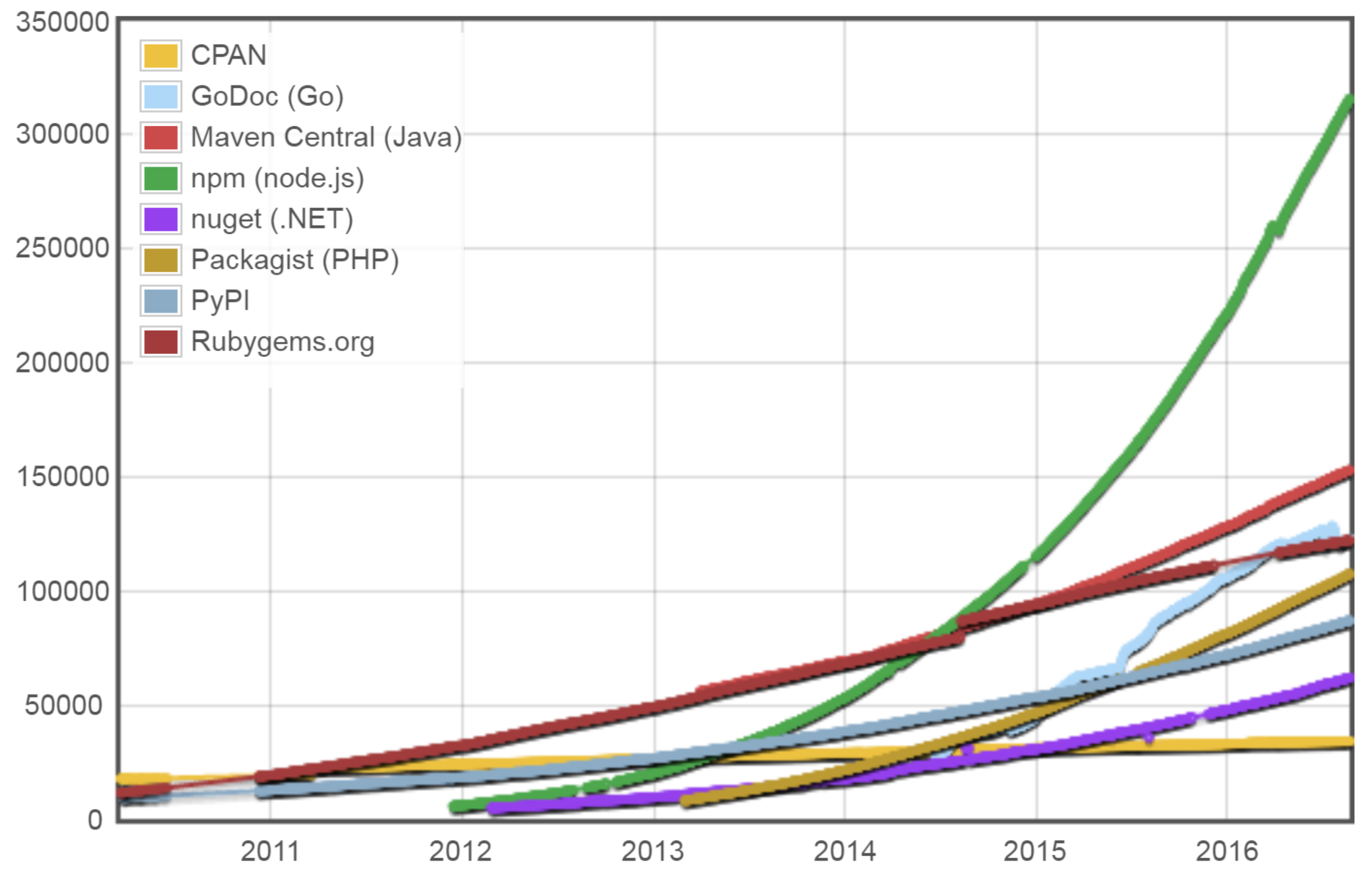
Optimizing and Integrating Node.js for the Cloud

Panagiotis Patros[1], Maria Patrou[1], Maxim Uzun[1], Jiapeng Zhu[1], Kenneth B. Kent[1], Michael Dawson[2]
University of New Brunswick, Faculty of Computer Science[1]
IBM Canada[2]
{Patros.Panos, Maria.Patrou, muzun, jiapeng.zhu, ken}@unb.ca,
Michael_Dawson@ca.ibm.com

Why Node.js?

Developers love Node.js: 315k modules on npmjs.com **3x** growth rate vs other runtimes / languages



Aug 22, 2016
http://www.modulecounts.com

Why the growth?

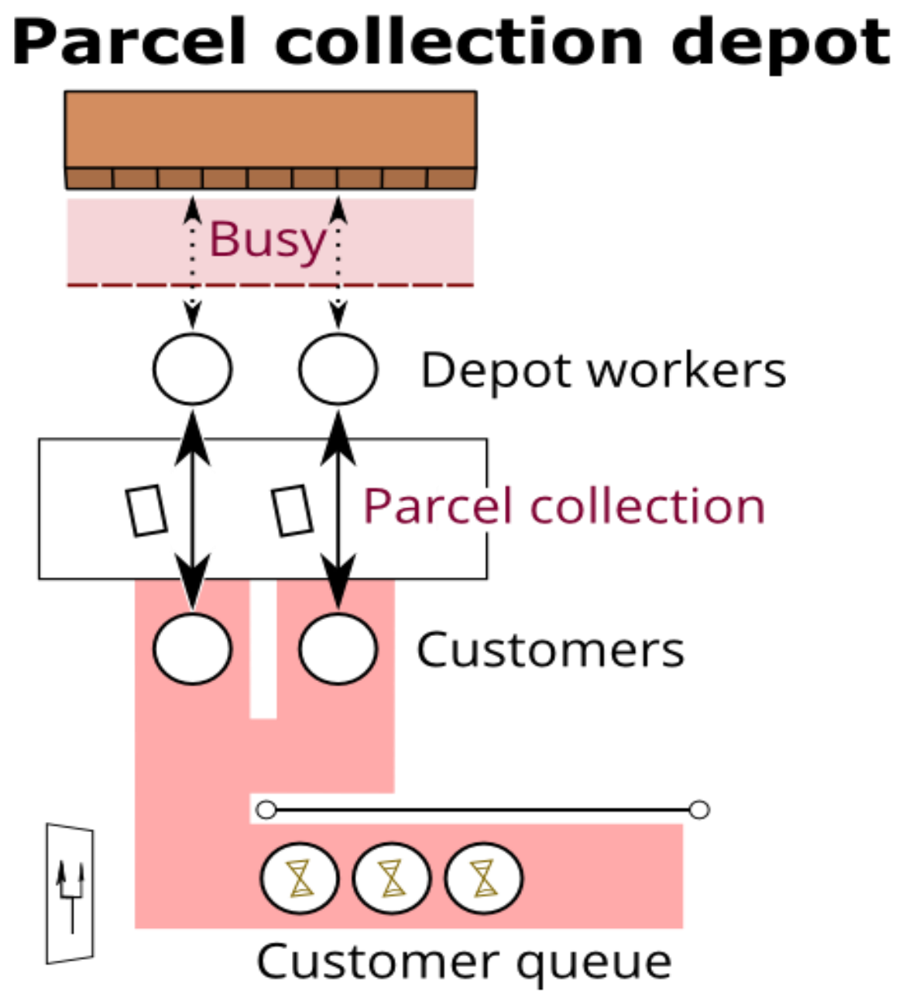
- It is JavaScript (Server Side)
  - Availability of JavaScript talent
  - Same language on client + server side
- End-to-End JavaScript stack
  - Greater productivity and integration with JSON APIs
- Event-driven, single-threaded model
  - Eliminates concurrency and thread-safety worries
- Thousands of concurrent connections with minimal overhead

Where Node.js?

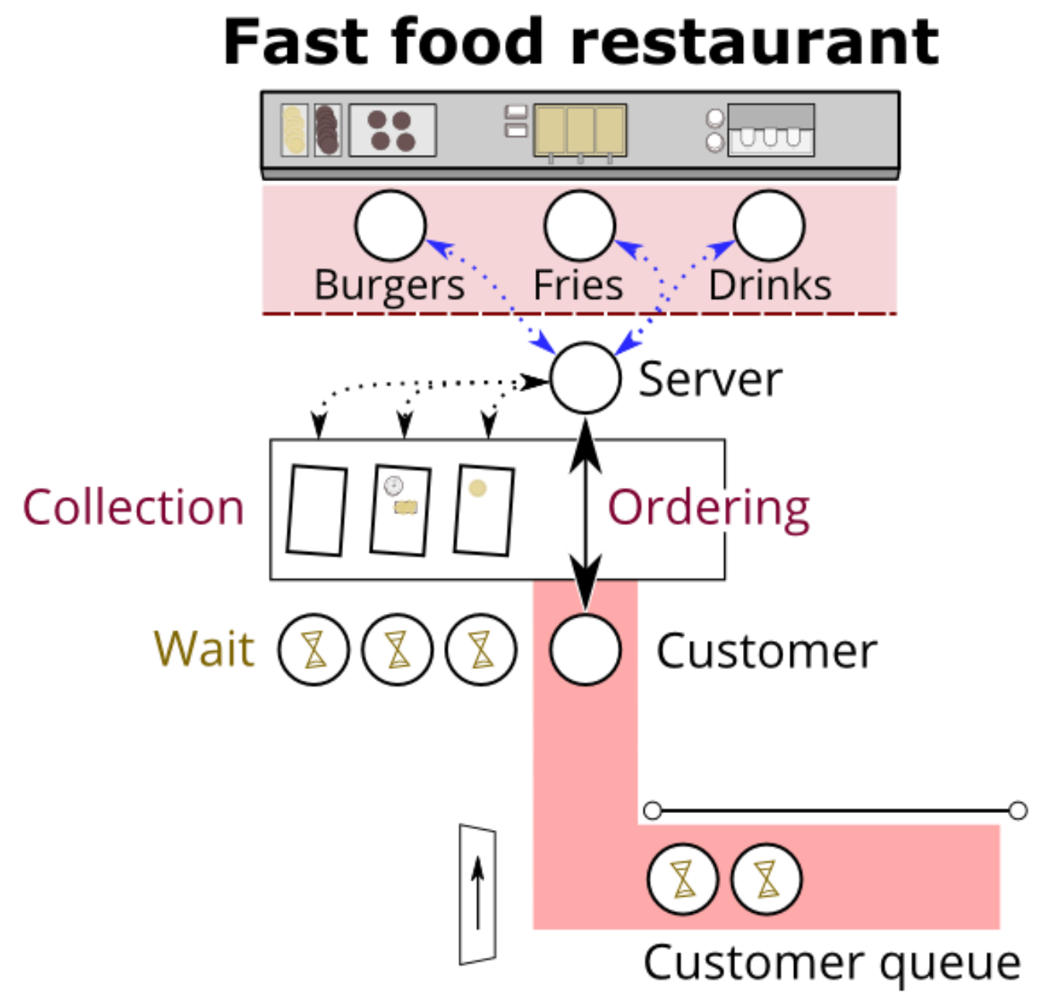
Good fit for highly scalable web applications

Inherently event based: perfect fit for asynchronous non-blocking I/O

Blocking I/O Example (e.g. Java)



Asynchronous I/O Example (Node.js)



Platform as-a-Service (PaaS) Clouds

- Abstract large parts of the software and hardware stack
  - Run user code and provide external services (e.g. DBs)
- Usually on isolated Linux containers
  - Potentially multiple containers on the same host
- Install environment and code
  - Language runtime, Application server, User code
- Scaling
  - Vertical (More Resources) and Horizontal (More Instances)

What is missing in Node.js?

- No multithreading
- Startup optimizations to speed scale out
- Hard to develop and debug live on the cloud
- Incomplete support for various new hardware features
- Needs to be maintained for new/emerging language features

Proposed Research Areas

- Develop Node.js scalability-oriented benchmarks
  - Identify areas for improving the scalability of Node.js
  - Develop Node.js best-practices, modules and core updates towards better scalability
- Investigate deployment and external-service communication features missing from Node.js
  - Propose and evaluate improvements
- Investigate live-development tools and features missing from Node.js
  - Develop solutions
- Investigate hardware features Node.js does not utilize
  - Develop Node.js improvements for hardware support
- Investigate Node.js extended language features
  - Incorporate extended language features into Node.js

The Project

Research Program at UNB with IBM Canada support

- Funding through CAS and NSERC
- Duration 3 years (2016-2019)
- Researchers
  - 2 PhD and 2 MCS students
    - *Talk to us and apply for grad school at UNB CS, if interested!*
    - *http://www.unb.ca/admissions/applying-to-unb/*
  - 3 Undergrad Summer students
  - UNB Faculty and IBM personnel

UNB EST. 1785

IBM Centre for Advanced Studies - Atlantic

FACULTY OF COMPUTER SCIENCE