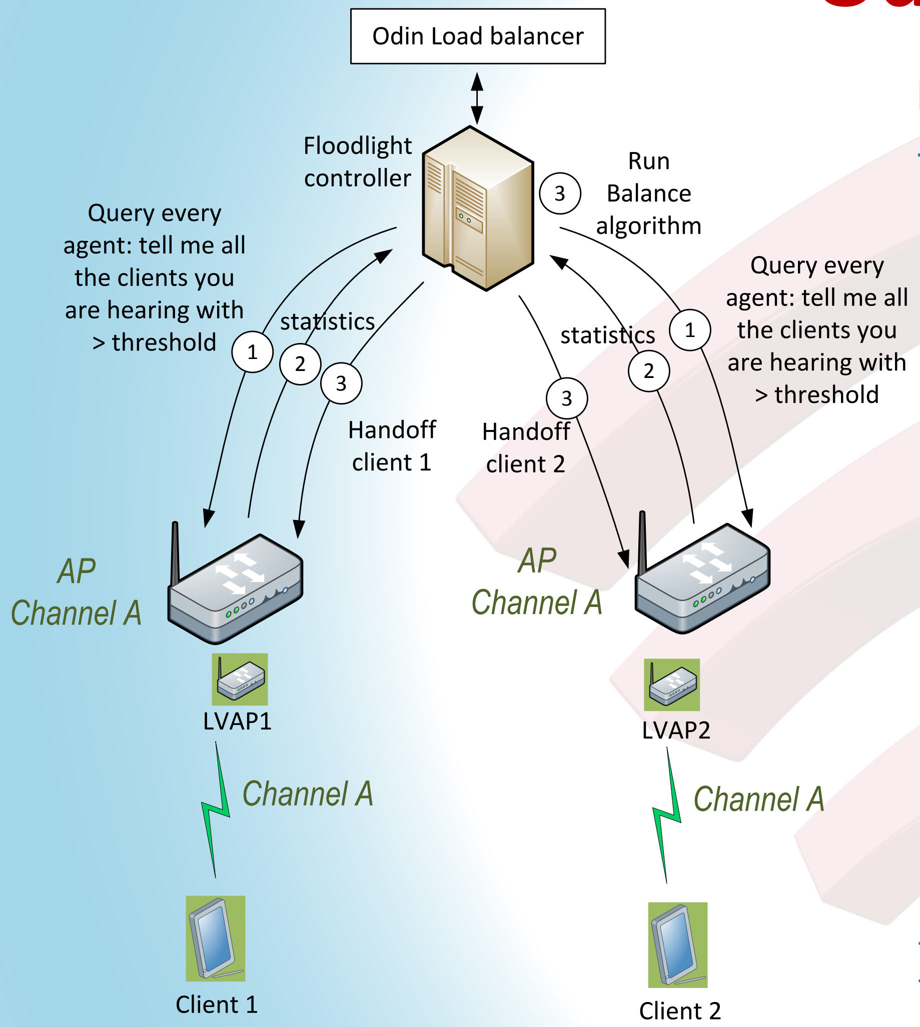
Odin Load Balancer: Proactive application

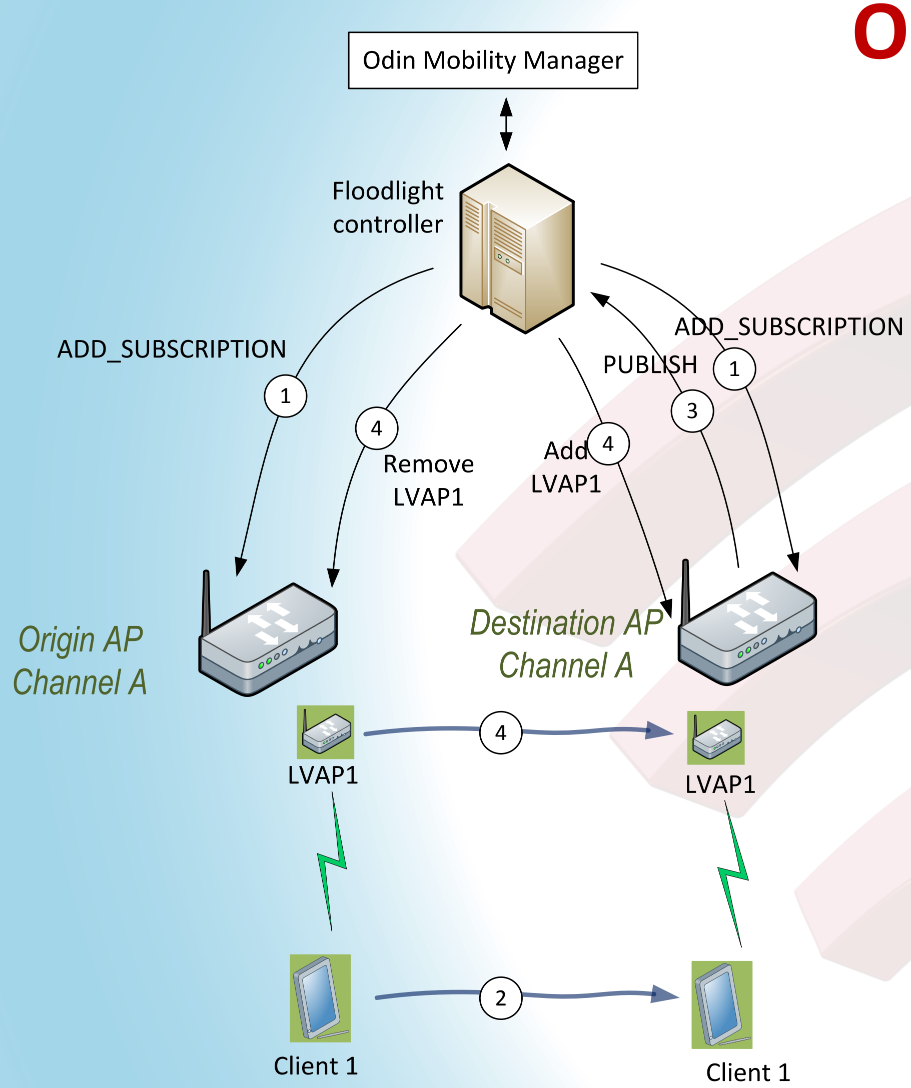


```
Every 60 seconds
{
        // build the “hearing table”
        For every agent
        {
                obtain the IPs and MACs heard above a threshold
        }
        // result: updated map of agents and clients they hear

        // run balance() function
        For every client
        {
                find the most suitable AP
                Handoff the client if necessary
        }
}
```

Odin Load Balancer: master/src/main/java/net/floodlightcontroller/odin/applications/SimpleLoadBalancer.java
Odin Agent: odinagent.cc

Odin Mobility Manager: Reactive application



```
For every Agent
{
        register subscription (if client_strength > threshold, call event handler)
}
----------------
//Event handler (triggered by a PUBLISH message from the AP)
{
        if the client is already in the AP
        {
                update statistics
        } else {
                if (hysteresis_period is finished) {
                        if (compare_signal_strength == true) {
                                handoff
                        }
                }
        }
}
```

Odin Load Balancer: master/src/main/java/net/floodlightcontroller/odin/applications/OdidnMobilityManager.java
Odin Agent: odinagent.cc