

# High availability with AppServers

Paul Koufalis • Progresswiz Consulting
Gilles Querret • Riverside Software

## **AppServer Operating Mode**

- Who's using AppServer ?
- We'll only cover Stateless and Statefree appservers
- State-reset and state-aware are deprecated

#### NameServer Overview

- The NameServer is a Java process managed by AdminService
- A NameServer register AppServer brokers by application service name
- When a client ask for a service name,
   NameServer returns host and port number of the AppServer broker
- NameServers only use UDP

## Stateless Appserver

- Agents are not dedicated to a client connection
- Client Appserver interaction :
  - Client connects to the nameserver which provides address of the broker from a list of AppServers
  - Client invokes an appserver request
  - The broker passes the request to any available agent
  - Agent executes the request, returns result to the broker, which returns result to the client
  - If client no longer requires connection, it disconnects from the broker

## Statefree Appserver

- Agents are also not dedicated to a client connection
- Client Appserver interaction :
  - Client sends a request to the NameServer to return a logical connection to an application service
  - NameServer returns a list of AppServers
  - Client creates physical connections for each AppServer broker in the list
  - Client invokes an appserver request
  - The broker passes the request to any available agent
  - Agent executes the request, returns result to the broker, which returns result to the client
  - Client releases physical connections

#### Stateless or State-free?

- Not a DBA / SysAdmin decision
- WebServices require State-free
- Context is easier to maintain in Stateless

 Connect / Disconnect required in order to refresh brokers list

## Simple setup

NameServer NS

**Listening on 10.0.0.10:5162/UDP** 

Accounting 10.0.0.10/3090

Finance 10.0.0.10/3091 Marketing 10.0.0.10/3091

Registration on NS1 Service name : Accounting



1

Registration on NS1 Service names : Finance, Marketing

AppServer broker AS1

**Listening on 10.0.0.10:3090/TCP** 

**AppServer broker AS2** 

**Listening on 10.0.0.10:3091/TCP** 

## NameServer Query

NameServer NS

**Listening on 10.0.0.10:5162/UDP** 

Accounting 10.0.0.10/3090

Finance 10.0.0.10/3091 Marketing 10.0.0.10/3091



Locate Finance in service names

UDP frame to 10.0.0.10:5162
Looking for application service finance

UDP frame back to client Use 10.0.0.10/3091

OE/Java/.Net client

## NameServer Neighborhood

- A NameServer can have neighbors sitting on the same machine
- When NS doesn't recognize an application service name, it forwards the query to its neighbor(s)
- Which will be able to return a broker IP and port number

#### Remote NameServer

- A remote NameServer is just telling NameServer running on a different machine on a specific port
- There's no process to execute
- Used with NS neighbors, so requests can be forwarded to a different server

#### Remote NameServer

NameServer NS Neighbor : RemNS

NameServer RemNS

Listening on 10.0.0.10:5162/UDP

Remote host: 10.0.0.11
Remote port: 5163

Accounting 10.0.0.10/3090



Registration on NS1
Service name: Accounting

**AppServer broker AS1** 

**Listening on 10.0.0.10:3090/TCP** 

NameServer NS2

Listening on 10.0.0.11:5163/UDP

Finance 10.0.0.11/3091 Marketing 10.0.0.11/3091



Registration on NS2
Service names: Finance,
Marketing

**AppServer broker AS2** 

Listening on 10.0.0.11:3091/TCP

## NameServer Query

NameServer NS2 Request NameServer NS NameServer RemNS forwarded to **Neighbor: RemNS N\$2 Remote host: 10.0.0.11 Listening on** Listening on Remote port: 5163 10.0.0.10:5162/UDP 10.0.0.11:5163/UDP No match **Accounting** Finance **Marketing** 10.0.0.10/3090 on NS Request 10.0.0.11/3091 10.0.0.11/3091 forwarded to **RemNS** UDP frame to 10.0.0.10:5162 UDP frame back to client Looking for application service Use 10.0.0.11/3091 finance

OE/Java/.Net client

#### **Use case**

- Proxy nameserver on low-end server
- Dispatch requests to real servers
- Move services to different locations without changing the main server URL

## NameServer Load Balancing

- Multiple AppServers can register under a single service name
- A priority weight can also be given to a broker and NS will dispatch requests accordingly
- Requires NS Load Balancing license

## NameServer Load Balancing

NameServer NS

**Listening on 10.0.0.10:5162/UDP** 

Accounting 10.0.0.10/3090 (25%) 10.0.0.10/3091 (75%)

Registration on NS1 Service name : Accounting

Weight: 10



AppServer broker AS1

**Listening on 10.0.0.10:3090/TCP** 

1

**Registration on NS1** 

**Service name: Accounting** 

Weight: 30

**AppServer broker AS2** 

**Listening on 10.0.0.10:3091/TCP** 

## NameServer Query

NameServer NS

**Listening on 10.0.0.10:5162/UDP** 

Locate Accounting in service names

Accounting 10.0.0.10/3090 (25%) 10.0.0.10/3091 (75%)

UDP frame to 10.0.0.10:5162
Looking for application service
Accounting

UDP frame back to client
Use 10.0.0.10/3090 in 25% of cases
Use 10.0.0.10/3091 in 75% of cases

OE/Java/.Net client

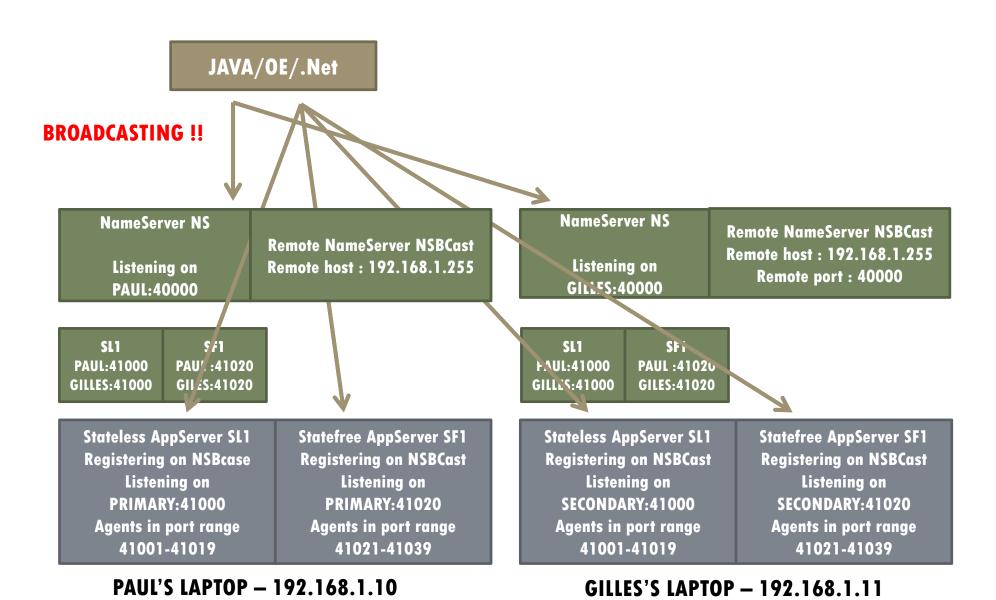
## NS Load Balancing use case

- Remote NS required
- Use case during server upgrades
- Either shut down or set priority weight to 0, then upgrade server
- Restart AS or set priority weight to any non-zero value
- Repeat same steps for every server

#### **UDP Broadcast**

- NameServers have to listen on same port number on a given VLAN
- UDP broadcast packet sent on this VLAN
- First packet received will be used, others will be discarded

#### **NS Broadcast**



## Questions?

## **Progresswiz Consulting**

- Based in Montréal, Québec, Canada
- Providing technical consulting in Progress<sup>®</sup>,
   UNIX, Windows, MFG/PRO and more
- Specialized in
  - Availability and business continuity planning
  - Performance tuning
  - Security

#### **Riverside Software**

- Based in Lyon, France
- Technical expertise in OpenEdge and Java environments
- Working on continuous integration process, automated deployment and source code analysis

contact@riverside-software.fr