*PUG Challenge Americas*

# 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

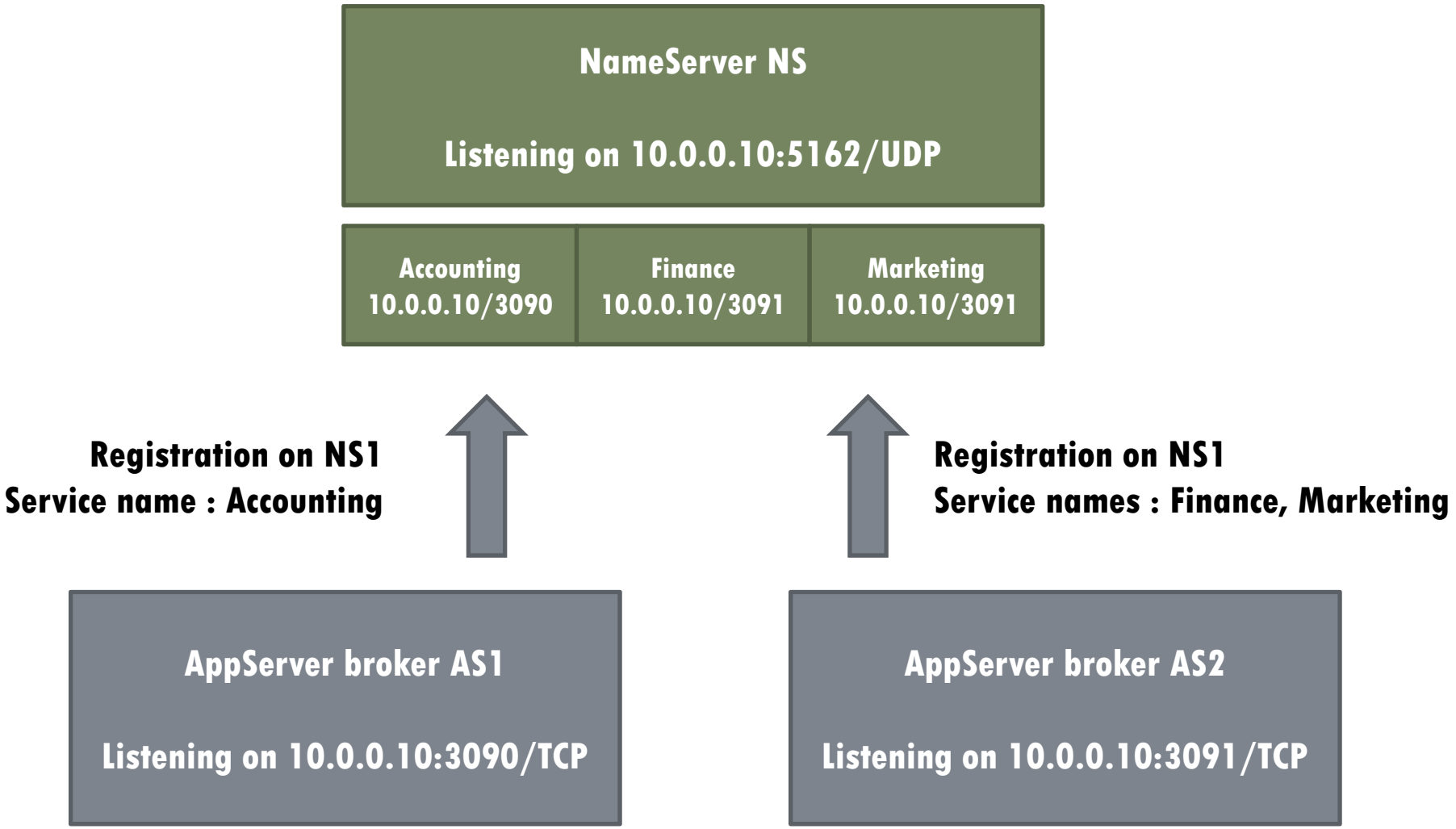
- Agents are also not dedicated to a client connection
- Client – Appserver interaction :
  - Client sends a request to the NameServer to return a logical connection to an application service
  - NameServer returns a list of AppServers
  - Client creates physical connections for each AppServer broker 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 ?

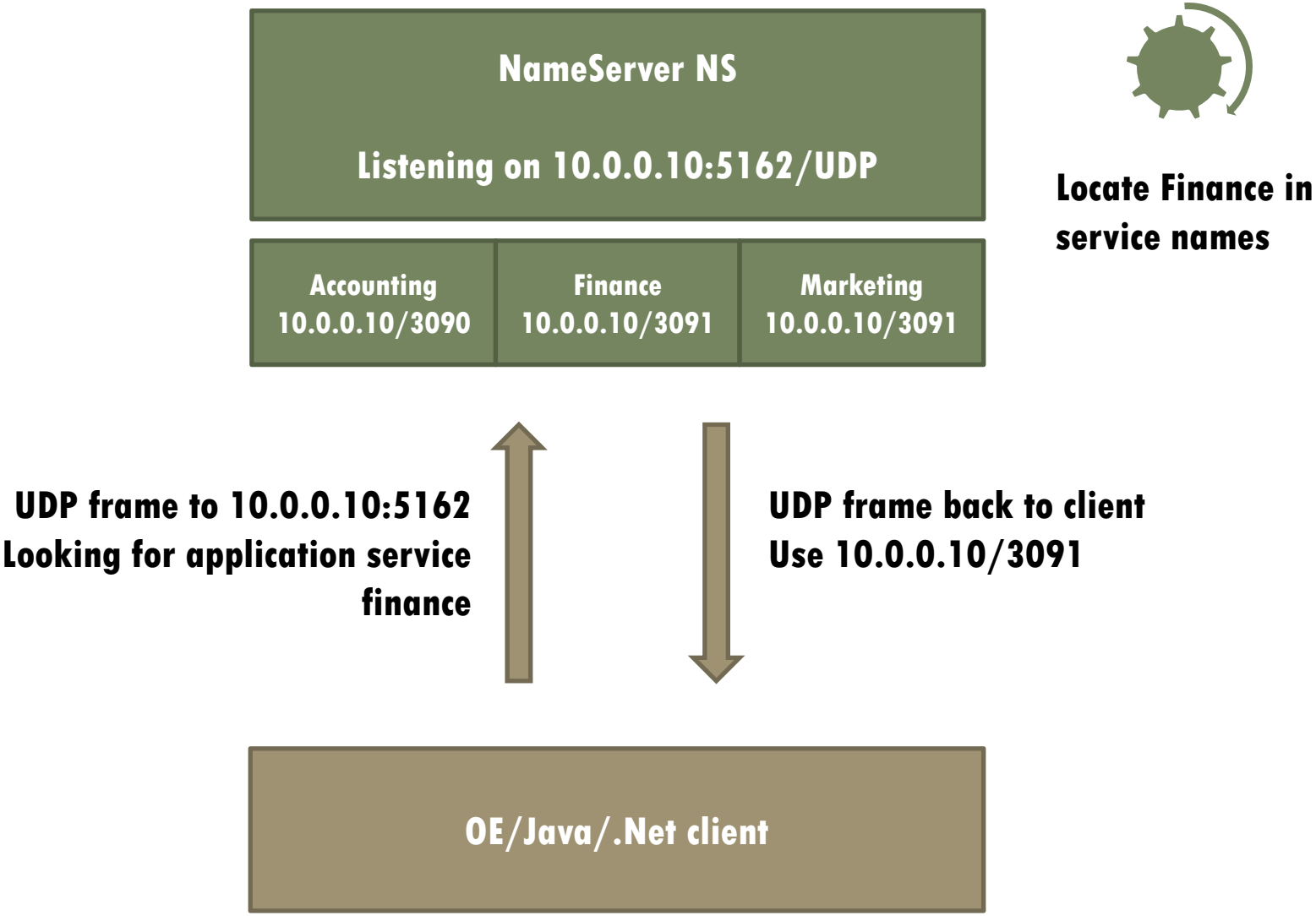
- Not a DBA / SysAdmin decision
- WebServices require State-free
- Context is easier to maintain in Stateless

- Connect / Disconnect required in order to refresh brokers list

# Simple setup



# NameServer Query



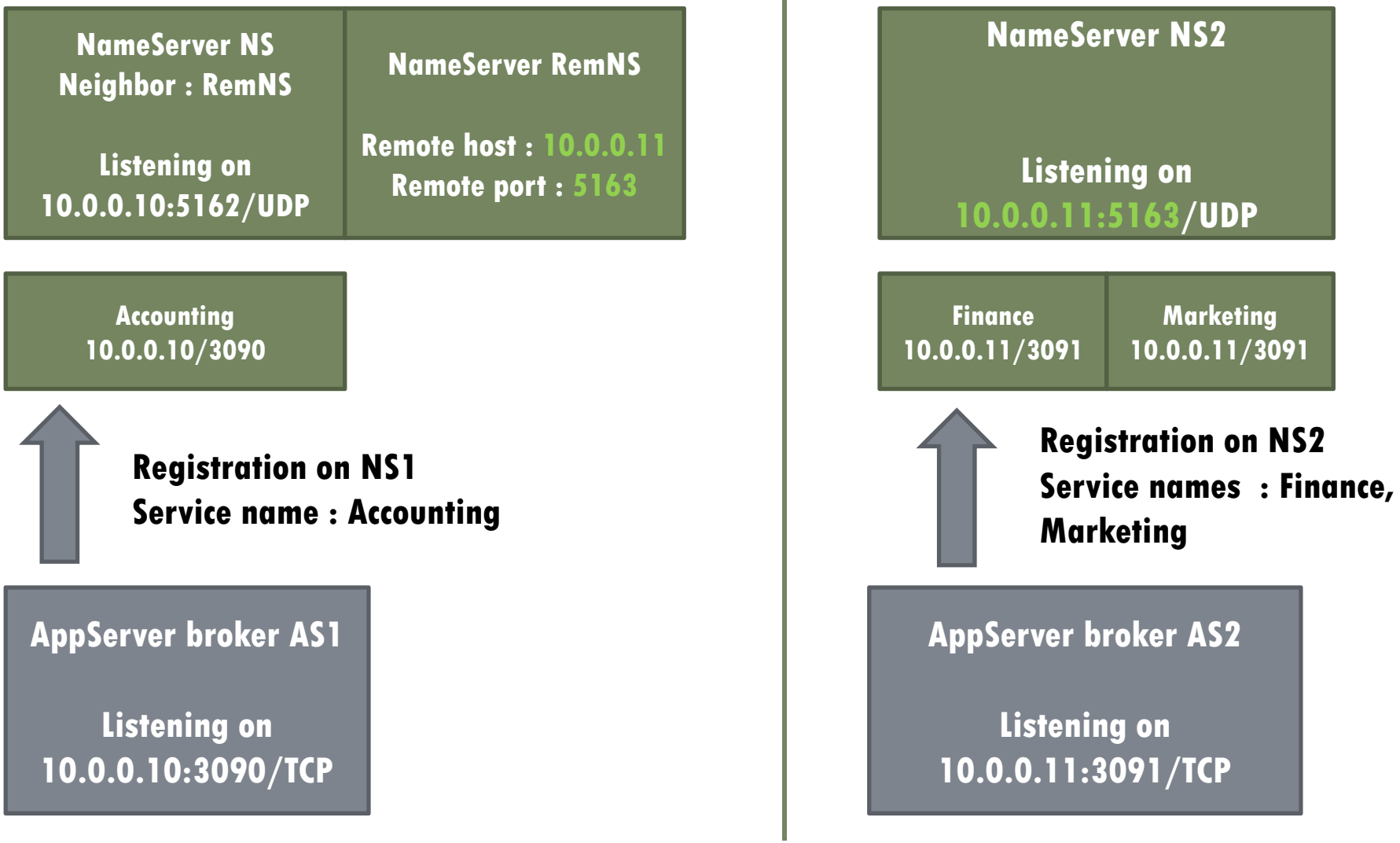
# NameServer Neighborhood

- 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

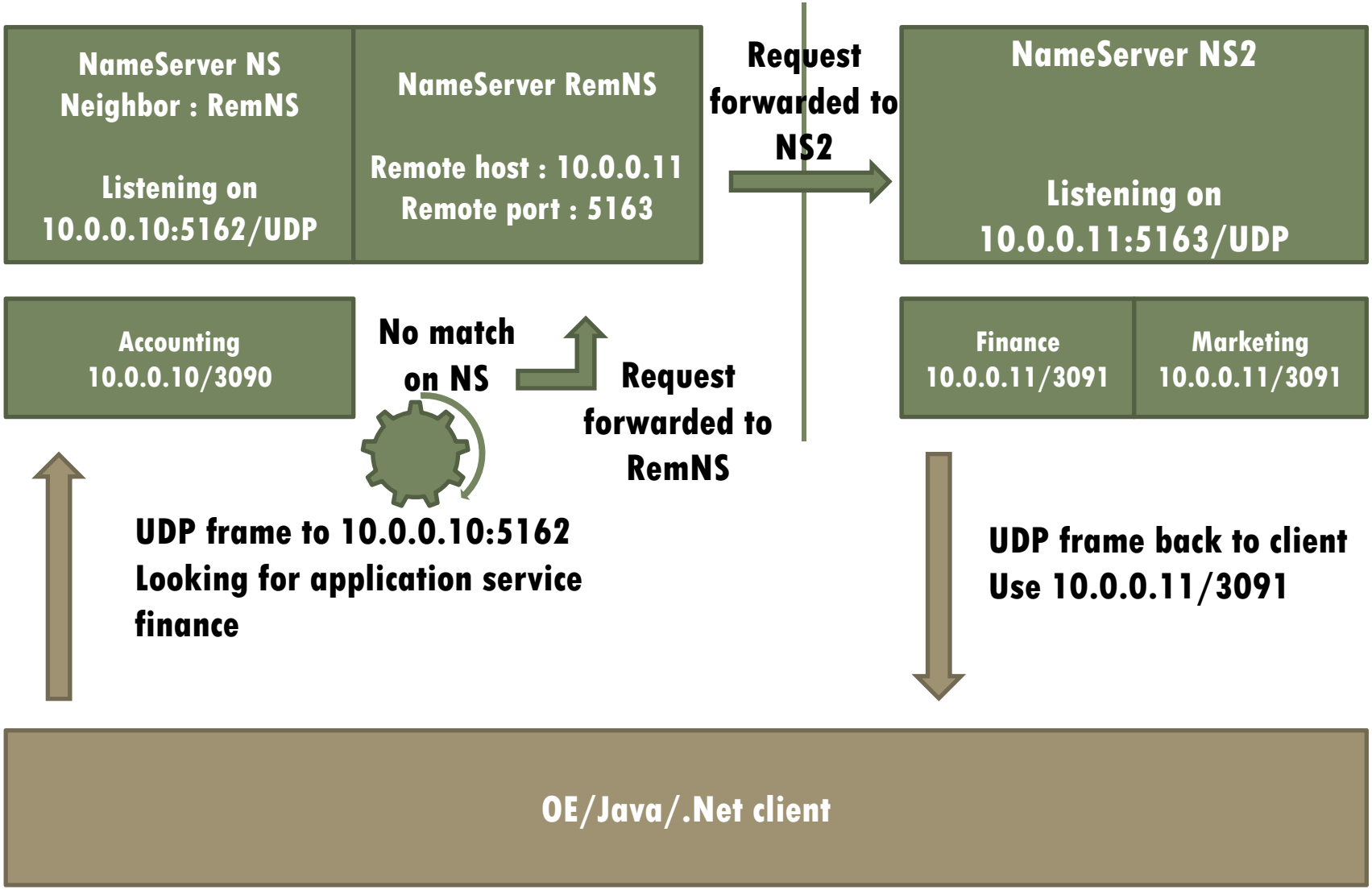
# Remote NameServer

- 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 Query



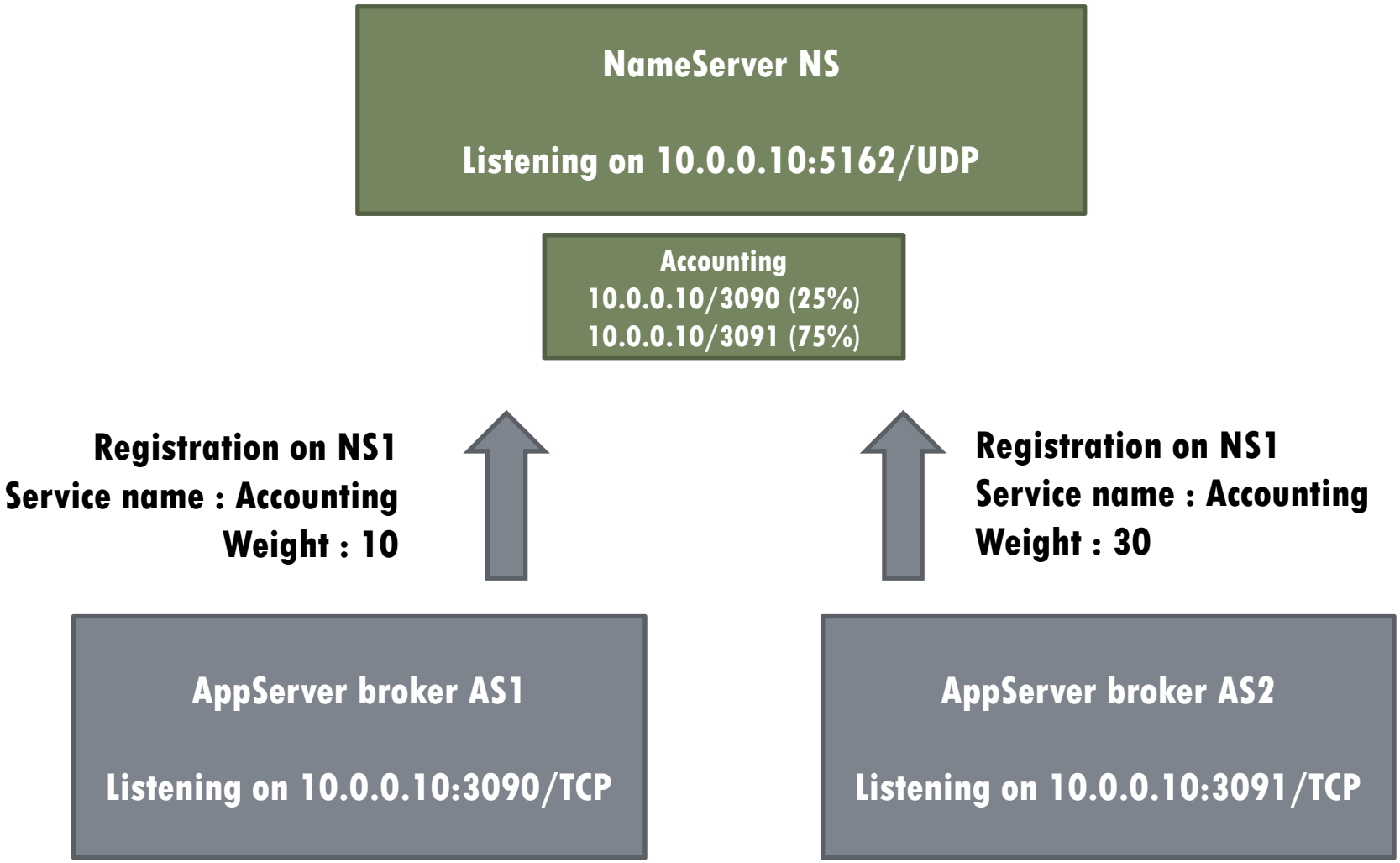
# Use case

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

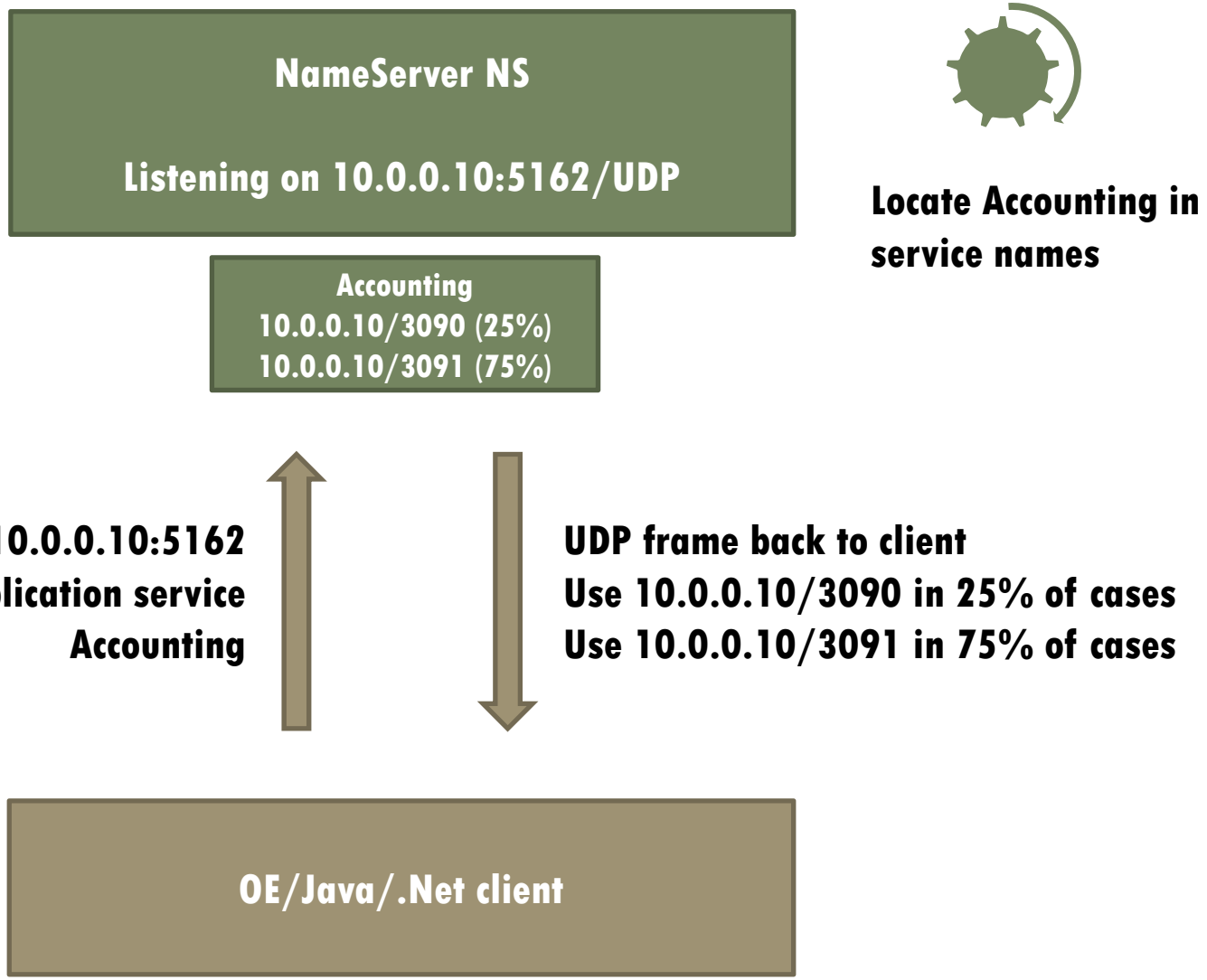
# NameServer Load Balancing

- 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 Query



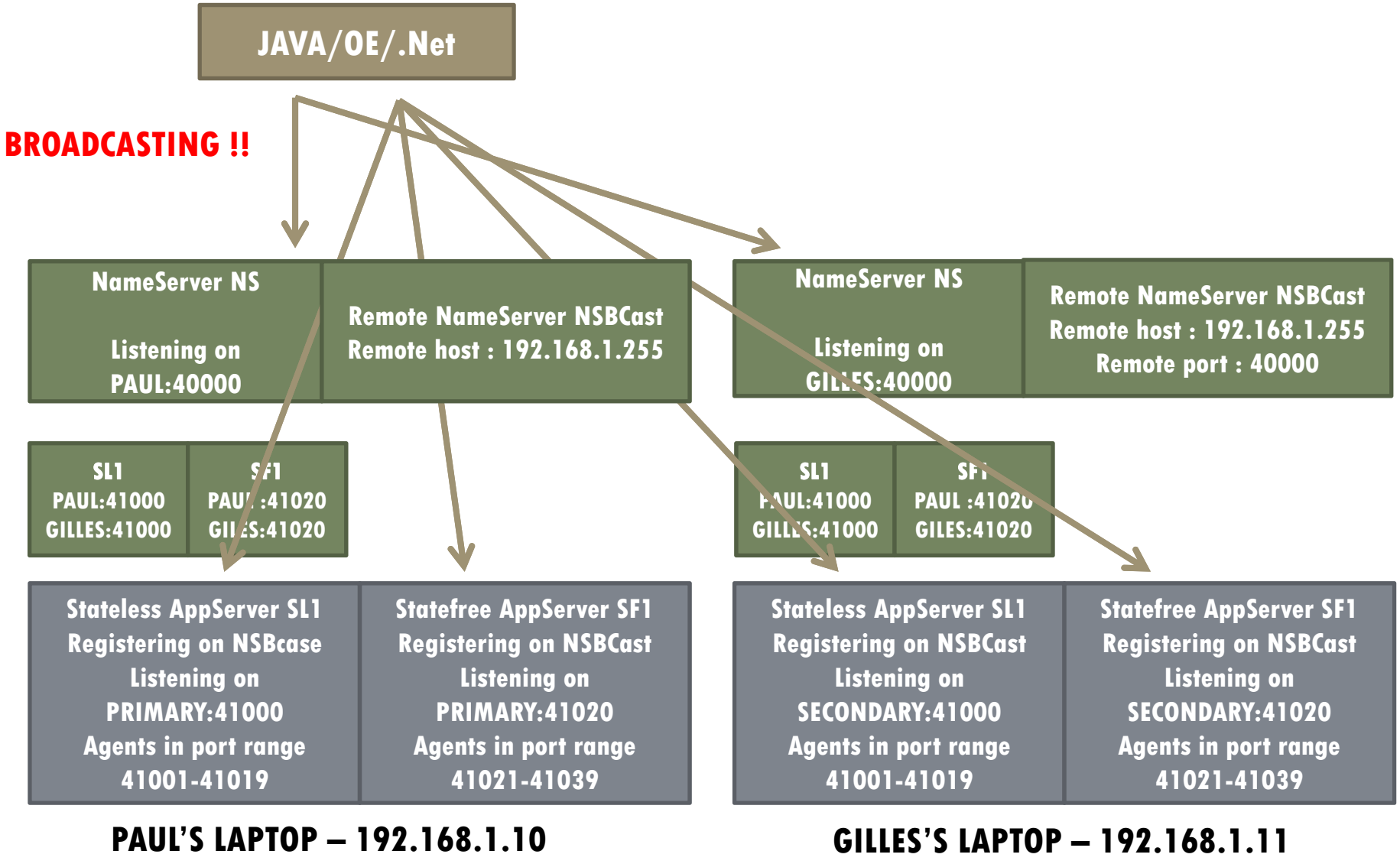
# NS Load Balancing use case

- 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®, 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