



INSTITUTO FEDERAL  
PIAUÍ  
Campus Parnaíba

# NodeJs - Introdução

Prof. Msc Denival A. dos Santos

# Introdução

## ▪ O que é?

- Node.js é um ambiente de execução de código JavaScript no lado do servidor, criado em 2009.
- Construído sobre o motor de JavaScript do Google Chrome: o **V8**.
  - Motor de JavaScript open source de alto desempenho criado e mantido pelo Google.
  - Escrito em C++.
  - Usado no Google Chrome, Node.js e outros projetos.
- Usa event-driven, modelo de E/S não bloqueante, que o torna leve e eficiente.
- Os pacotes disponíveis é o maior ecossistema de bibliotecas open-source do mundo.

# Introdução

## ■ Vantagens

- Rápido
- Escalável
- Grande comunidade
- Tudo em JavaScript
- Streams, Socket.io, etc.

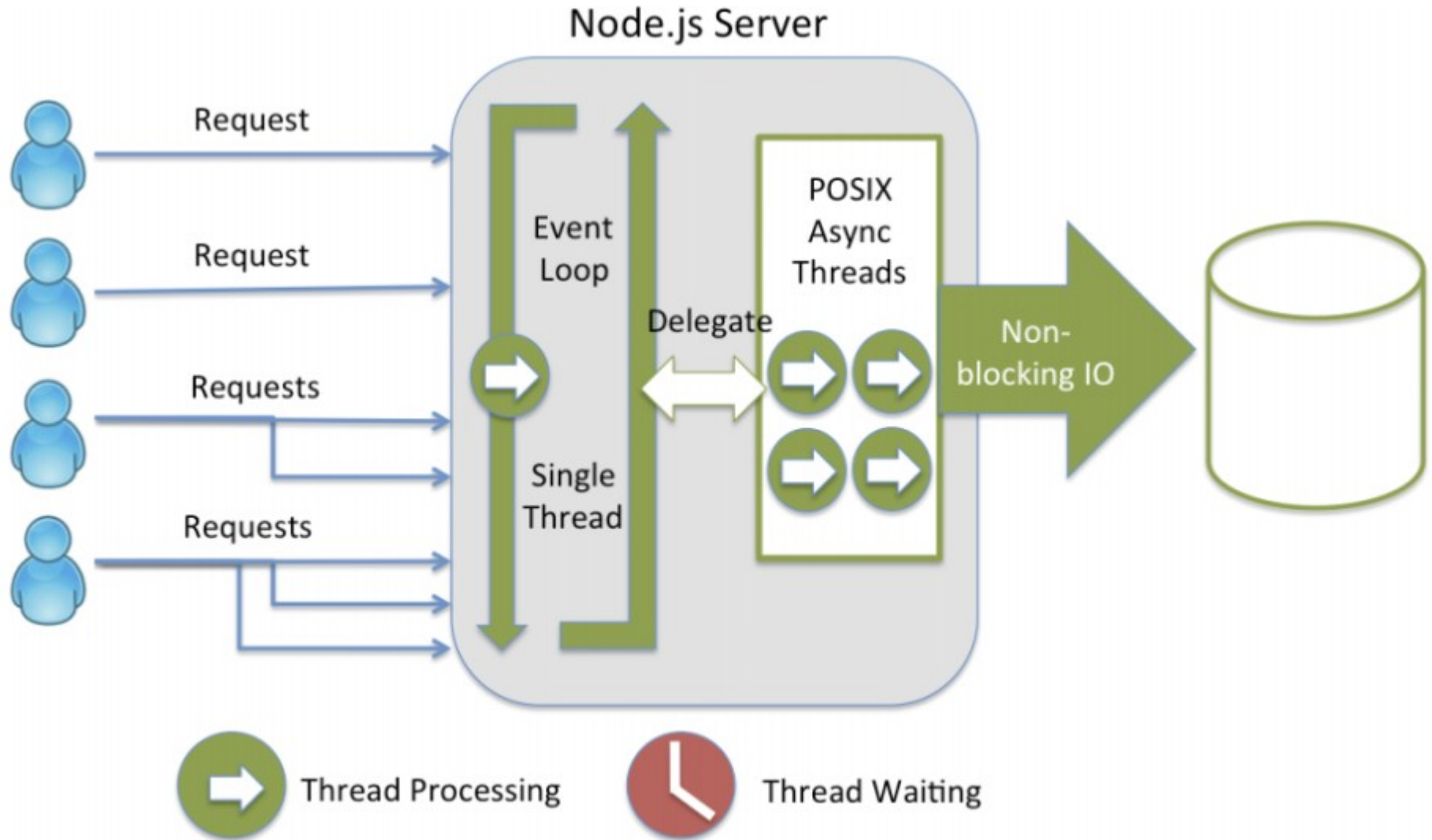
## ■ Desvantagens

- Uso intenso de CPU.
- Curva de aprendizado longa.

# Introdução

- O Node.js é composto basicamente por:
  - Um runtime JavaScript (Google V8, o mesmo do Chrome);
  - Uma biblioteca para I/O de baixo nível (libuv);
  - Bibliotecas de desenvolvimento básico (os core modules);
  - Um gerenciador de pacotes via linha de comando (NPM);
  - Um gerenciador de versões via linha de comando (NVM);
  - Utilitário REPL via linha de comando;

# Introdução



# Comandos

- Verificar se esta funcionando e qual a versão.

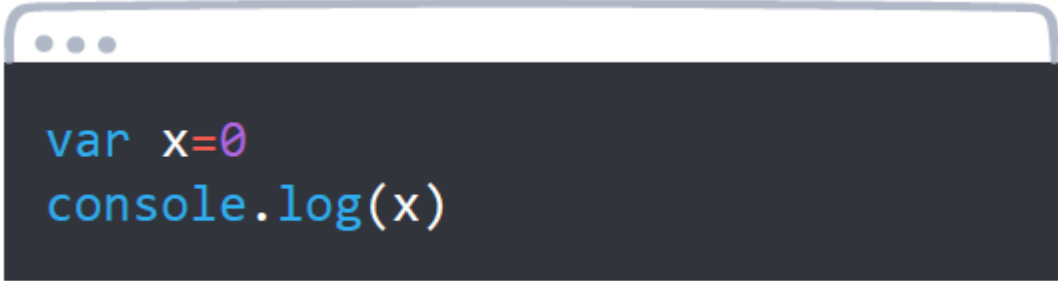
**node -v**

- Invocar o utilitário REPL do Node.js (Read-Eval-Print-Loop)

**node**

- Execução de scripts

**node <arquivo>**

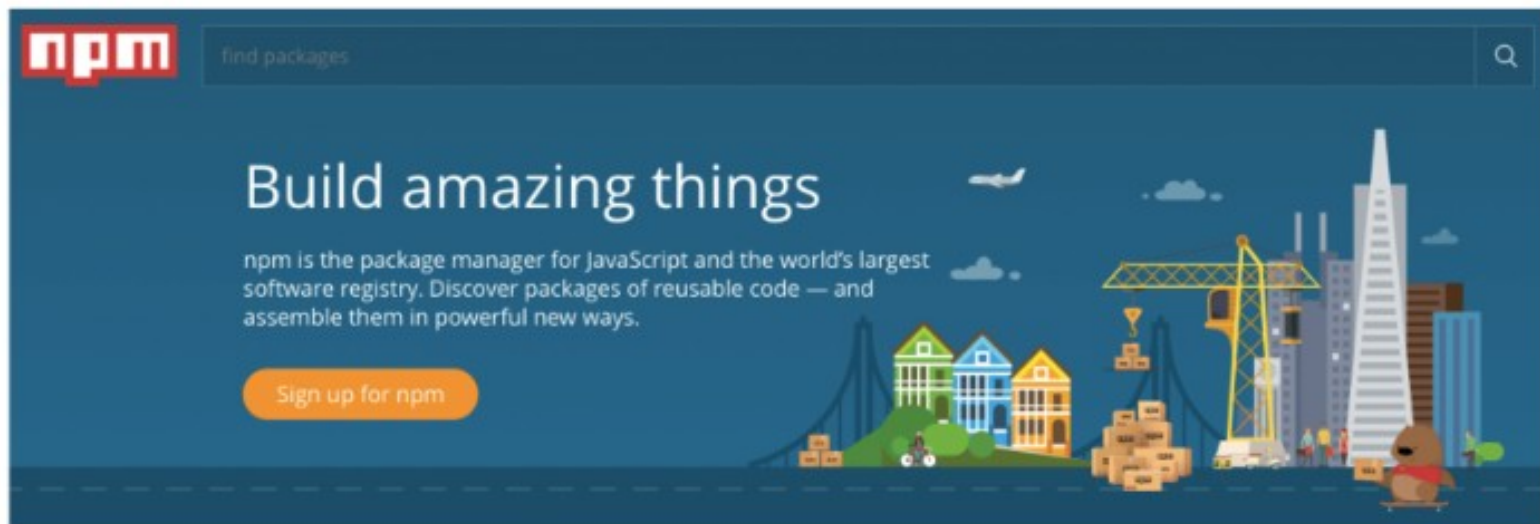
A terminal window with a dark background and light blue text. It contains two lines of JavaScript code: 'var x=0' and 'console.log(x)'. The window has a title bar with three dots on the left.

```
var x=0
console.log(x)
```

# Módulos nativos

- Buffer
- C/C++ Addons
- Child Processes
- Cluster
- Console
- Crypto
- Debugger
- DNS
- Errors
- Events
- File System
- Globals
- HTTP
- HTTPS
- Modules
- Net
- OS
- Path
- Process
- Punycode
- Query Strings
- Readline
- REPL
- Stream
- String Decoder
- Timers
- TLS/SSL
- TTY
- UDP/Datagram
- URL
- Utilities
- V8
- VM
- ZLIB

- Gerenciador de pacotes do node.js.
  - node package manager
- <https://www.npmjs.com/>
- `npm -v`
- `npm install <pacote>`





# Utilizar o pacote

- **Require (pacote)** - é a maneira mais fácil de incluir módulos existentes em arquivos separados.
- Require ler o arquivo JavaScript, interpreta o script e em seguida retorna o conteúdo do objeto exports.



```
log.js
module.exports.log = function (msg) {
  console.log(msg);
}

app.js
1 var msg = require('./log.js');
2
3 msg.log('Hello World');
4
```

# Utilizar o pacote

```
= function soma(x,y) {  
  |   console.log(x+y) ;  
  | }  
  
= function subtracao(x,y) {  
  |   console.log(x-y) ;  
  | }  
  
= function multiplicacao(x,y) {  
  |   console.log(x*y) ;  
  | }  
  
= function divisao(x,y) {  
  |   console.log(x/y) ;  
  | }  
  
= module.exports = {  
  |   soma,  
  |   subtracao,  
  |   divisao  
  | };  
=
```

# Frameworks para Node.js

## ■ Ejemplos

- Express.js
- Hapi.js
- Sails.js
- Meteor
- Geddy
- Etc.