

# Profound.js Clustering

Node.js by default is using a single core of a processor for code execution. Therefore, Node.js introduced a cluster module to spawn processes. "Cluster" was introduced to scale an application execution on multiple processor cores by creating worker processes. Worker processes share a single port, therefore, requests are routed through a shared port.

An alternate approach for scaling an application is to set up multiple Profound.js instances and use a load-balancer. The load-balancer method will allow you to scale to multiple servers and supports the setup of stateless and stateful apps.

This page shows the changes required to use clustering with Profound.js.

You can read more about that cluster module on the Node.js website <https://nodejs.org/api/cluster.html>

Create a new config file for the cluster module in your Profound.js installation directory (/profoundjs by default) called cluster.json

**profoundjs/cluster.json**

```
{  
  "workers": 4,  
  "min_workers": 4  
}
```

Modify the start.js file to include the cluster module

profoundjs/start.js

```
#!/usr/bin/env node

var cluster = require("cluster");

if (cluster.isMaster) {
  var conf = require("./cluster.json");

  // Figure out number of workers.
  var numWorkers = conf.workers;
  if (numWorkers == 0) {
    var numCPUs = require("os").cpus().length;
    if (numCPUs < conf.min_workers) {
      numWorkers = conf.min_workers;
    } else {
      numWorkers = numCPUs;
    }
  }

  // Fork workers.
  //console.log("Starting " + numWorkers + " workers.");
  for (var i = 0; i < numWorkers; i++) {
    cluster.fork();
  }

  cluster.on('exit', (worker, code, signal) => { });
} else {

  // Load Profound.js
  var profoundjs = require("profoundjs");

  // Apply configuration
  var config = require("./config.js");
  profoundjs.applyConfig(config);

  // Start Profound.js server
  var isWorker = profoundjs.server.listen();
  if (isWorker) {

    // This is the top-level Express Application.
    // Custom Express coding can be added here.
    var express = profoundjs.server.express;
    var app = profoundjs.server.app;
    app.use(express.json()); // default to use JSON-encoded post data
  }
}
```