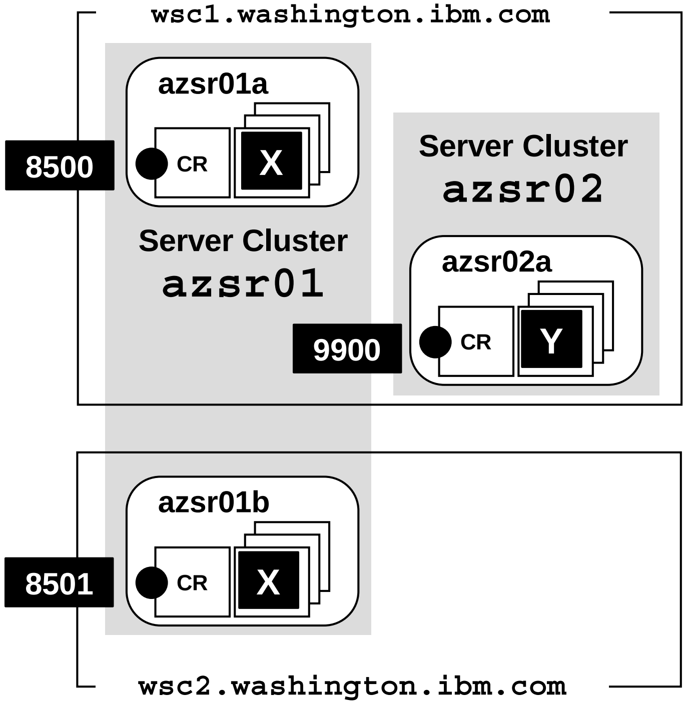
```
<ServerCluster Name="azsr01">
    <Server CloneID="B9F0..."
        <Transport
          Hostname="wsc1.washington.ibm.com"
          Port="8500"/>
    </Server>
    <Server CloneID="A7FC..."
        <Transport
          Hostname="wsc2.washington.ibm.com"
          Port="8501"/>
    </Server>
</ServerCluster>

<ServerCluster Name="azsr02">
    <Server CloneID="C3FF..."
        <Transport
          Hostname="wsc1.washington.ibm.com"
          Port="9900"/>
    </Server>
</ServerCluster>

<UriGroup Name="ApplX">
    <Uri Name="/X/*"/>
</UriGroup>
<UriGroup Name="ApplY">
    <Uri Name="/Y/*"/>
</UriGroup>

<Route ServerCluster="azsr01"
    UriGroup="ApplX"/>
<Route ServerCluster="azsr02"
    UriGroup="ApplY"/>
```

1

2

3

???
Which Server?



http://www.plugin.com/X/index.html

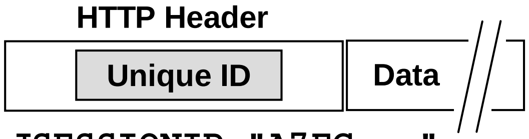
Assume this host is where HTTP Server with Plugin is running

Context Root

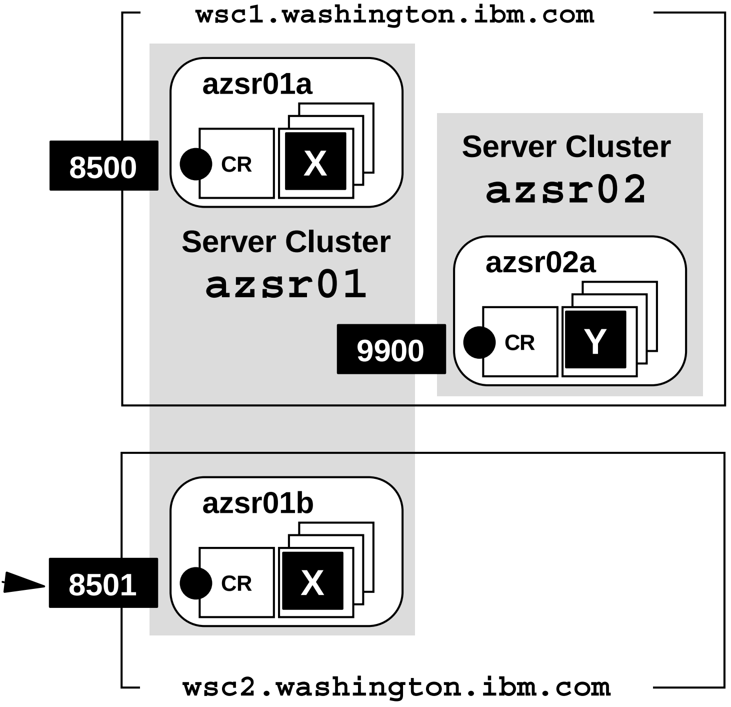
Next: routing between servers in a Cluster ...

Affinity or Round-Robin Routing

We saw how `<UriGroup>` and `<Route>` worked together to get URL to Server*Cluster*. Server it goes to depends on if HTTP Header has "AffinityCookie"



```
JSESSIONID="A7FC..."

<ServerCluster Name="azsr01">
    <Server CloneID="B9F0..."
        <Transport
          LoadBalanceWeight="2"
          Hostname="wsc1.washington.ibm.com"
          Port="8500"/>
    </Server>
    <Server CloneID="A7FC..."
        <Transport
          LoadBalanceWeight="2"
          Hostname="wsc2.washington.ibm.com"
          Port="8501"/>
    </Server>
</ServerCluster>

<UriGroup Name="ApplX">
    <Uri AffinityCookie="JSESSIONID"
     Name="/X/*"/>

<Route ServerCluster="azsr01"
    UriGroup="ApplX"/>
```

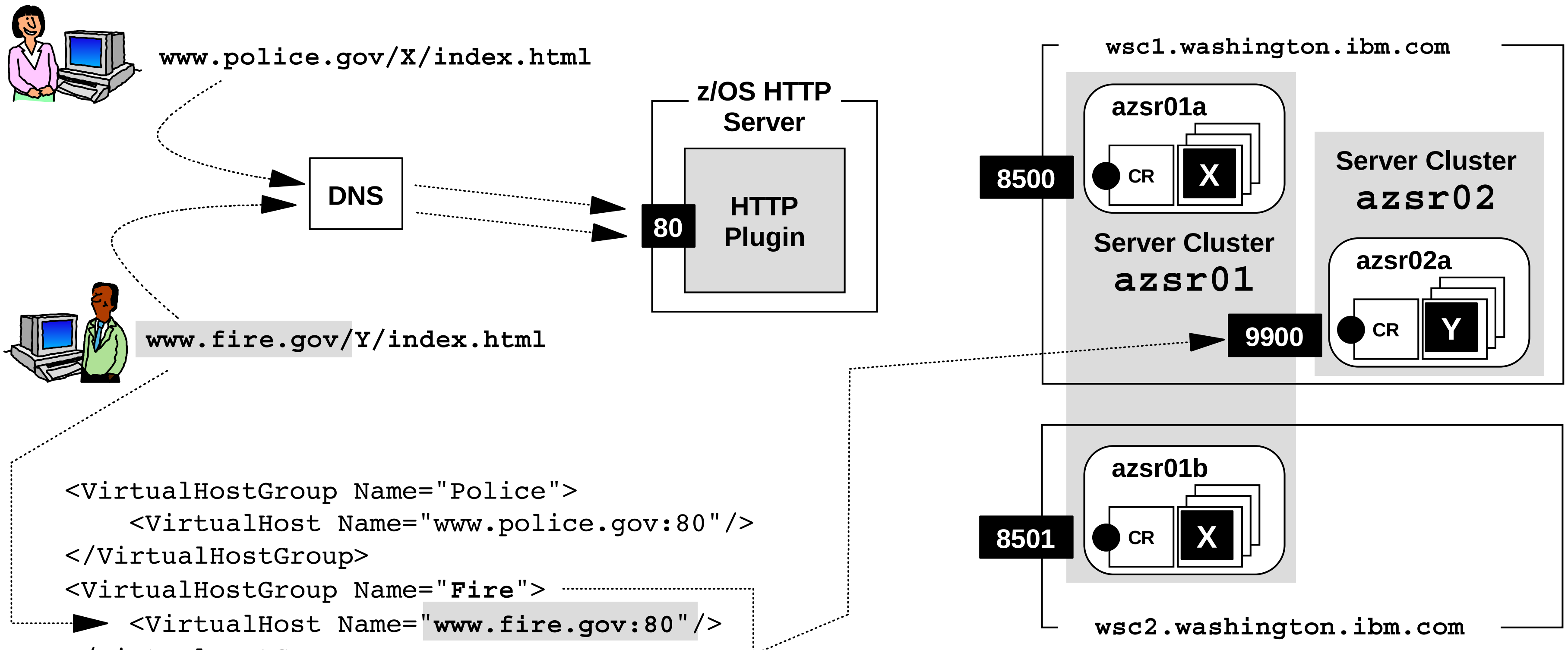


WebSphere "Session Manager" places JESSIONID cookie into HTTP Header when (and if) session object created

If no JSESSIONID in HTTP Header, then round-robin based on "LoadBalanceWeight"

Virtual Hosts and Routing to Cluster

In addition to routing based on Context Root, you may also route based on host value found on URL:



```
<VirtualHostGroup Name="Police">
    <VirtualHost Name="www.police.gov:80"/>
</VirtualHostGroup>
<VirtualHostGroup Name="Fire">
    <VirtualHost Name="www.fire.gov:80"/>
</VirtualHostGroup>

<ServerCluster Name="azsr01">
<ServerCluster Name="azsr02">

<Route ServerCluster="azsr01"
    VirtualHostGroup="Police"/>
<Route ServerCluster="azsr02"
    VirtualHostGroup="Fire"/>
```

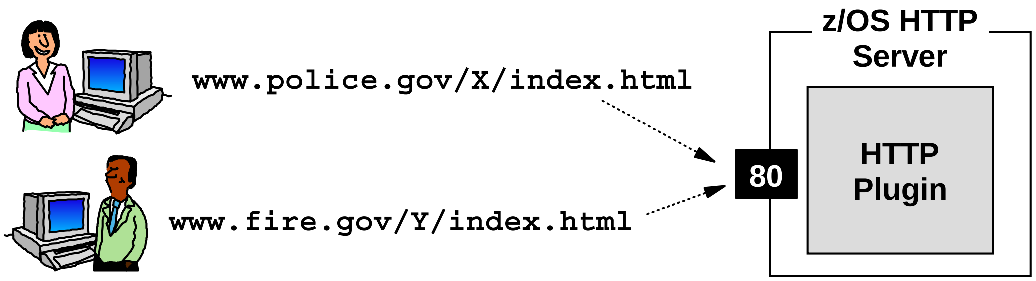
Processing within ServerCluster just as before

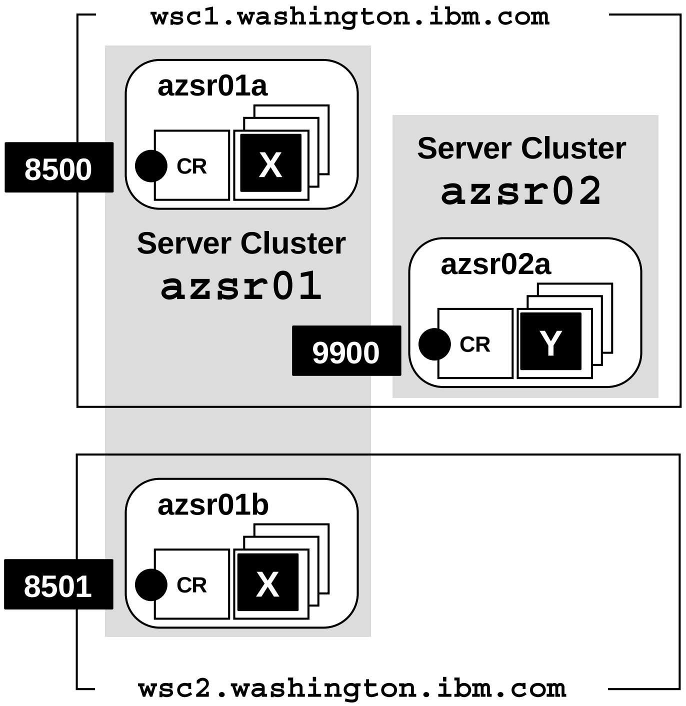
- (Including affinity processing and round-robin)

Backend server host names/ports need not be the same as what comes from clients

- XML in ServerCluster block points to actual backend host names and ports to be used
- You can "hide" actual backend host name values from public

Combination: URI *and* Virtual Host





```
<VirtualHostGroup Name="Police">
    <VirtualHost Name="www.police.gov:80"/>
</VirtualHostGroup>
<VirtualHostGroup Name="Fire">
    <VirtualHost Name="www.fire.gov:80"/>
</VirtualHostGroup>

<ServerCluster Name="azsr01">

<ServerCluster Name="azsr02">

<UriGroup Name="ApplX">
    <Uri Name="/X/*"/>
</UriGroup>
<UriGroup Name="ApplY">
    <Uri Name="/Y/*"/>
</UriGroup>

<Route ServerCluster="azsr01"
    UriGroup="ApplX"
    VirtualHostGroup="Police"/>
<Route ServerCluster="azsr02"
    UriGroup="ApplY"
    VirtualHostGroup="Fire"/>
```

Only requests with *both* `www.police.gov` and context root of `/X` will get routed to `azsr01` cluster

- If *only* `UriGroup` on `Route`, then *all* requests with that context root get routed there, regardless of host name on URL
- If *only* `VirtualHostGroup` on `Route`, then *all* requests for that host get routed there, regardless of context root value

One more variation on this, then we'll get to troubleshooting

Multiple Context Roots per UriGroup

```
<VirtualHostGroup Name="Police">
    <VirtualHost Name="www.police.gov:80"/>
</VirtualHostGroup>
<VirtualHostGroup Name="Fire">
    <VirtualHost Name="www.fire.gov:80"/>
</VirtualHostGroup>

<ServerCluster Name="azsr01">

<ServerCluster Name="azsr02">

<UriGroup Name="ApplX">
    <Uri Name="/X/*"/>
    <Uri Name="/A/*"/>
    <Uri Name="/B/*"/>
    <Uri Name="/C/*"/>
      :
</UriGroup>

<UriGroup Name="ApplY">
    <Uri Name="/Y/*"/>
</UriGroup>

<Route ServerCluster="azsr01"
    UriGroup="ApplX"
    VirtualHostGroup="Police"/>
<Route ServerCluster="azsr02"
    UriGroup="ApplY"
    VirtualHostGroup="Fire"/>
```

You may code multiple URIs in the `<UriGroup>` block of XML

- **Many different context roots will get routed to ServerCluster `azsr01`**

In this example `<Route>` has `VirtualHostGroup` as well.

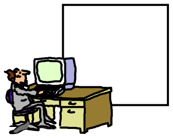
- **All those URLs must have host of `www.police.gov`**

Yes, multiple `VirtualHost` names permited per `VirtualHostGroup`

Lots of permutations to this

There's an opportunity to introduce ambiguity into the XML. You should be careful to avoid this ...

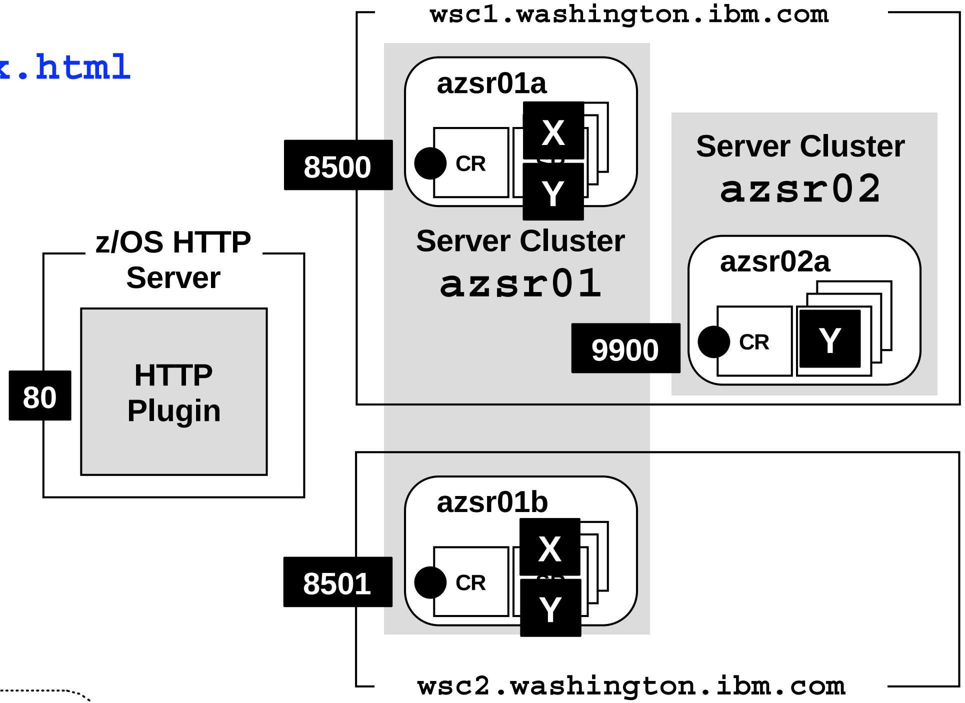
Avoid Ambiguity



http://www.plugin.com/Y/index.html

```
<ServerCluster Name="azsr01">

<ServerCluster Name="azsr02">

<UriGroup Name="ApplX">
    <Uri Name="/X/*"/>
    <Uri Name="/Y/*"/>   ✓
</UriGroup>

<UriGroup Name="ApplY">
    <Uri Name="/Y/*"/>   ✓
</UriGroup>

<Route ServerCluster="azsr01"
    UriGroup="ApplX"/>
<Route ServerCluster="azsr02"
    UriGroup="ApplY"/>
```

???
Which ServerCluster?



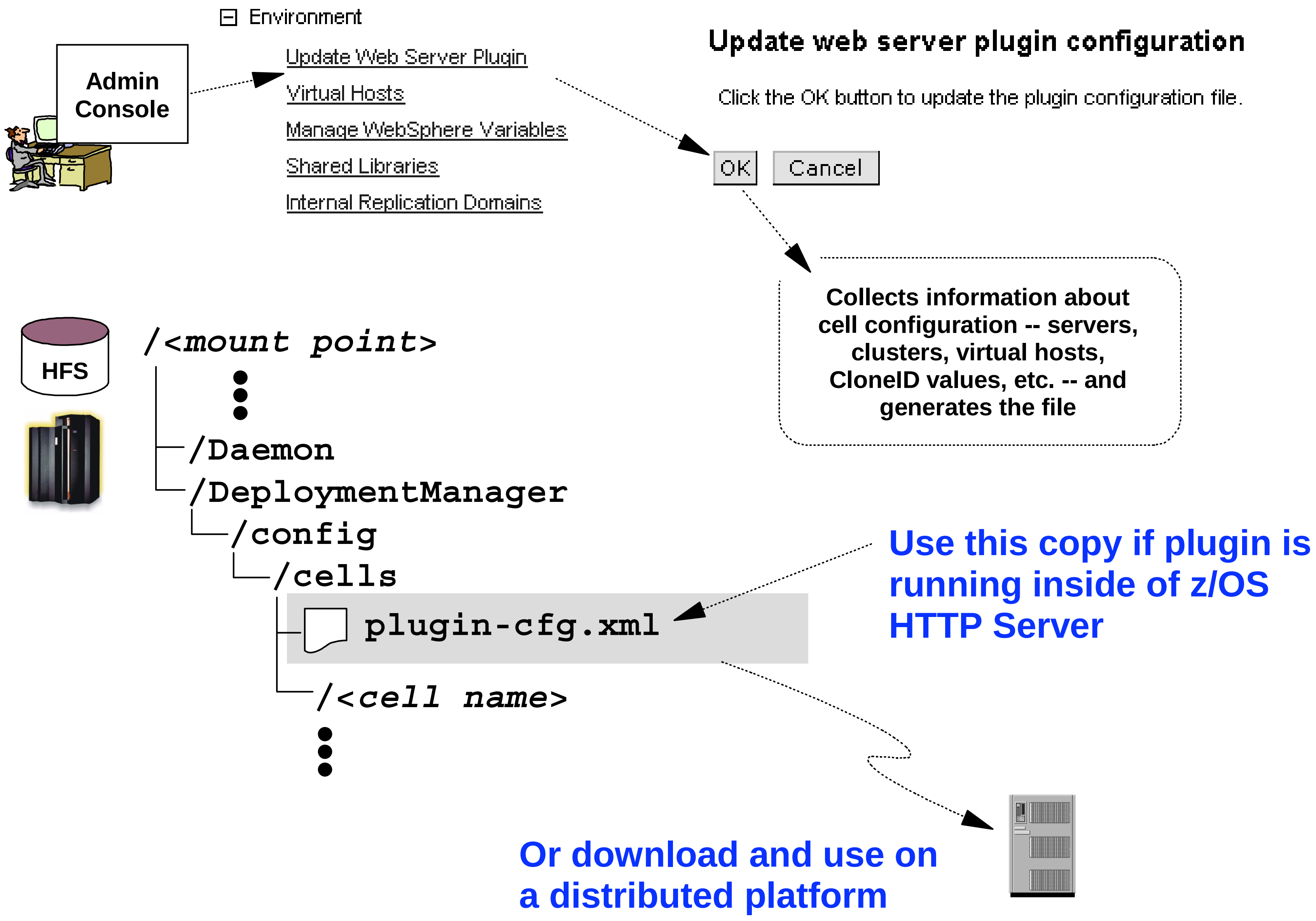
It appears the last `<Route>` statement in XML that matches is the one that applies ... but my testing wasn't that exhaustive. Other rules may apply.

Avoid ambiguity like this. Use VirtualHosts to resolve to a single ServerCluster.

Or don't code second ServerCluster's `<Uri Name="/Y/*">` in XML

Next: Where does `plugin-cfg.xml` file come from initially?

Generating plugin-cfg.xml



Collects information about cell configuration -- servers, clusters, virtual hosts, CloneID values, etc. -- and generates the file

Use this copy if plugin is running inside of z/OS HTTP Server

Or download and use on a distributed platform

Few Notes About Generated XML

Generated XML a great starting point, but probably not exactly what you need ...

```
<?xml version="1.0" encoding="Cp1047"?>
<Config>
    <Log LogLevel="Trace" Name="/wasv5config/azcell/DeploymentManager/logs"/>
    <VirtualHostGroup Name="default_host">
        <VirtualHost Name="wsc1.washington.ibm.com:9518"/>
        <VirtualHost Name="wsc1.washington.ibm.com:9519"/>
        <VirtualHost Name="wsc2.washington.ibm.com:9548"/>
        <VirtualHost Name="wsc2.washington.ibm.com:9558"/>
    </VirtualHostGroup>
    <ServerCluster Name="azsr01Cluster">
        <Server CloneID="B9F91E06DC4511C100000C0C0000000109521845"
            BalanceWeight="2" Name="aznodea_azsr01a">
            nsport Hostname="wsc1.washington.ibm.com" Port="9548" Protocol="http"/>
            >
            CloneID="B9F95C1EDD90F28500000BF4000000040952l8
            BalanceWeight="2" Name="aznodeb_azsr01b">
            <Transport Hostname="wsc2.washington.ibm.com" Port=
        </Server>
    </ServerCluster>
    <ServerCluster Name="dmgr_azdmnode_Cluster">
        <Server CloneID="B9F9295C449786C4000001380000001E09521845" Name="azdmnode_dmgr">
            <Transport Hostname="wsc1.washington.ibm.com" Port="9518" Protocol="http"/>
        </Server>
    </ServerCluster>
    <UriGroup Name="default_host_azsr01Cluster_URIs">
        <Uri AffinityCookie="JSESSIONID"
            AffinityURLIdentifier="jsessionid" Name="/mem/*"/>
        <Uri AffinityCookie="JSESSIONID"
            AffinityURLIdentifier="jsessionid" Name="/MyIVT/*"/>
    </UriGroup>
    <Route ServerCluster="azsr01Cluster"
        UriGroup="default_host_azsr01Cluster_URIs" VirtualHostGroup="default_host"/>
</Config>
```

May not look *exactly* like actual generated file

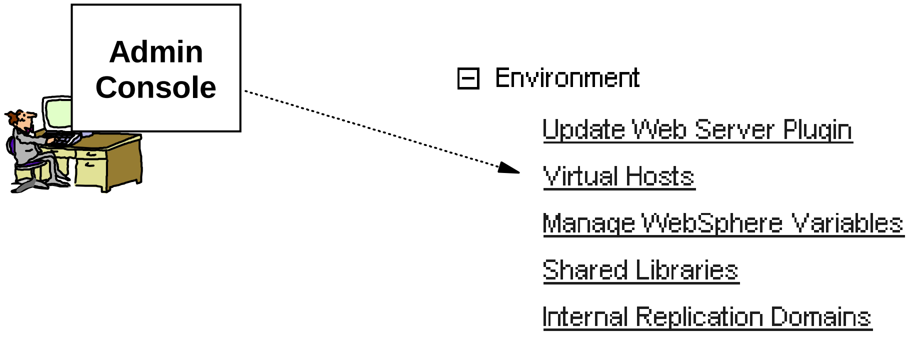
Virtual Host for the Plugin itself won't appear here. You'll probably need to hand-code another.

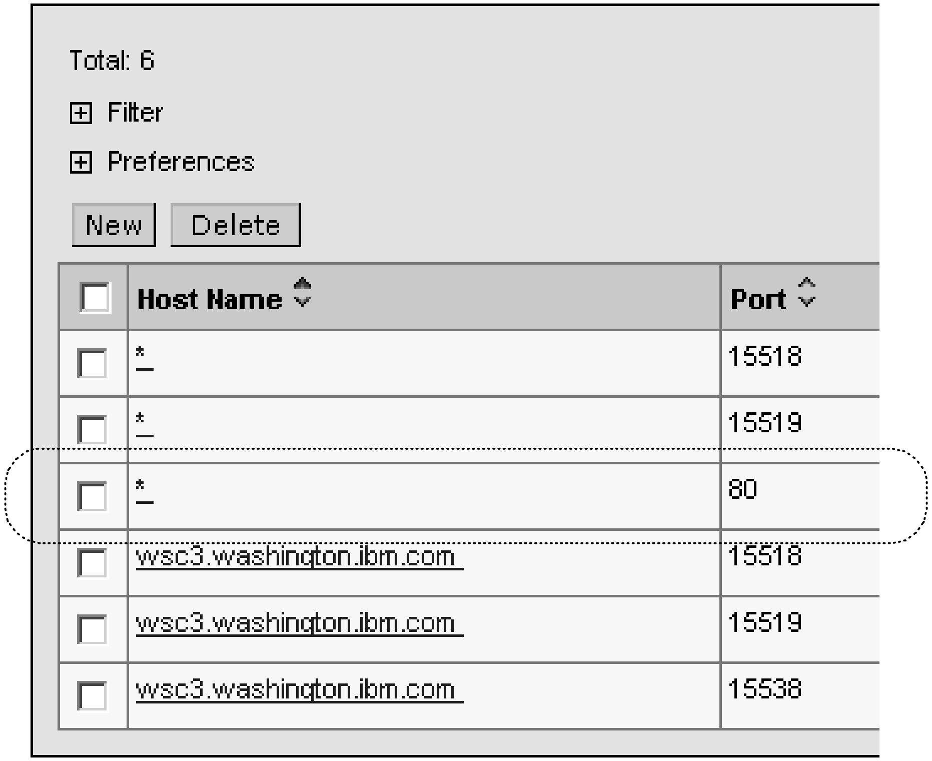
Creates `ServerCluster` for Deployment Manager. Unnecessary unless you're coming through Plugin to get to Admin Console

Routes generated use *both* `UriGroup` and `VirtualHostGroup`

One Update Needed in Runtime

Must create a "Virtual Host Alias" with port 80:





Total: 6

⊞ Filter

⊞ Preferences

☐	Host Name	Port
☐	*	15518
☐	*	15519
☐	*	80
☐	wsc3.washington.ibm.com	15518
☐	wsc3.washington.ibm.com	15519
☐	wsc3.washington.ibm.com	15538

The "Host Name" must match what client used to access HTTP server

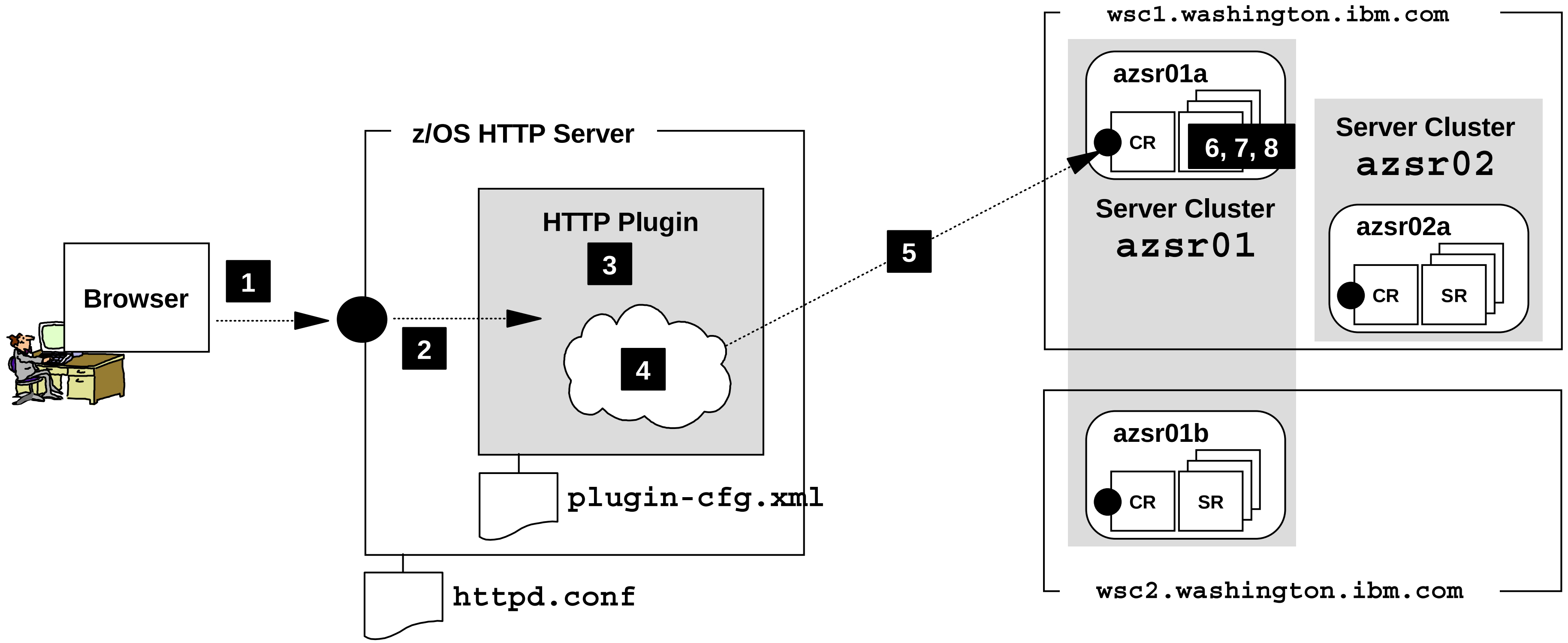
- Easiest: asterisk wild card ... permits any value

Interestingly, the port must be 80, regardless of the port on which the HTTP server is listening

To run application, that application must be bound to Virtual Host in which this alias is defined.

On to troubleshooting ...

Troubleshooting Overview



1 URL gets to HTTP Server

2 URL passed into Plugin

3 Plugin initialized

4 URL matches XML processing

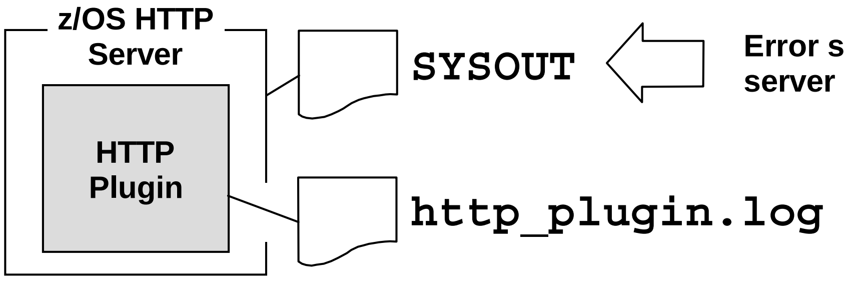
5 Plugin sees Server as up

6 Plugin's "VH Alias" present

7 Server recognizes context root

8 Application is running

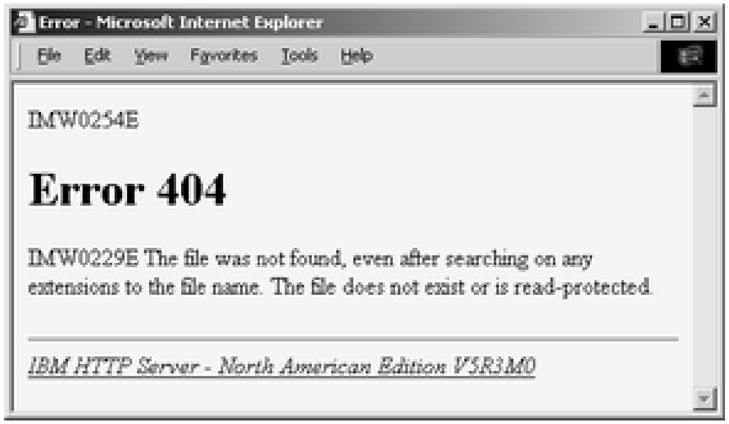
Request Must Match "Service"



Error seen here is simply URL failing to match on any statement and HTTP server throwing 404 message

Request never gets "over the wall" into Plugin, so you'll see nothing in here about request

```
                   http://www.plugin.com/ABC/index.html
httpd.conf

  Service   /ApplX/*   /<path>/ihs390WAS50Plugin_http.so:service_exit
  Service   /ApplY/*   /<path>/ihs390WAS50Plugin_http.so:service_exit
    :
  Pass      /*         /<document root>
```


Error - Microsoft Internet Explorer
File Edit View Favorites Tools Help

IMW0254E

Error 404

IMW0229E The file was not found, even after searching on any extensions to the file name. The file does not exist or is read-protected.

IBM HTTP Server - North American Edition V5R3M0

Lack of a proper "Service" statement for the URL typically results in 404 message

Remember to update `httpd.conf` when new applications deployed in backend runtime

Did Plugin Initialize?

```
JESMSGLG JES2                 2 BBOWEB    S
JESJCL   JES2                 3 BBOWEB    S
JESYSMSG JES2                 4 BBOWEB    S
SYSPRINT BBOWEB             101 BBOWEB    O
SYSOUT   BBOWEB             105 BBOWEB    O
```

Look for the "Smiley Face"

```
WebSphere HTTP Plug-in for z/OS and OS/390 initialization went OK :-)
```

If no "Smiley Face," then look for the "Frowny Face:"

```
WebSphere HTTP Plug-in for z/OS and OS/390 initialization FAILED (rc = 4) :-(
```

It's possible that the Plugin failed to initialize even though no " :-(" is present:

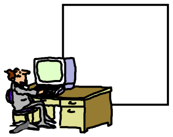
Hint: when no " :-(" found, search on 'Failed to'

For example, if pointer to Plugin's module is incorrect

```
Failed to load DLL module /<Config Root>/DeploymentManager/bin/ihs390WAS50Plugin_http.so
EDC5205S DLL module not found. (errnojr=0534011c)
```

Let's take a look at all the things that must be present in `httpd.conf` configuration to permit Plugin to intialize ...

Browser Symptom: No Plugin

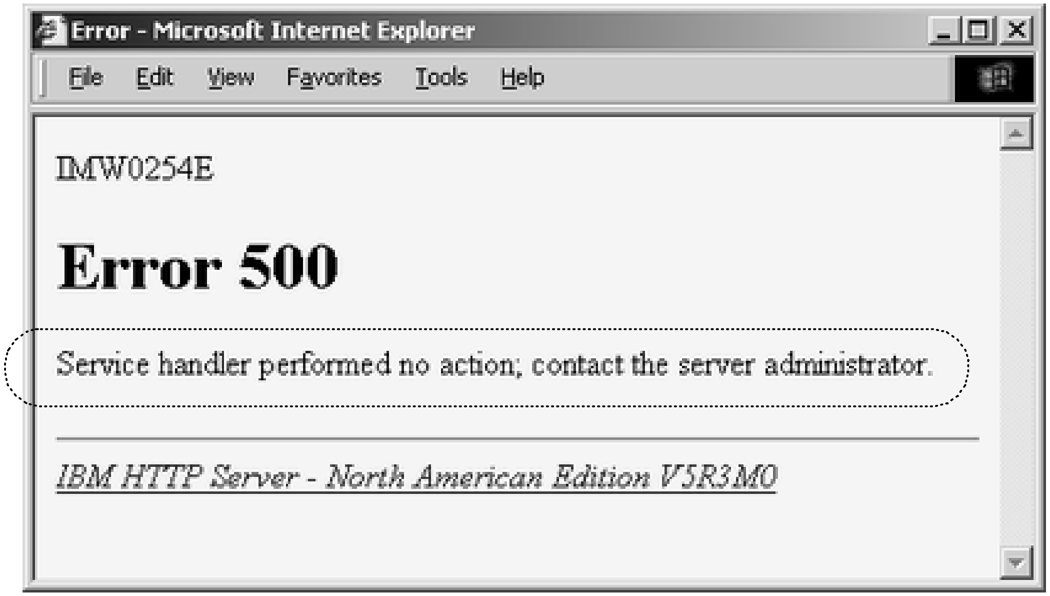


```
http://www.plugin.com/ApplX/index.html
```

httpd.conf

```
ServerInit   /<path>/ihs390WAS50Plugin_http.so:init_exit   /<path>/plugin-cfg.xml
Service      /ApplX/*  /<path>/ihs390WAS50Plugin_http.so:service_exit
ServerTerm   /<path>/ihs390WAS50Plugin_http.so:term_exit
```

But the Plugin isn't initialized, so the `service_exit` isn't available



Error - Microsoft Internet Explorer
File Edit View Favorites Tools Help

IMW0254E

Error 500

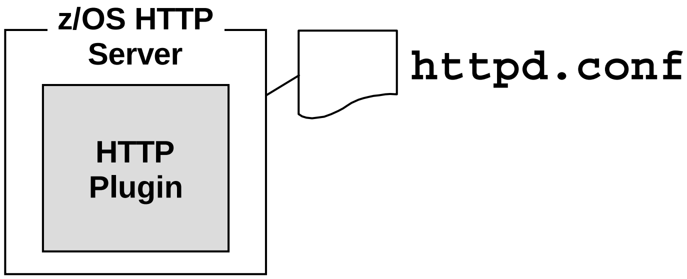
Service handler performed no action; contact the server administrator.

IBM HTTP Server - North American Edition V5R3M0

Caution! "Service Handler Performed No Action" may result even when Plugin is initialized.

Always check for Plugin initialization as first thing

Primary Causes of Initialization Failures



If Plugin doesn't initialize, look at four primary causes first. All have to do with the `ServerInit` statement in the `httpd.conf` file:

```
                             1                           2          3         4
ServerInit  /<path>/ihs390WAS50Plugin_http.so:init_exit  /<path>/plugin-cfg.xml
Service     /<URL>/*  /<path>/ihs390WAS50Plugin_http.so:service_exit
ServerTerm  /<path>/ihs390WAS50Plugin_http.so:term_exit
```

1 Plugin ".so" module not found or can't be loaded

- Check directory path to module
- Check case (it matters on directory paths)
- Check file name, including case
- Check permissions on directories and file itself (HTTP Server ID needs "read" minimum)

2 The "exit" specified on module incorrect

- Must be `init_exit`, not `service_exit` or `term_exit` like the other statements

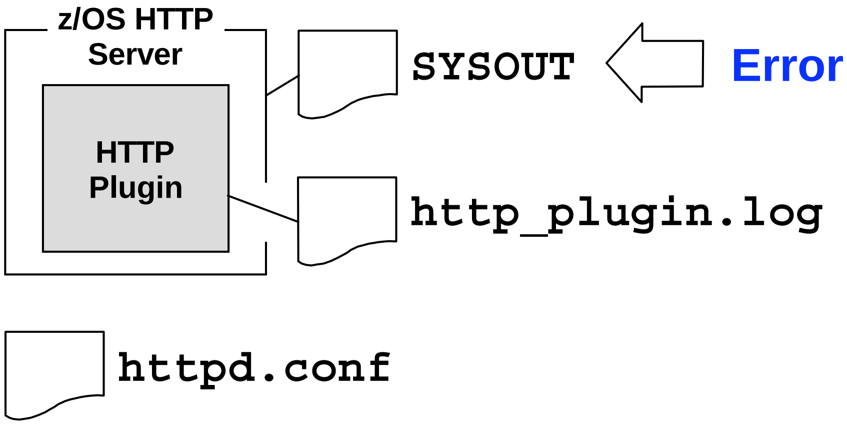
3 Plugin's XML file not found or can't be loaded

- Plugin's XML file specified as parameter on end of ServerInit statement
- Check directory path to file, file name, case and permissions

4 Bad contents of Plugin's XML file

- See Plugin's log file for pointer to line of XML file that's in error

Plugin Module Not Found or Loaded


z/OS HTTP Server

HTTP Plugin

SYSOUT

Error symptom seen here

http_plugin.log

httpd.conf

```
ServerInit  /<path>/ihs390WAS50Plugin_http.so:init_exit  /<path>/plugin-cfg.xml
```

Anything that prevents the HTTP Server from locating and loading module will cause this problem

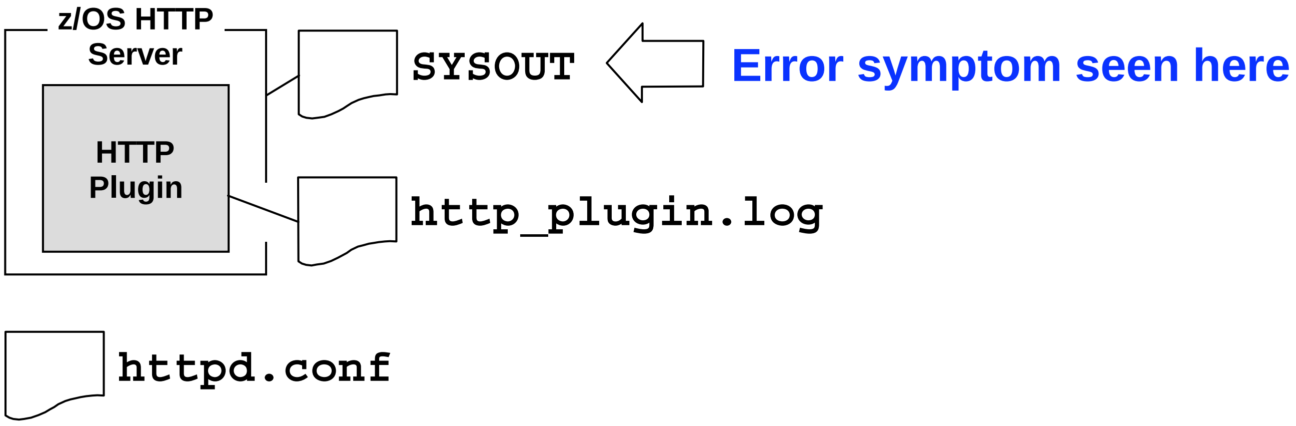
- Incorrect directory; incorrect case
- Wrong module name; incorrect case
- Restrictive permissions

No "smiley," no "frowny" ... just the following:

```
 :
Failed to load DLL module /<path>/DeploymentManager/bin/ihs390WAS50Plugin_http.so
EDC5205S DLL module not found. (errnojr=0534011c)
 :
```

HTTP Server must be able to locate module before it can load it

Wrong "Exit" on ServerInit



```
ServerInit  /<path>/ihs390WAS50Plugin_http.so:wrong_exit  /<path>/plugin-cfg.xml
```

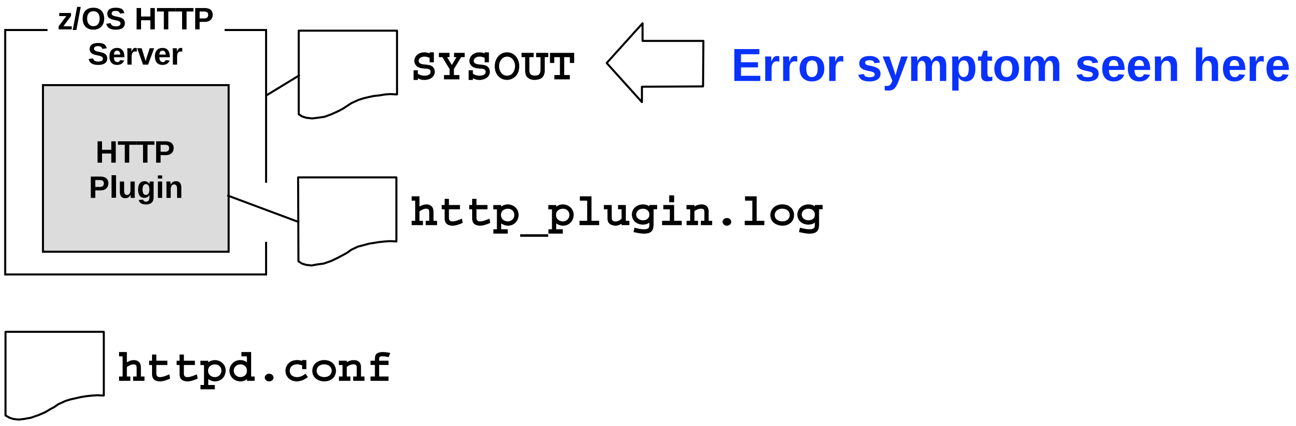
Common error: copying `Service` statement line to form `ServerInit` and then forgetting to change exit

Anything other than **`:init_exit`** is incorrect exit.

```
API... Successful loading shared library "/<path>/ihs390WAS50Plugin_http.so"
API... Trying to get fn pointer "wrong_exit" from module "/<path>/ihs390WAS50Plugin_http.so"
Failed to load function wrong_exit: EDC5214I Requested function not found in this DLL.
IMW0437E Return code 123 loading function wrong_exit from DLL module /<Plugin module>
IMW0438E Serverinit Error: server did not load functions from DLL module /<Plugin module>
```

**Plugin module has three exits: `init_exit`, `server_exit` and `term_exit`.
Only `init_exit` used to load module.**

plugin-cfg.xml Not Found or Specified



Error symptom seen here

```
ServerInit  /<path>/ihs390WAS50Plugin_http.so:wrong_exit  /<path>/plugin-cfg.xml
```

- Incorrect directory; incorrect case
- Parameter simply missing
- Wrong file name; incorrect case
- Restrictive permissions

Anything that prevents the Plugin from locating and reading XMLwill cause this problem

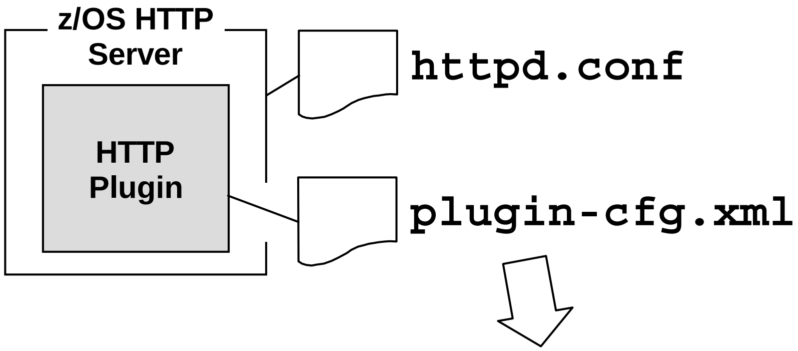
```
WebSphere HTTP Plug-in for z/OS and OS/390 initializing with configuration file : /<path>/<file>
ws_common: websphereUpdateConfig: Failed parsing the plugin config file
WebSphere HTTP Plug-in for z/OS and OS/390  initialization FAILED (rc = 3) :-(
IMW0438E Serverinit Error: server did not load functions from DLL module /<Plugin
```

Find on :-(

Note "rc=3" ... bad XML contents is a "rc=4". "rc=3" means file itself can't be found, rather than what's inside file is bad.

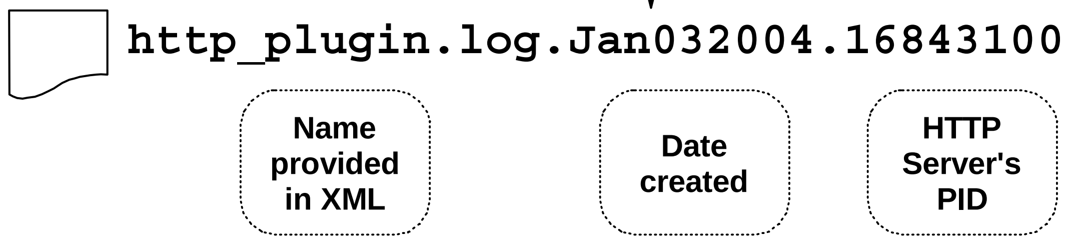
Plugin module must be able to find XML file. There's no "default" that's taken.

Where Plugin Logging Goes



The Plugin's log file is important for debugging problems related to the Plugin's processing

```
<?xml version="1.0"?>
<Config>
    <Log LogLevel="[level]" Name="/[path]/http_plugin.log"/>
      :
```



`Error`	Error messages
`Warn`	Warning + Error messages
`Trace`	Trace + Warning + Error messages

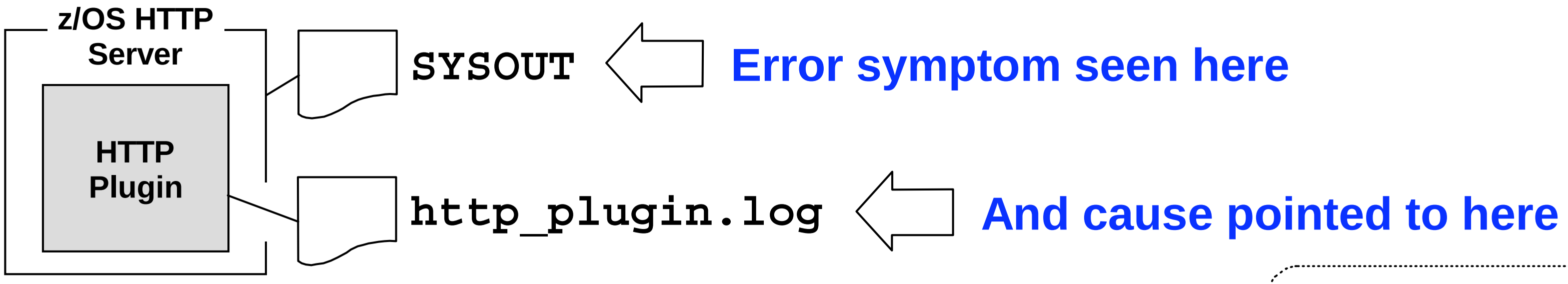
File is in EBCDIC and is quite readable.

Beware of "Trace" -- lots of output.

Default location:

`/<config root>/DeploymentManager/logs`

Bad Contents of plugin-cfg.xml



Error symptom seen here

And cause pointed to here

"rc=4" implies XML found, but contained bad data

SYSOUT

```
ws_common: websphereUpdateConfig: Failed parsing the plugin config file
WebSphere HTTP Plug-in for z/OS and OS/390  initialization FAILED (rc = 4) :-(
```

plugin-cfg.xml

```
****** **************************** Top of Data ***********************
000001 <?xml version="1.0" encoding="Cp1047"?>
000002 <Config>
000003     <Log LogLevel="Error" Name="/etc/bboweb/http_plugin.log"/>
000004     <ServerCluster Name="azsr01Cluster"
000005         <Server CloneID="B9F91E06DC4511C100000C                45"
000006             LoadBalanceWeight="2" Name="aznodea
```

Missing > at end of line 4

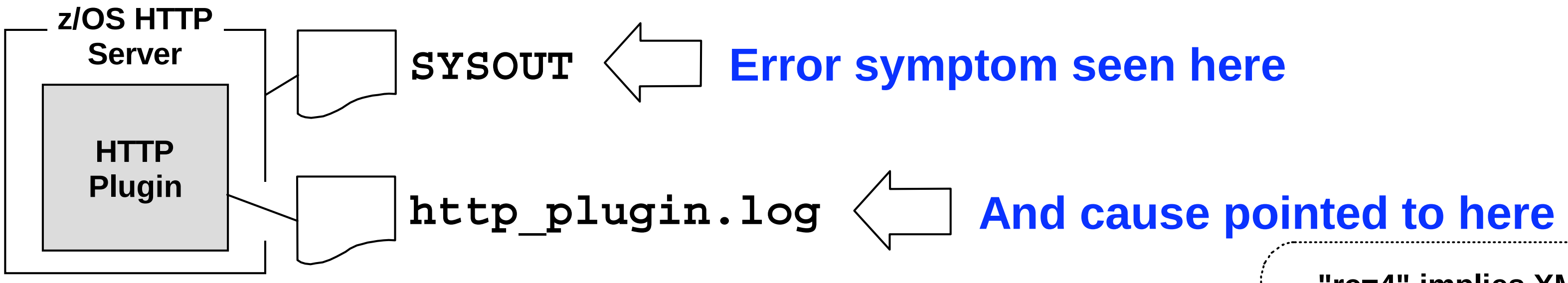
http_plugin.log.<*date*>.<*pid*>

```
ERROR: ... Expected '=' token; got 'Server'. line 5 of /<path>/plugin-cfg.xml
```

"Error" tracing is all that's needed

Much better than V3.5 or V4 plugin, which offered no hint as to where in "was.conf" file problem could be found.

Can't Resolve Host Name in XML



Error symptom seen here

And cause pointed to here

"rc=4" implies XML found, but contained bad data

SYSOUT

```
ws_common: websphereUpdateConfig: Failed parsing the plugin config file
WebSphere HTTP Plug-in for z/OS and OS/390  initialization FAILED (rc = 4)  :-(
```

plugin-cfg.xml

```
000004     <ServerCluster Name="azsr01Cluster">
000005         <Server CloneID="B9F91E06DC4511C100000C0C0000
000006             LoadBalanceWeight="2" Name="aznodea_azsr0
000007             <Transport Hostname="www.not-there.com" P
```

Hostname specified can't be resolved

http_plugin.log.<*date*>.<*pid*>

```
ERROR: ws_transport: transportSetServerAddress: unable to resolve host name: <host name>
ERROR: lib_sxp: sxpParse: End element returned FALSE for Transport. line 7 of <plugin-cfg.xml>
```

"Error" tracing is all that's needed

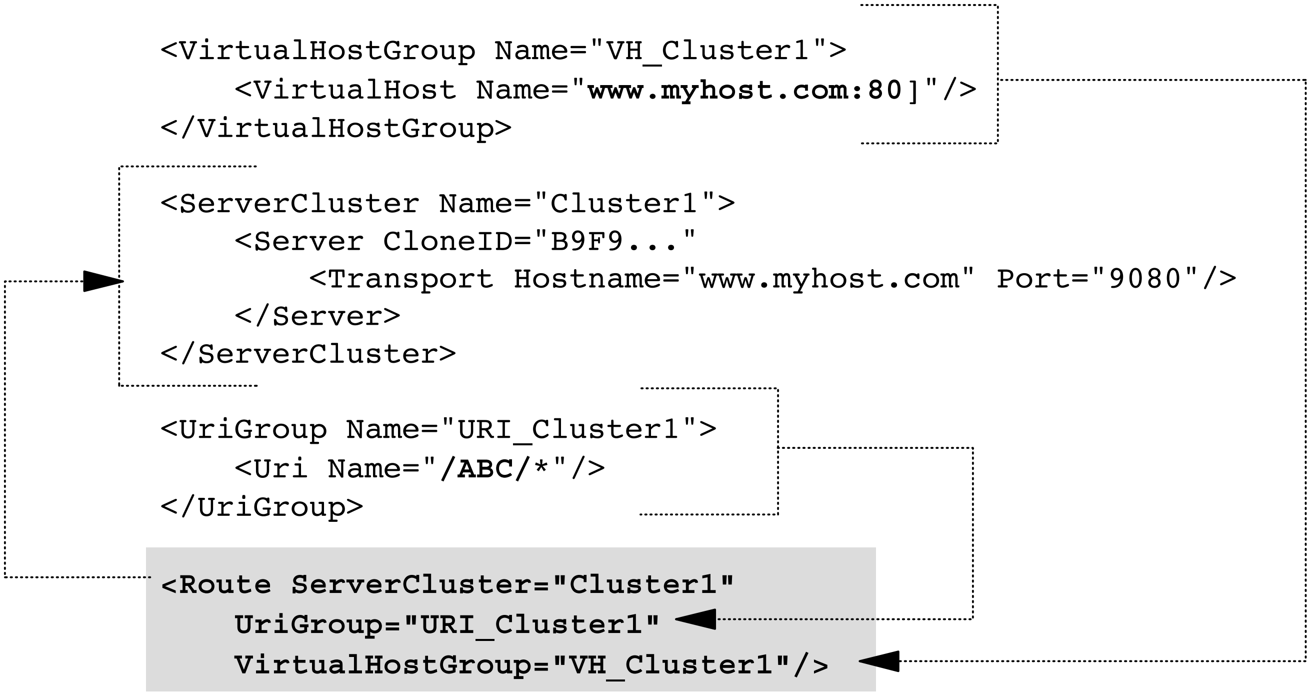
Watch out for typos in your host name values

(IP addresses aren't resolved; wrong addresses are treated like an IP stack that's not present; Plugin will simply balance to other server in cluster if possible)

Request Must Get Mapped to "Route"

It all depends on how you have your `<Route>` block coded. Rule is `<Route>` must have:

`UriGroup` reference *or* **`VirtualHostGroup` reference** *or* **Both**



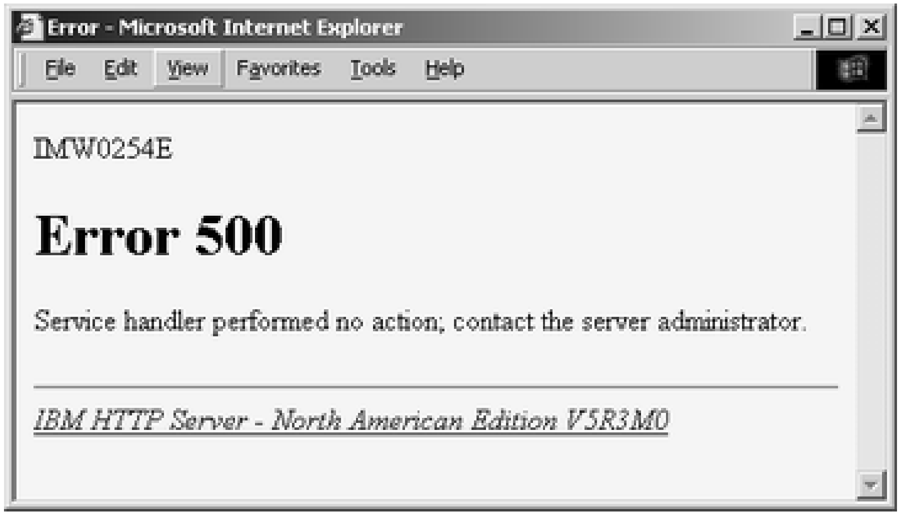
```
<VirtualHostGroup Name="VH_Cluster1">
    <VirtualHost Name="www.myhost.com:80]"/>
</VirtualHostGroup>

<ServerCluster Name="Cluster1">
    <Server CloneID="B9F9..."
        <Transport Hostname="www.myhost.com" Port="9080"/>
    </Server>
</ServerCluster>

<UriGroup Name="URI_Cluster1">
    <Uri Name="/ABC/*"/>
</UriGroup>

<Route ServerCluster="Cluster1"
    UriGroup="URI_Cluster1"
    VirtualHostGroup="VH_Cluster1"/>
```

This example has both `UriGroup` and `VirtualHostGroup` references on the `<Route>` statement

URL	*Match?*	*Why*
`www.yourhost.com/ABC/...`	No	No match on Virtual Host
`www.myhost.com/XYZ/...`	No	No match on URI
`www.myhost.com/ABC/...`	Yes	Matches both Virtual host *and* URI

Let's see the error symptoms ...

Route Not Mapped Symptom


Error - Microsoft Internet Explorer
File Edit View Favorites Tools Help

IMW0254E

Error 500

Service handler performed no action; contact the server administrator.

IBM HTTP Server - North American Edition V5R3M0

Regardless of type of failure:

- Failure to match URI
- Failure to match VirtualHost
- Failure to match combination of both

The browser error is the same:

- Error 500 -- "Service handler performed no action."

Caution: this might also indicate the Plugin isn't initialized.

The Plugin's log tells the story:

No match of URI

```
TRACE: ws_common: webspehereUriMatch: Failed to match: /Test/index.html
TRACE: ws_common: websphereFindServerGroup: No route found
TRACE: ws_common: websphereHandleRequest: Failed to find a server group
TRACE: ws_common: websphereEndRequest: Ending the request
```

No match of VHost

```
TRACE: ws_common: websphereVhostMatch: Failed to match: www.plugin.com:8070
TRACE: ws_common: websphereFindServerGroup: No route found
TRACE: ws_common: websphereHandleRequest: Failed to find a server group
TRACE: ws_common: websphereEndRequest: Ending the request
```

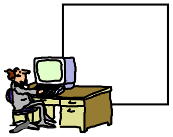
If *both* VH and URI are mismatched, VH error will appear in Trace

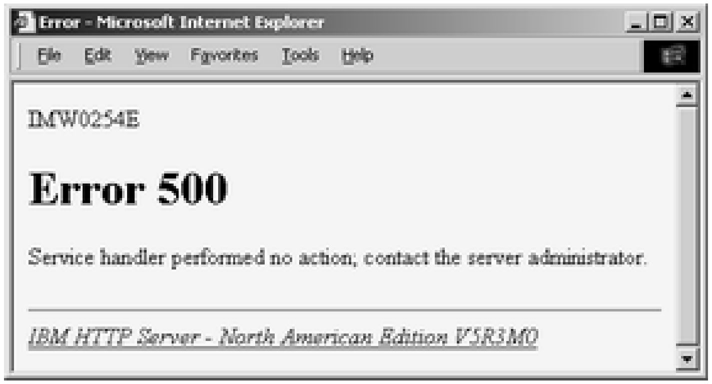
Need `Loglevel="Trace"` set for this information

Next up: when the backend server isn't available ...

Plugin Must See Server as "Up"

What happens when a URL maps to a `<Route>`, but all the servers in that Cluster are down? (For example, you simply forgot to start those servers)





Error - Microsoft Internet Explorer
File Edit View Favorites Tools Help

IMW0254E

Error 500

Service handler performed no action; contact the server administrator.

IBM HTTP Server - North American Edition V5R3M0

Browser error is the ubiquitous "Error 500"

```
TRACE: ws_common: websphereExecute: Executing the transaction with the app server
TRACE: ws_                          Stream: Getting the stream to the app server
TRACE: ws_                          StreamDequeue: Checking for existing stream from the queue
ERROR: ws_                          Stream: Failed to connect to app server, OS err=1128
ERROR: ws_                          ecute: Failed to create the stream
ERROR: ws_server: serverSetFailoverStatus: Marking aznodea_azsr01a down
   :
TRACE: ws_common: websphereExecute: Executing the transaction with the app server
TRACE: ws_                          Stream: Getting the stream to the app server
TRACE: ws_                          StreamDequeue: Checking for existing stream from the queue
ERROR: ws_                          Stream: Failed to connect to app server, OS err=1128
ERROR: ws_                          ecute: Failed to create the stream
ERROR: ws_server: serverSetFailoverStatus: Marking aznodeb_azsr01b down
   :
ERROR: ws_common: websphereWriteRequestReadResponse:
              Failed to find an app server to handle this request
```

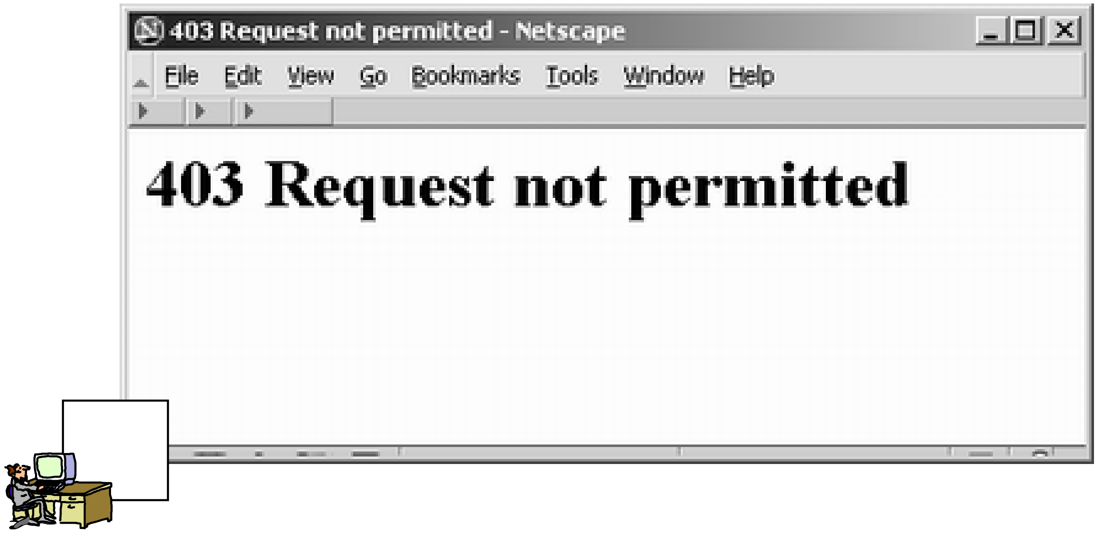
Tries first server in Cluster and fails. Marks server as "down"

Tries second server in Cluster and fails. Marks server as "down"

Runs out of servers in cluster and gives up

"TrustedProxy" Not Set

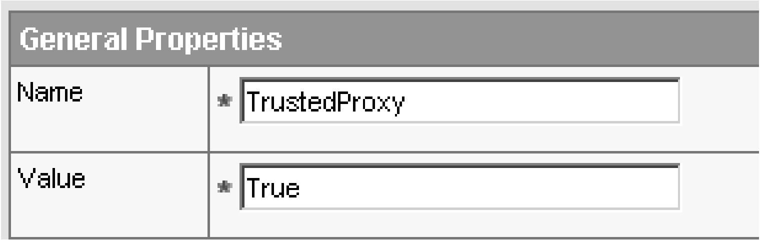
If the server's HTTP port does not have the custom property "TrustedProxy" set, then the server won't permit the flow from the Plugin:


403 Request not permitted - Netscape
File Edit View Go Bookmarks Tools Window Help

403 Request not permitted

The plugin serves as a proxy -- it forwards requests on to the application server. This tells the server to "trust" the inbound request.

Application Servers ⇨ *<server>* ⇨ Web Container ⇨ HTTP Transport ⇨ *<port>* ⇨ Custom Properties



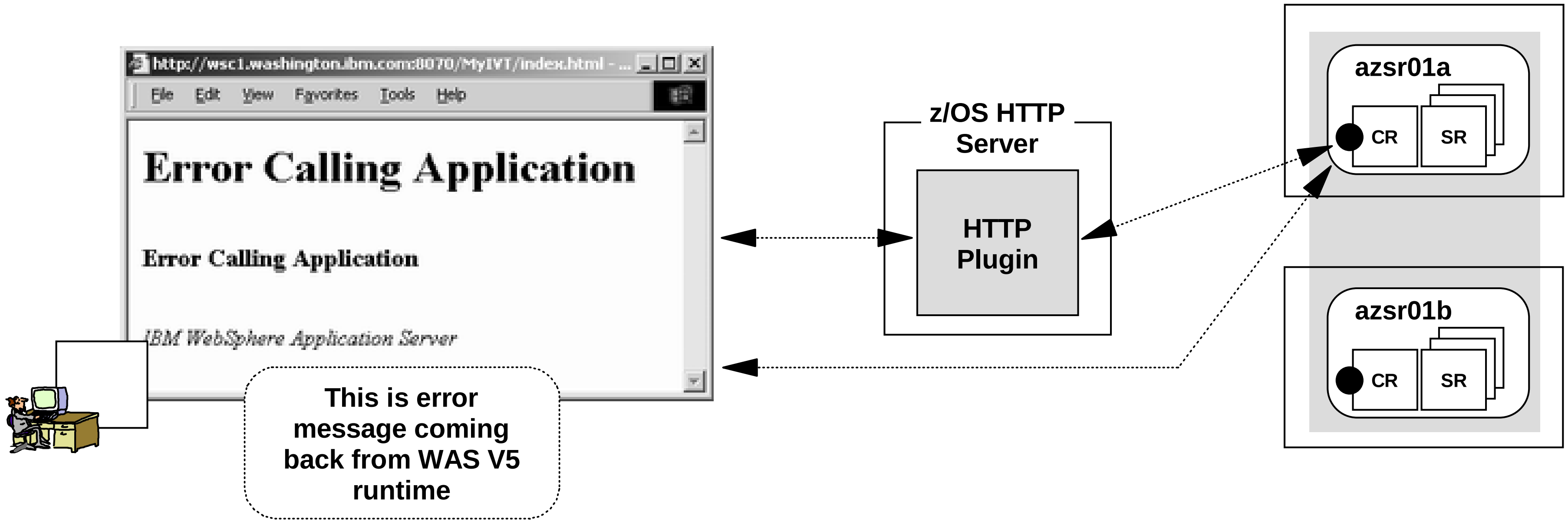
General Properties

Name	* TrustedProxy
Value	* True

- Do this for both the non-SSL and SSL port
- Do this for all servers that receive plugin flows
- Stop/restart server to pick up change

No Virtual Host Match for Client URL

If no virtual host alias in the WebSphere runtime matches the URL sent in by the client, then WAS runtime will reject:



http://wsc1.washington.ibm.com:8070/MyIVT/index.html - ...
File Edit View Favorites Tools Help
z/OS HTTP Server
HTTP Plugin
azsr01a
CR
SR
azsr01b
CR
SR

Error Calling Application

Error Calling Application

IBM WebSphere Application Server

This is error message coming back from WAS V5 runtime

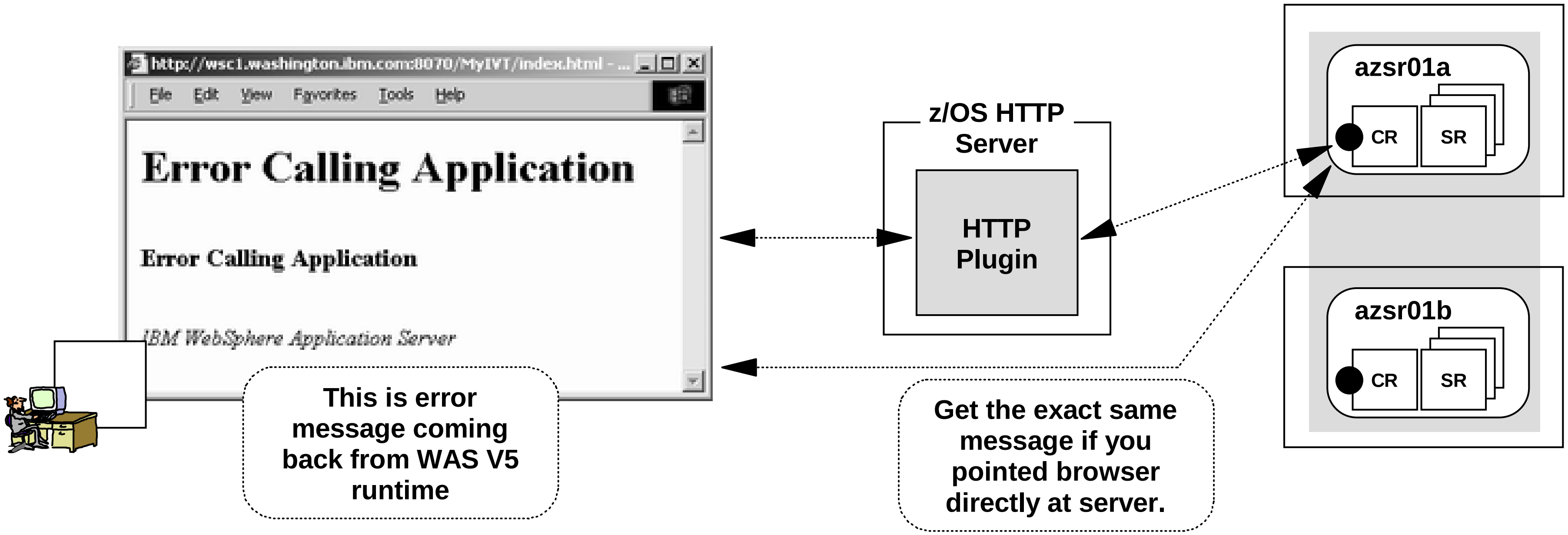
Key Points:

- Plugin is doing its job ... Plugin trace will show normal processing
- This illustrates the difference between virtual host in plugin XML and virtual host in WAS runtime
- Key off the browser error message -- this is Application Server message, which means flow got to application server

All issues related to Plugin initialization, route mapping and servers being up are not the issue

URL Context Root Must Match Appl's

The UriGroup values in the `plugin-cfg.xml` may not match the actual Context Roots in the server. The Plugin will pass the request back, only to have it fail:



http://wsc1.washington.ibm.com:8070/MyIVT/index.html - ...

Error Calling Application

Error Calling Application

IBM WebSphere Application Server

This is error message coming back from WAS V5 runtime

Get the exact same message if you pointed browser directly at server.

Key Points:

- Plugin is doing its job ... Plugin trace will show normal processing
- This illustrates Plugin has no idea what applications are installed

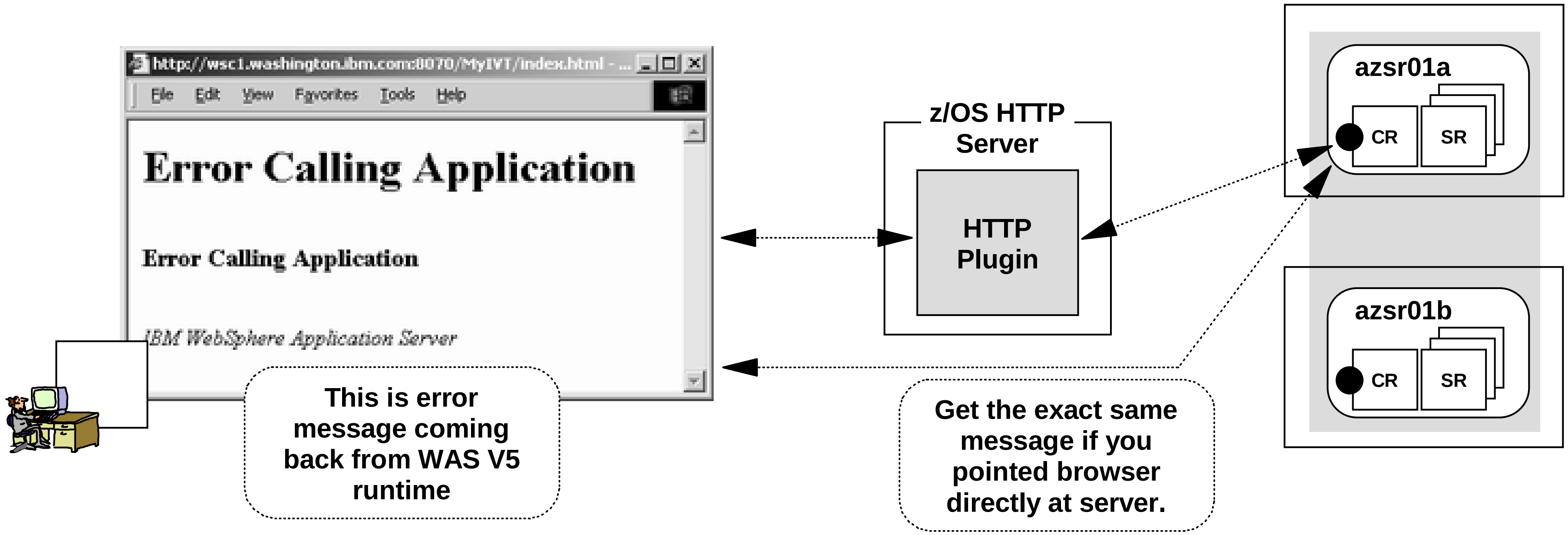
  Generated XML will have the Context Roots of actual applications. But XML file is open to hand-editing and Plugin will send along any request that maps to a route.

- Debugging this will require looking at application server traces

  Plugin trace can be used to determine which application server request went to

Application Must Be Started

If the application is valid in every respect except just not started, then you get error message out of WebSphere Application Server runtime:



http://wsc1.washington.ibm.com:8070/MyIVT/index.html - ...
File Edit View Favorites Tools Help
z/OS HTTP Server
HTTP Plugin
azsr01a
CR
SR
azsr01b
CR
SR

Error Calling Application

Error Calling Application

IBM WebSphere Application Server

This is error message coming back from WAS V5 runtime

Get the exact same message if you pointed browser directly at server.

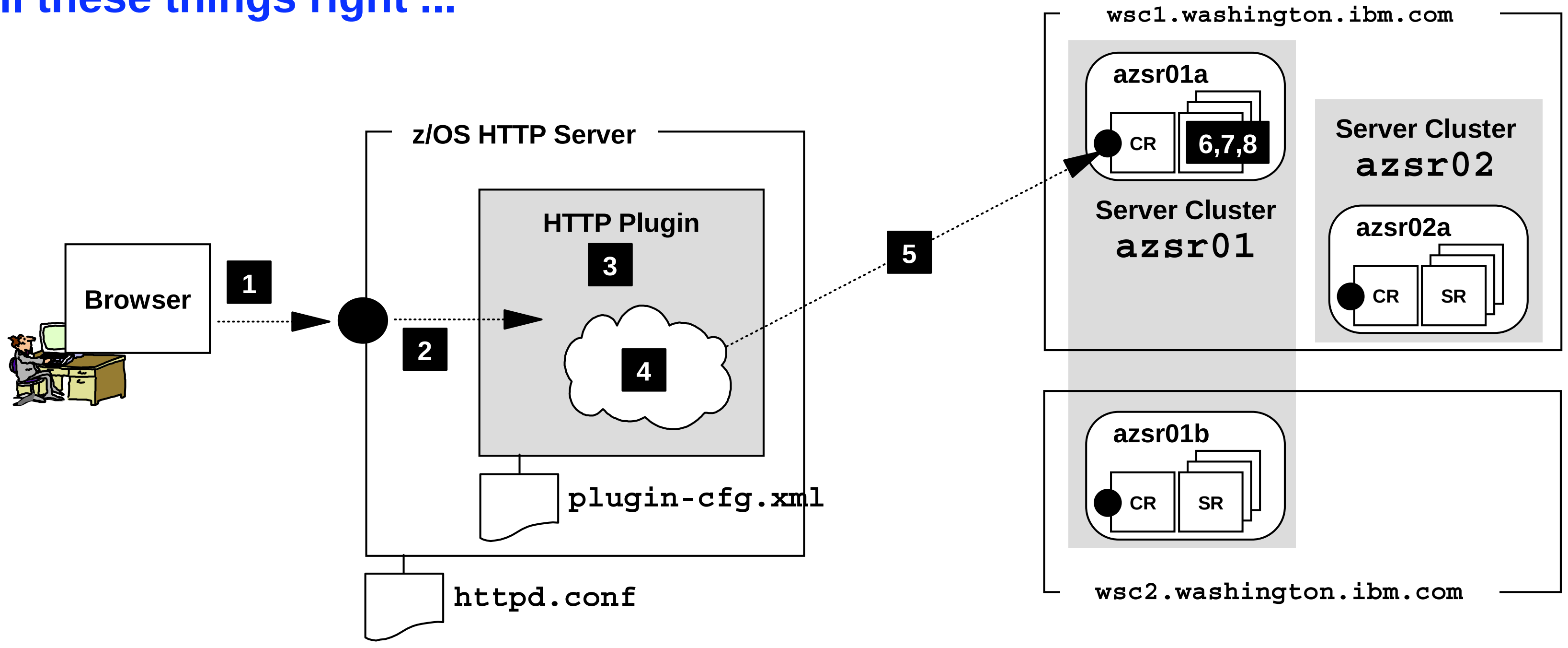
Key Points:

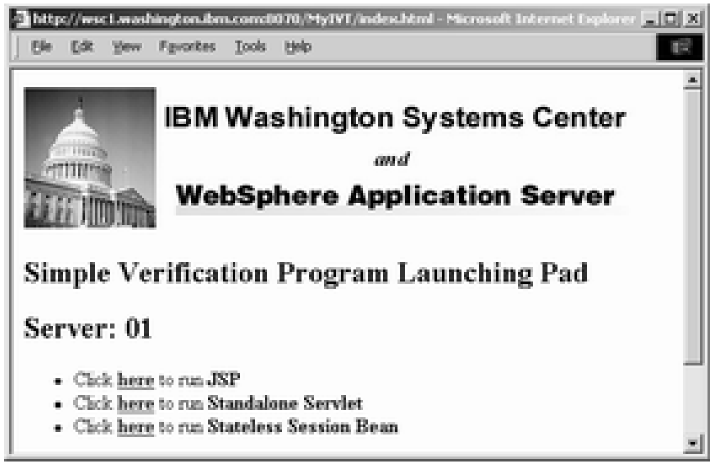
- Plugin is doing its job ... Plugin trace will show normal processing
- This illustrates Plugin doesn't know about application status
- Key off the browser error message -- this is Application Server message, which means flow got to application server

All issues related to Plugin initialization, route mapping and servers being up are not the issue

Success!

Get all these things right ...





http://wsc1.washington.ibm.com:8070/MyIVT/index.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help

IBM Washington Systems Center

and

WebSphere Application Server

Simple Verification Program Launching Pad

Server: 01

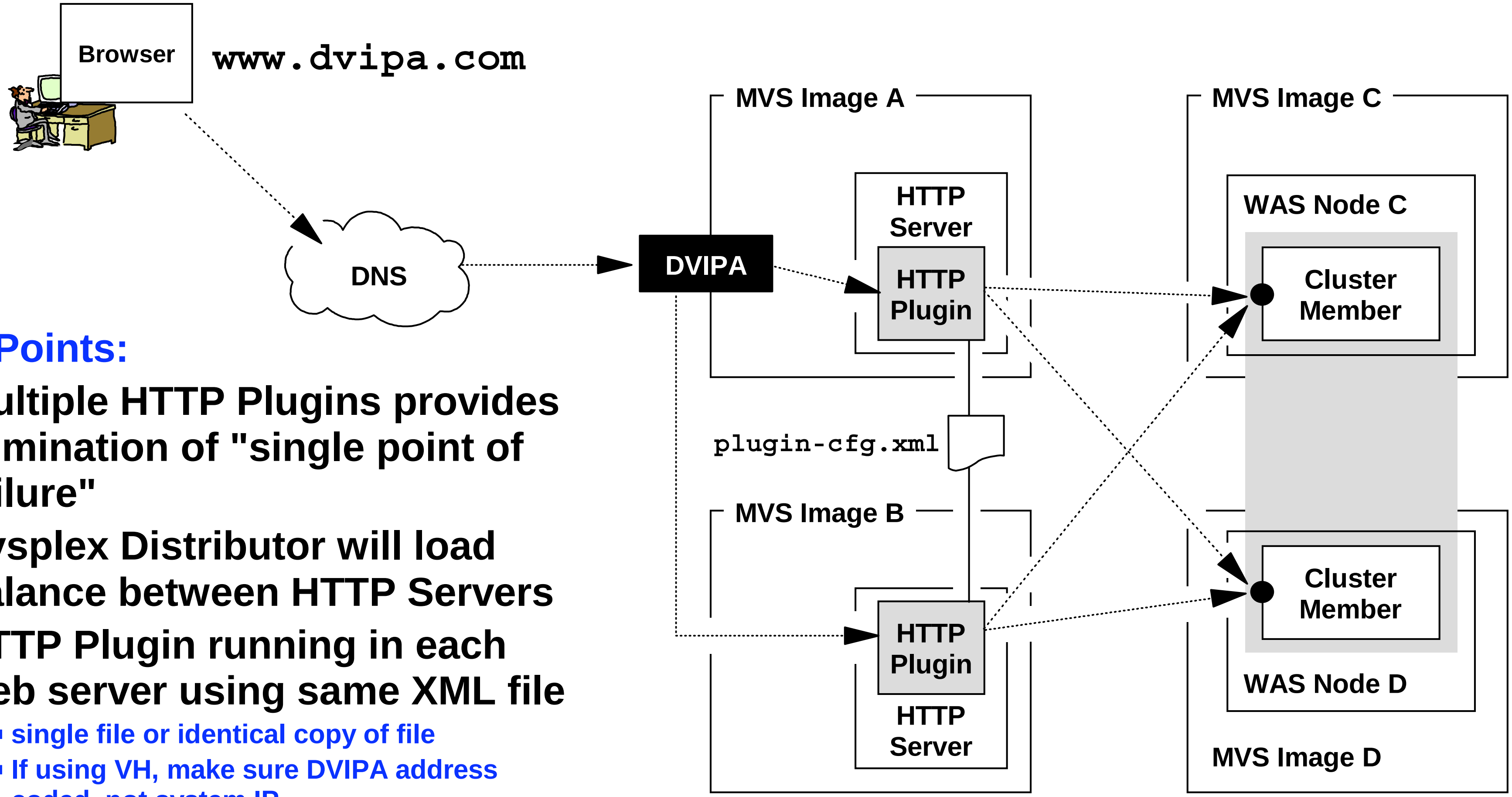
- Click here to run JSP
- Click here to run Standalone Servlet
- Click here to run Stateless Session Bean

And the system will perform and return application result to the browser

Last point: combining Plugin with Sysplex Distributor ...

Sysplex Distributor and Plugin

Even if Session Affinity is a requirement, it's possible to incorporate Sysplex Distributor out front of multiple Plugins:



Key Points:

- Multiple HTTP Plugins provides elimination of "single point of failure"
- Sysplex Distributor will load balance between HTTP Servers
- HTTP Plugin running in each web server using same XML file
  - single file or identical copy of file
  - If using VH, make sure DVIPA address coded, not system IP
- Plugin maintains Session Affinity to backend servers