Node Modules: Callbacks and Error Handling

Jogesh K. Muppala

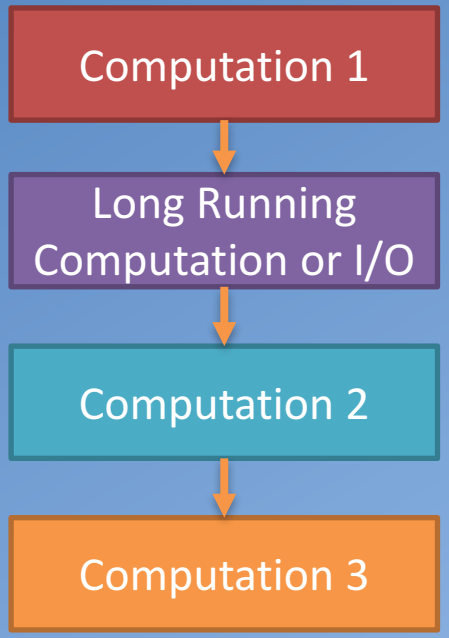
THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

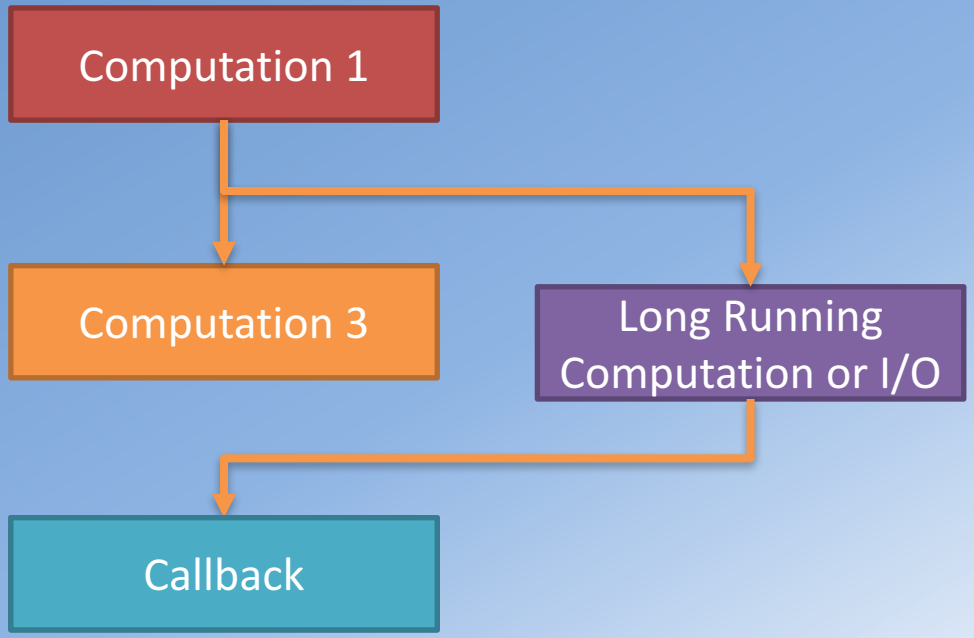
Two Salient Features of JavaScript

- First-class functions: A function can be treated the same way as any other variable
- Closures:
  - A function defined inside another function has access to all the variables declared in the outer function (outer scope)
  - The inner function will continue to have access to the variables from the outer scope even after the outer function has returned

Asynchronous Programming

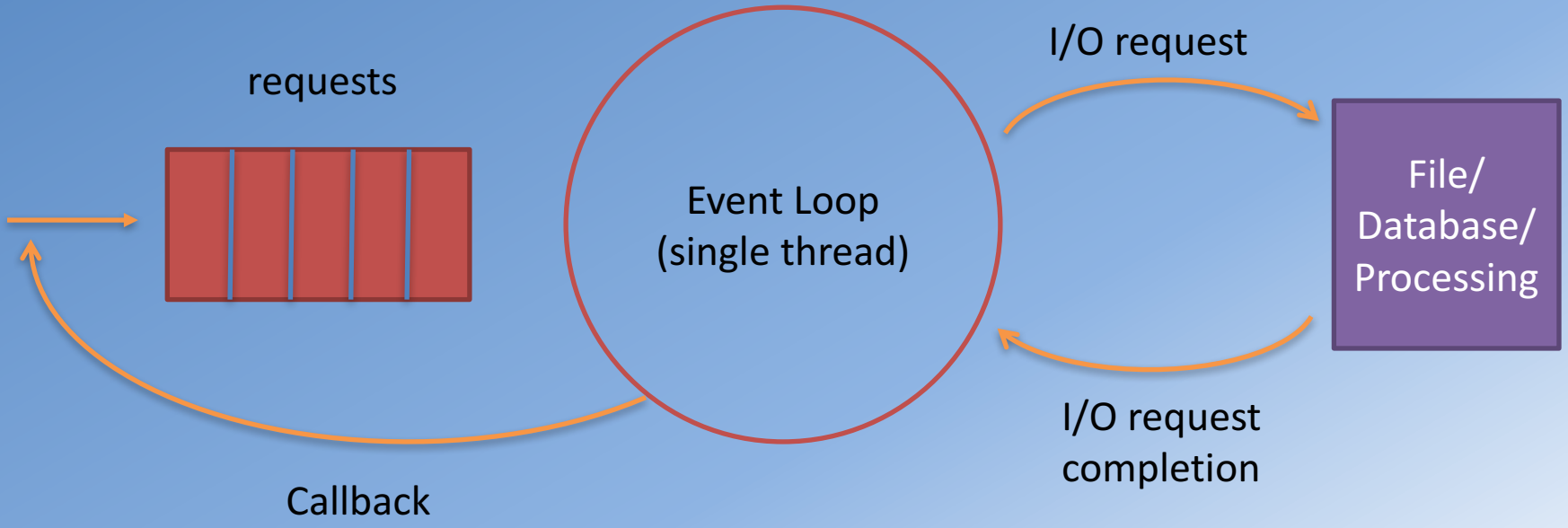


Synchronous Programming

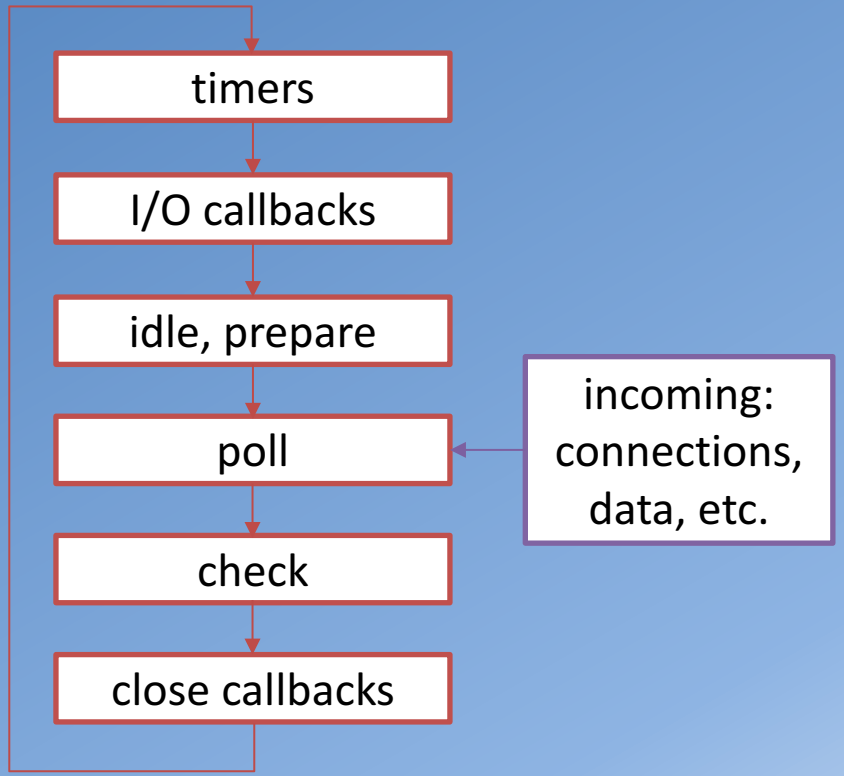


Asynchronous Programming

Node, Async I/O and Callbacks



Event Loop



- **timers**: this phase executes callbacks scheduled by setTimeout() and setInterval().
- **I/O callbacks**: executes almost all callbacks with the exception of close callbacks, the ones scheduled by timers, and setImmediate().
- **idle, prepare**: only used internally.
- **poll**: retrieve new I/O events; node will block here when appropriate.
- **check**: setImmediate() callbacks are invoked here.
- **close callbacks**: e.g. socket.on('close', ...).