node.js Introduction

What is node.js?

- An open-source runtime environment that contains JavaScript code for execution in the server to aid in asynchronous request-response strategy.
- A library with variety of modules needed for network and web application development.
- Is developed by Ryan Dahl in 2009.
- Is part of V8 JavaScript, a Google's implementation for the Chrome browser, with "Just in Time" compilation services to achieve high speeds for executions.
- Uses a non-blocking, event-driven I/O environment to handle high-critical real-time applications running across distributed devices.
- Prevents its applications from data buffers, regardless of its extended service processing tasks.

How node.js works?

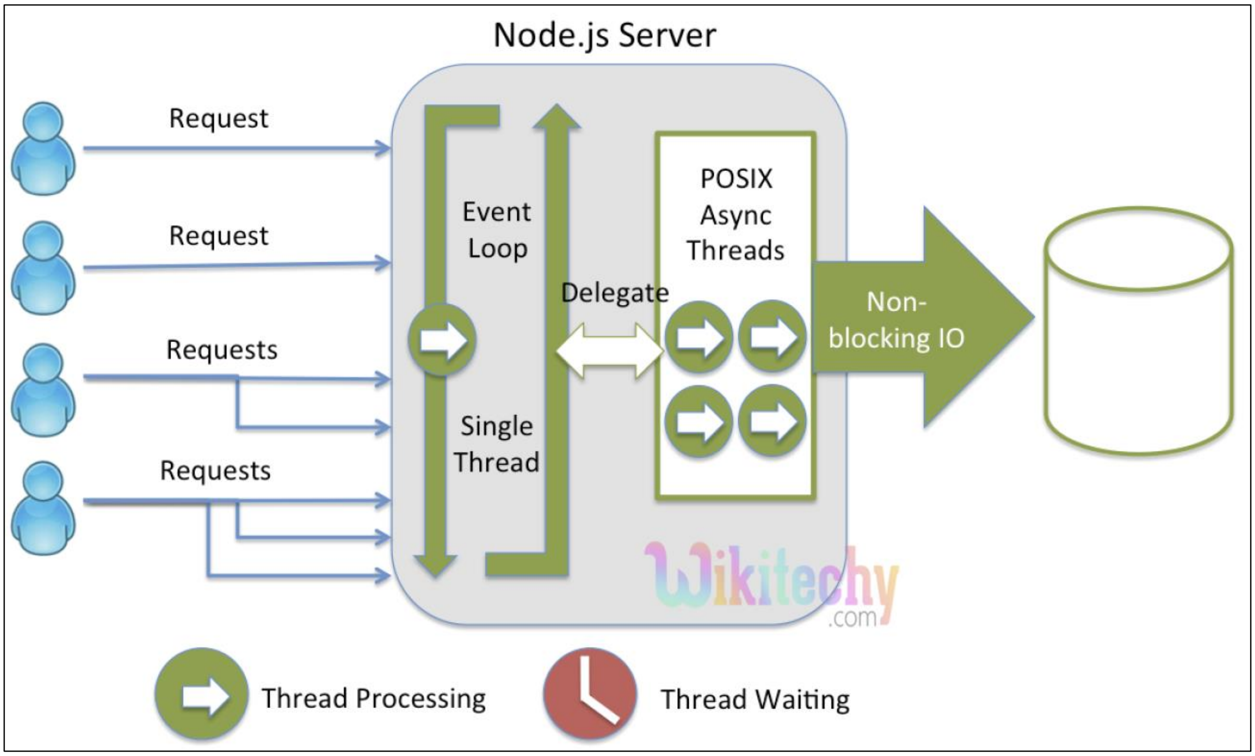

Fig1. Node.js Handling Request

1. Multiple requests are sent to the Node.js server.
2. The event-driven nature of Node.js creates a loop of events for all requests and handles them under a single thread.
3. These events are then delegated to the appropriate non-blocking event handler.
4. The response received travels the reverse way to the requested end point.
5. Since the process happens under a single thread, the response is sent asynchronously without causing page refresh.

Why use node.js with DB?

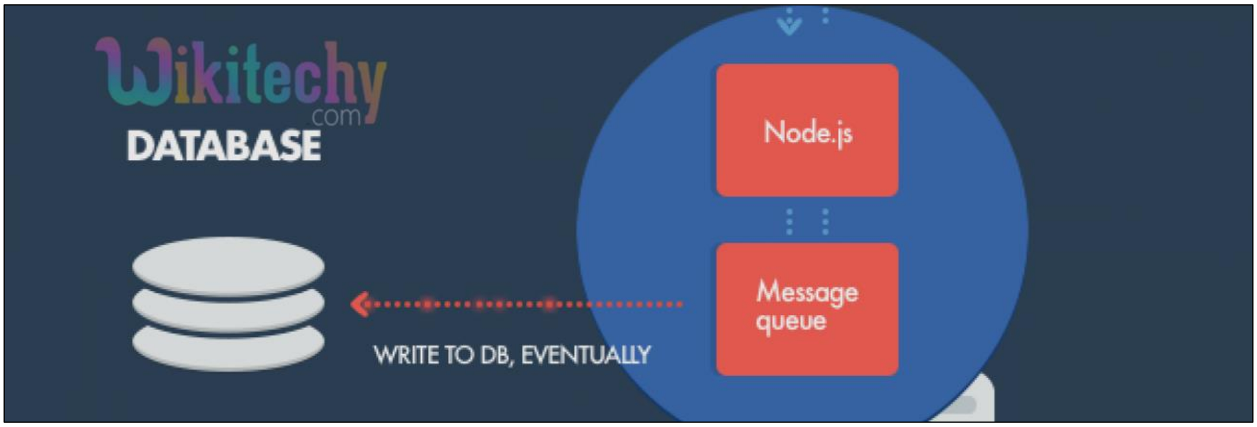

Fig2. Node.js with DB

- Multiple data updates to a database can create chaos in normal synchronized processes.

- As in fig2 node.js manages these simultaneous connections via message queues.

- This approach is generally useful when there are concurrent batch uploads to database, that are collected for use in later stage.

- Eg. User Log are collected and uploaded to database which are used later for report preparations. Such data uploads don't require immediate acknowledgements and hence they are handled by node.js event driven queues.