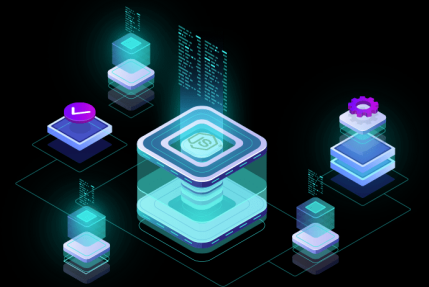


OVERVIEW

PM2 (Process Manager 2) is an open source node.js process management tools in production environment. Some key features such as automatic load balancing over multi-core CPU、declarative application configuration、high availability、monitoring、scalable and so on.



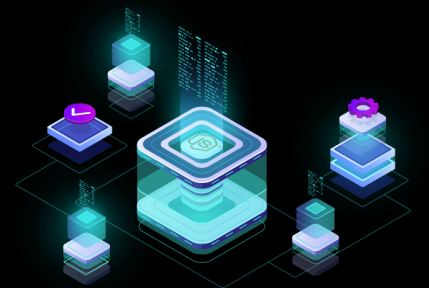


Quick Start & Process Management

- **Install**
 - `[sudo] npm install pm2 -g`
- **Generate configuration file**
 - `pm2 init`
 - `pm2 init simple`
- **Modify and custom configuration file**
 - ```
{

 name: 'server-end',
 script: 'app.js',
 cwd: '/home/express/server-end',
 watch: false,

}
```



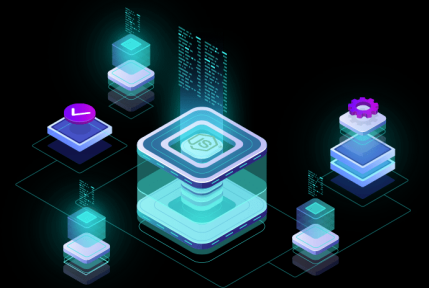


- Start (stop,restart,reload,delete) application
  - `pm2 start ecosystem.config.js`
  - `pm2 start ecosystem.config.js --env test`
  - `pm2 reload ecosystem.config.js --only server-end`
  - `pm2 del <app_name>`
  - `pm2 restart all`
  - `pm2 stop <app_id>`
- 0-second-downtime reload
  - `pm2 reload <app_name>`
  - `pm2 reload <app_name> --update-env`
- View logs
  - `pm2 logs`
  - `pm2 log <app_name>`
- Scale application
  - `pm2 scale <app_name> <number>`
  - `pm2 scale <app_name> <+number>`





- Graceful restart/reload
  - `pm2 startOrRestart <ecosystem.config.js>`
  - `pm2 startOrReload <ecosystem.config.js>`
  - `pm2 startOrGracefulReload <ecosystem.config.js>`
- View info
  - `pm2 ls`
  - `pm2 info <app_name|app_id >`
  - `pm2 monit`



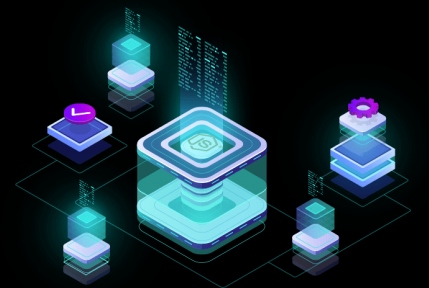
# Custom Configuration

- **App name**
  - **name: string**
  - **We can use this name to manage this application later**
- **Startup file**
  - **script: string**
  - **The main file to startup**
- **Project path**
  - **cwd: string**
  - **The directory of app launch**



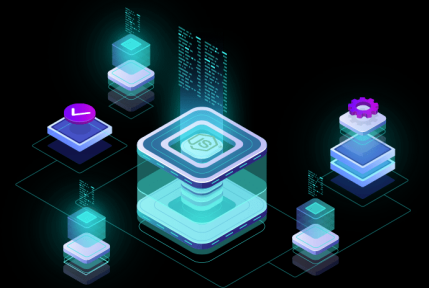


- Number of instances
  - instances: string/number
  - The number of app instance
    - 0/max Equal to the number of CPU cores
    - 3 3 app instance
    - -1 CPU cores - 1
    - -2 CPU cores - 2
- Cluster mode
  - exec\_mode: string
  - Cluster mode "fork" or "cluster", node.js use "cluster"
- Watch and automatic reload
  - watch: Boolean / string
  - if a file change in the folder or subfolder, the app will reload





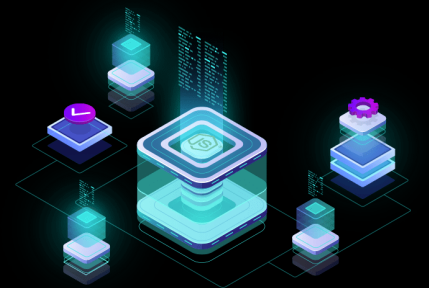
- Memory limit (restart)
  - `max_memory_restart`: string
  - The app will be restarted if it exceeds the amount of memory specified. human-friendly format : it can be "10M", "100K", "2G" and so on...
- Inject environment variables
  - `env`: object
  - `env_`: object
  - It will inject environment variables before lunch
  - `pm2 start ecosystem.config.js --env production`
  - Use `--update-env` to update environment variables
- Log files
  - `log_date_format`: string
  - `merge_logs`: boolean
    - Merge output and error
  - `out_file`: string
    - Output file path, can be `'/dev/null'` to disable logs saving
  - `error_file`: string
    - Error file path, can be `'/dev/null'` to disable logs saving







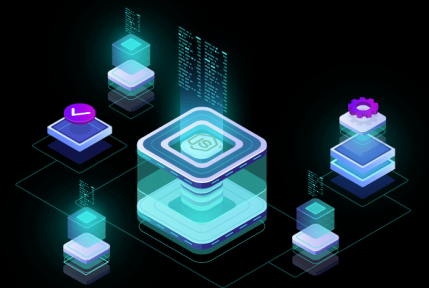
- Control flow
  - min\_uptime
  - listen\_timeout
  - kill\_timeout
  - ....



# GRACEFUL START/SHUTDOWN

- Graceful Stop

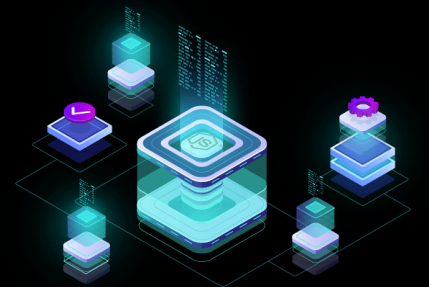
- To allow graceful restart/reload/stop processes, make sure you intercept the SIGINT signal and clear everything needed (like database connections, processing jobs...) before letting your application exit.
- ```
process.on('SIGINT', function() {  
    db.stop(function(err) {  
        process.exit(err ? 1 : 0);  
    });  
});
```





- Graceful start

- `wait_ready: true` in a configuration file
- `add process.send('ready')`
 - ```
var listener = app.listen(0, function() {
 // Here we send the ready signal to PM2
 process.send('ready');
});
```



# Startup Script Generator

- **Auto start**
  - **pm2 startup**
  - **PM2 can generate startup scripts and configure them in order to keep your process list intact across expected or unexpected machine restarts.**
  - **pm2 save**

