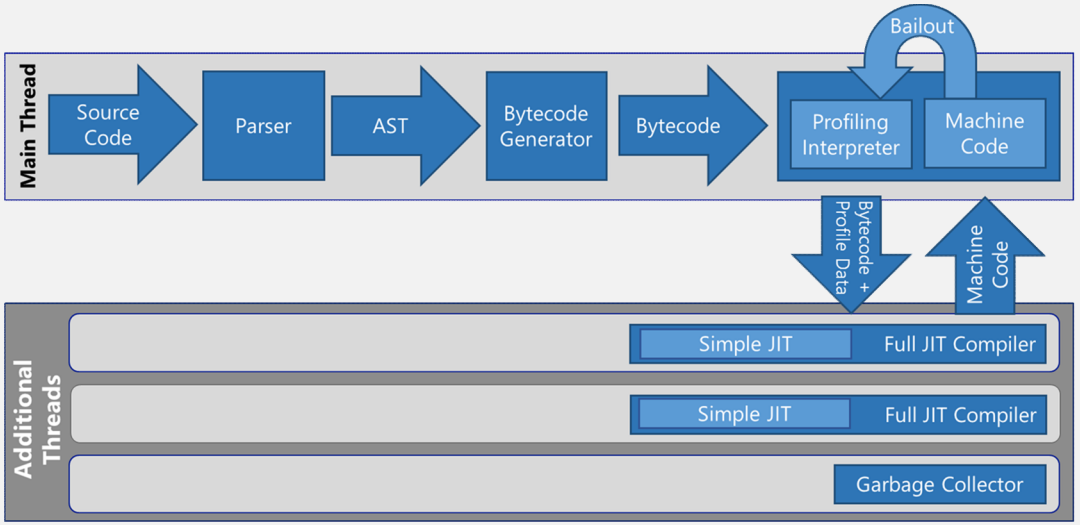
# ChakraCore

Gaurav Seth
@GauravSeth
Principal PM Manager
Chakra & TypeScript

Microsoft

Microsoft & Chakra  JavaScript

# JavaScript




Interpreter for fast start-up & portability

Multi-threaded multi-phased JIT for throughput

Concurrent & partial collector

2008

2015

# JavaScript - Beyond the Browser

## Store based applications, Server applications, Services, IoT…


node.js

Microsoft Azure
DocumentDB

Hi, I'm Cortana.

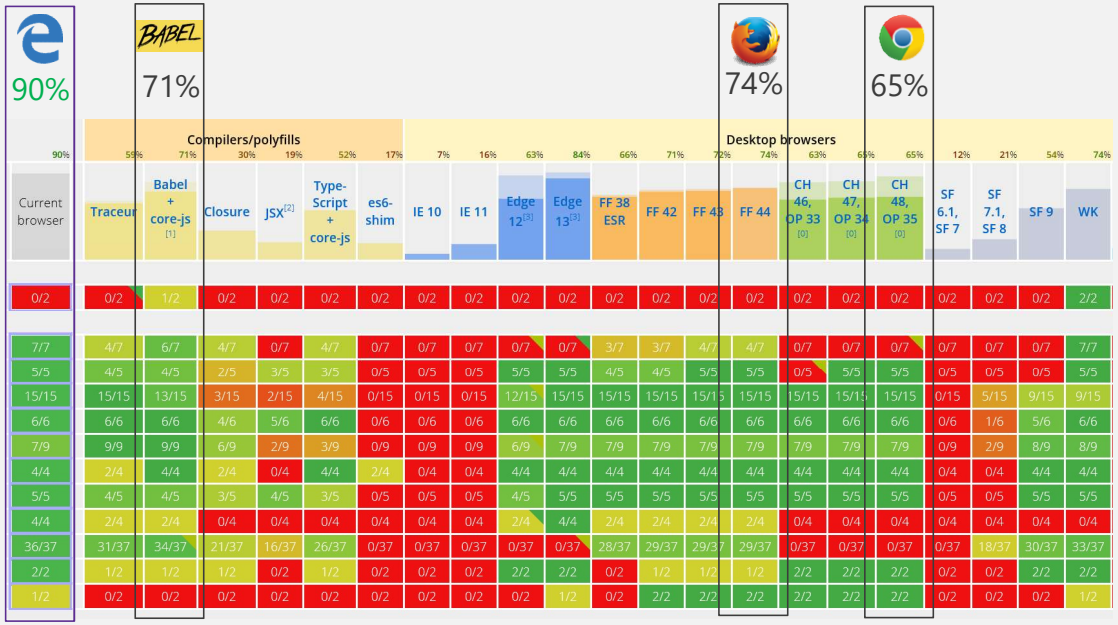
Outlook.com

# Language Landscape

Push the boundaries of interoperable language landscape to enable developers to create great apps

# Language Support

## ECMAScript 6




aka.ms/ES6CompatChart

## And beyond...

Async Functions, Exponentiation operator etc.

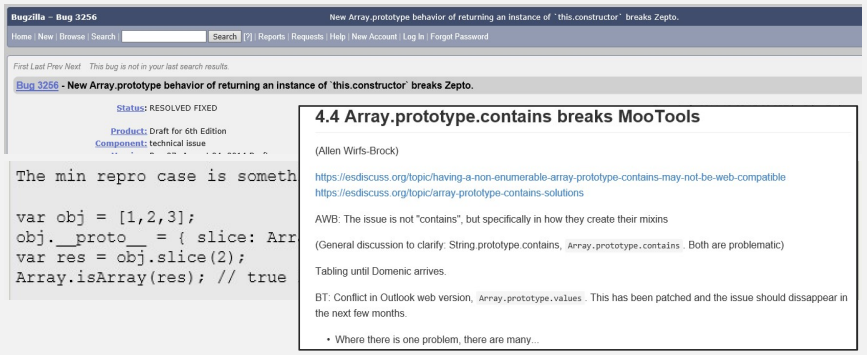
## Pushing Interop

Active member of ECMA TC39 committee

- Editor for ES2016
- Editor for test262

Leading language implementation

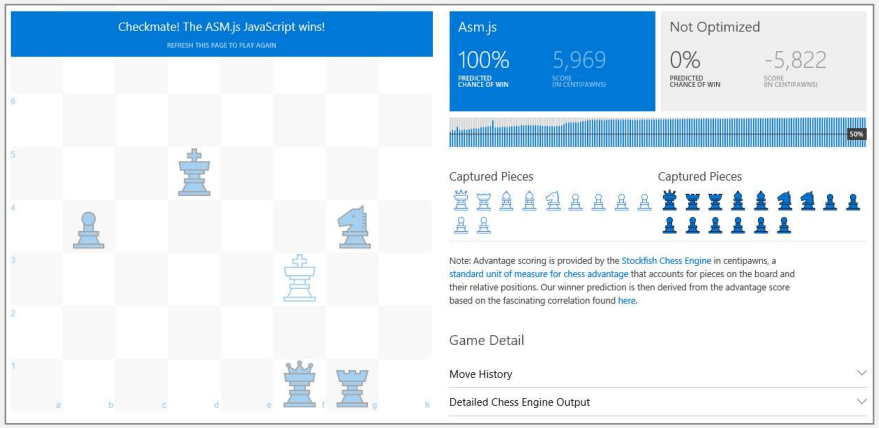
- Challenging for engineering
- Data driven spec evolution
- Sharing breaking change data with the committee/other implementers to evolve ECMAScript



# ASM.js

## Predictable near native performance for the web platform

- In partnership with Mozilla
- Optimizable, low-level subset of JavaScript
- Leveraged principally by transpiling C/C++ code to run on the Web platform
- AOT compilation provides predictable performance
- Potential to provide speed benefits to many more scenarios over a period of time

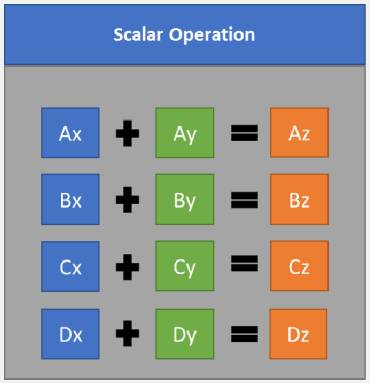



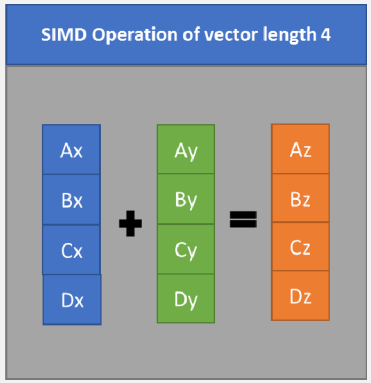

[aka.ms/asmjsChess](aka.ms/asmjsChess)

# SIMD: Single Instruction, Multiple Data (ES2016)

- "In Development" in partnership with Intel
- Exploits data level parallelism using hardware capabilities
- Allows same operations to be performed on multiple data points simultaneously

Four add operations without SIMD

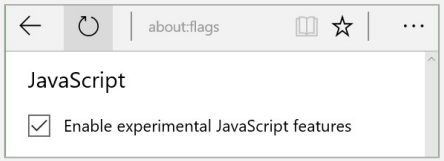



Single add operation with SIMD

- Large number of existing processors support SIMD operations
- Benefits applications in fields like games, 3D graphics, audio/video processing, multimedia applications etc.

[aka.ms/JumpingBirds](aka.ms/JumpingBirds)

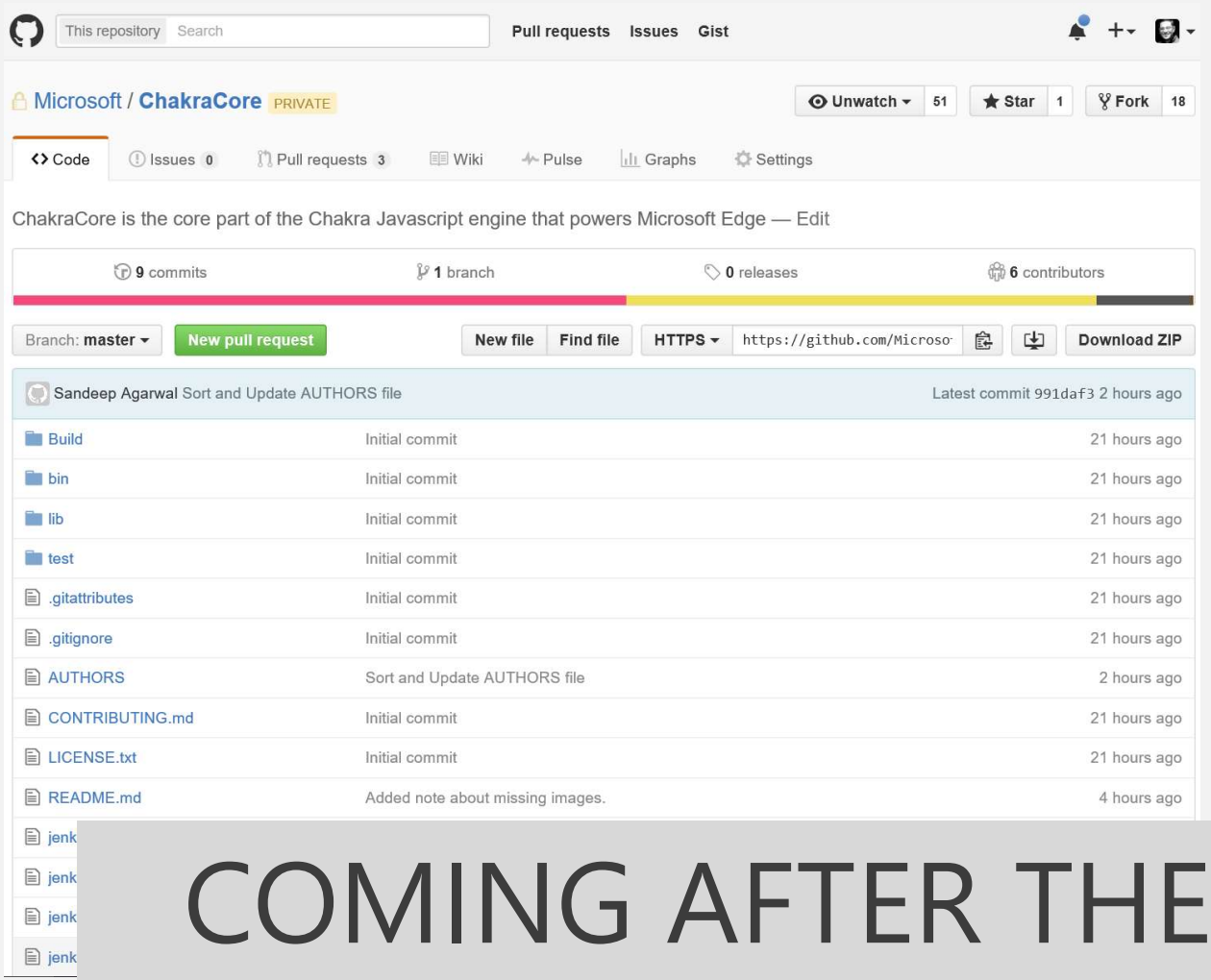


# Next step in our Journey

Chakra is going

OPEN SOURCE

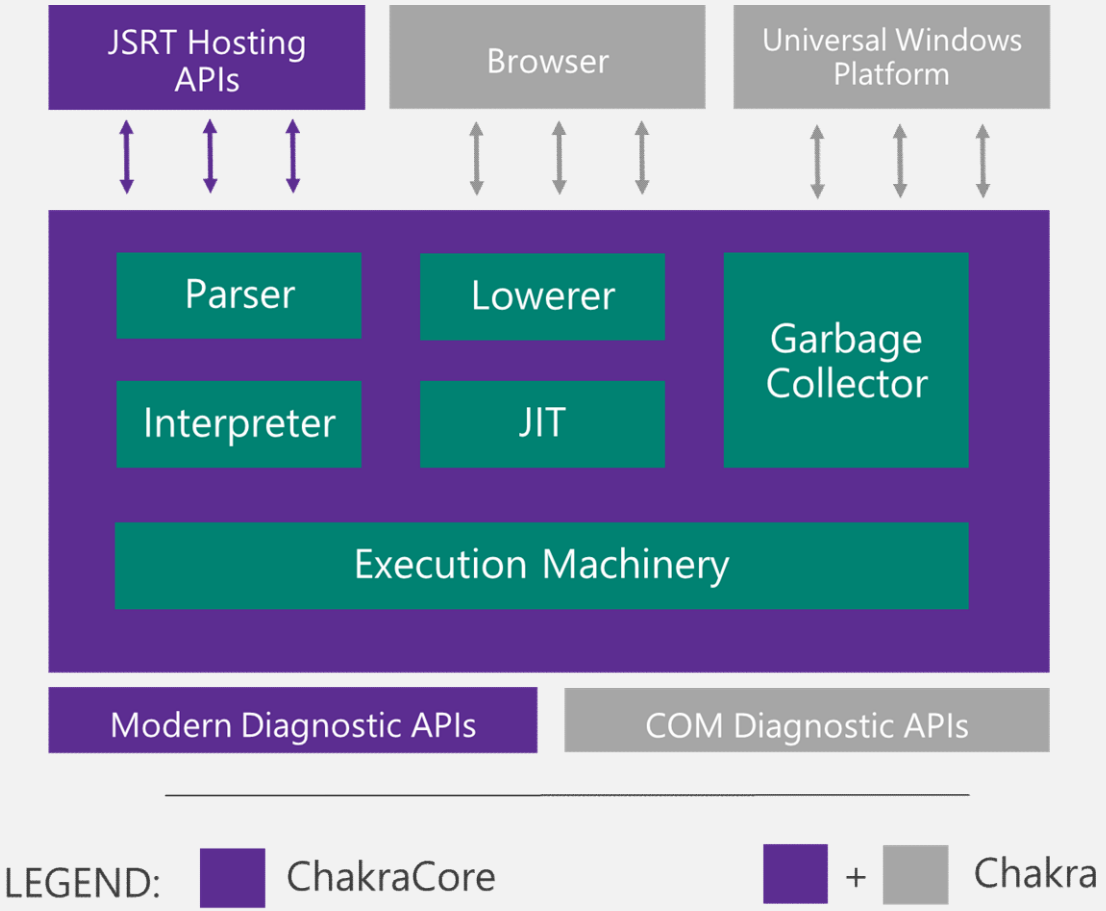
# ChakraCore – Open Source *core* of Chakra

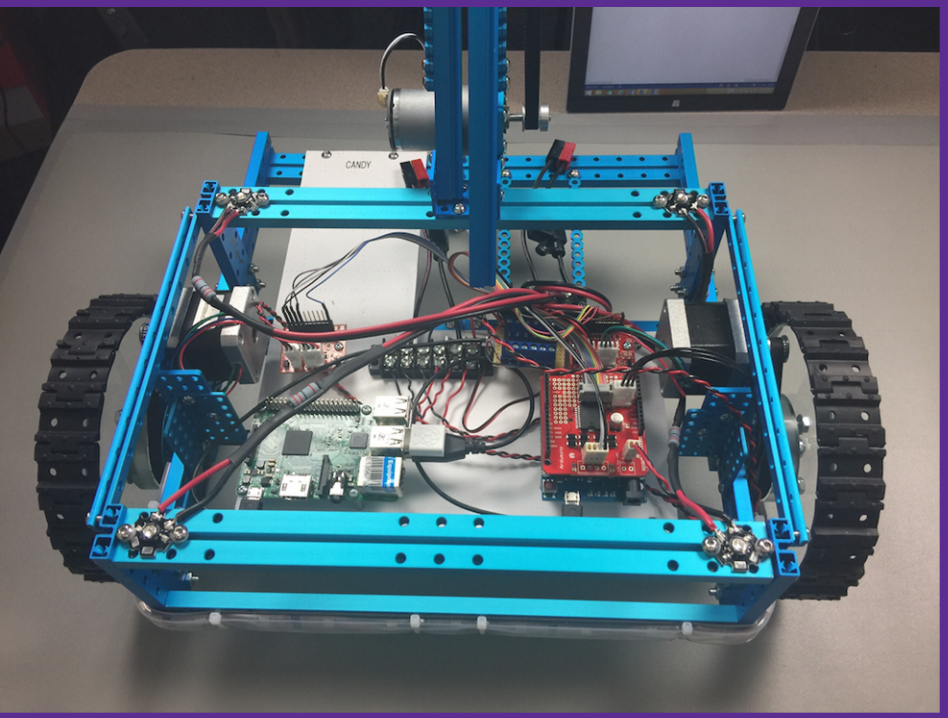



- Browse the sources
- Submit pull requests
- Help us find issues
- Embed it in your apps
- Innovate on top

**COMING AFTER THE HOLIDAYS**

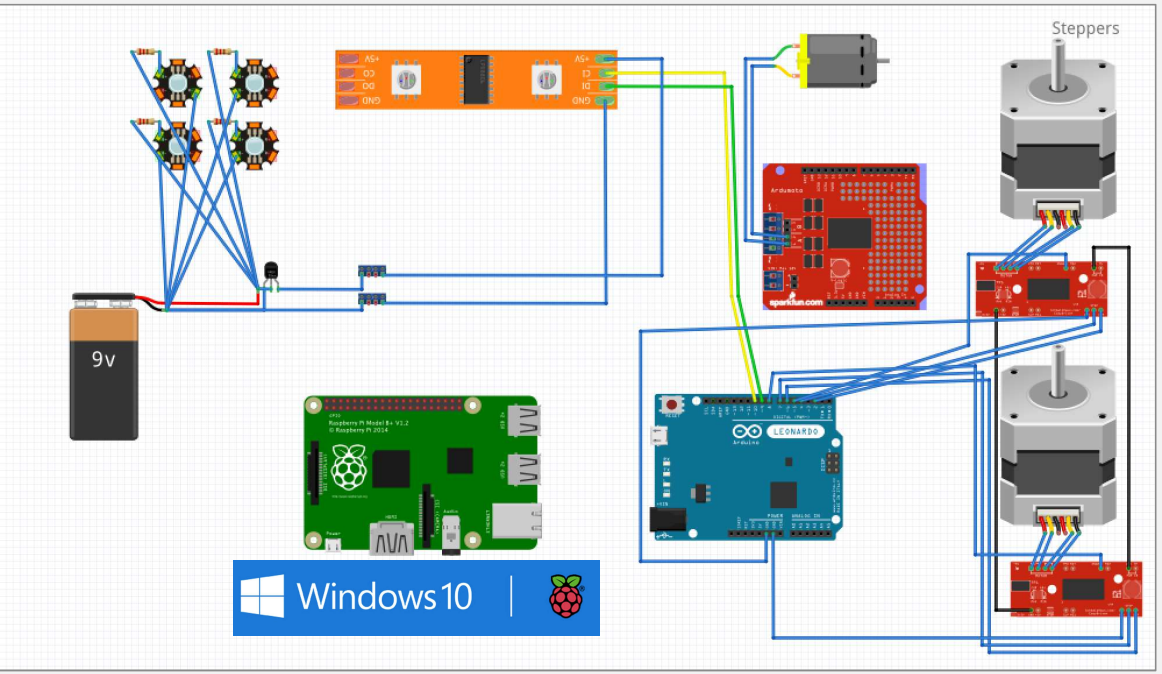
# ChakraCore


JSRT Hosting APIs
Browser
Universal Windows Platform
Parser
Lowerer
Garbage Collector
Interpreter
JIT
Execution Machinery
Modern Diagnostic APIs
COM Diagnostic APIs
LEGEND:
ChakraCore
+
Chakra




# Demos

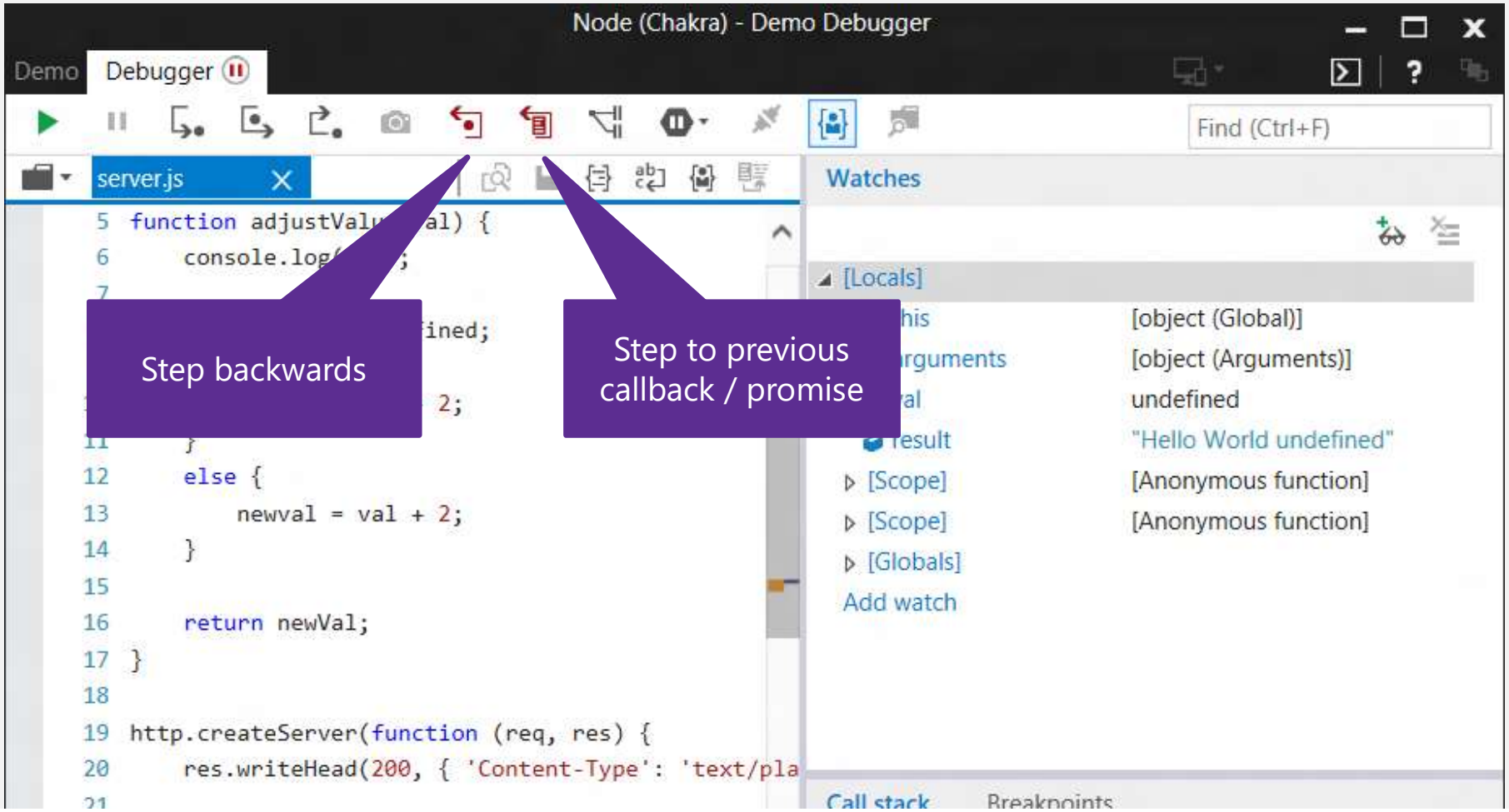
# Candy Robot




- Raspberry Pi 2
- Windows 10 IoT Core
- Node.js (with ChakraCore)

- 100+ npm modules including cylon, serialport, firmata & express

# Time Travel Debugging – Peek into the future



Mikeal Rogers,
Node Foundation

# Performance

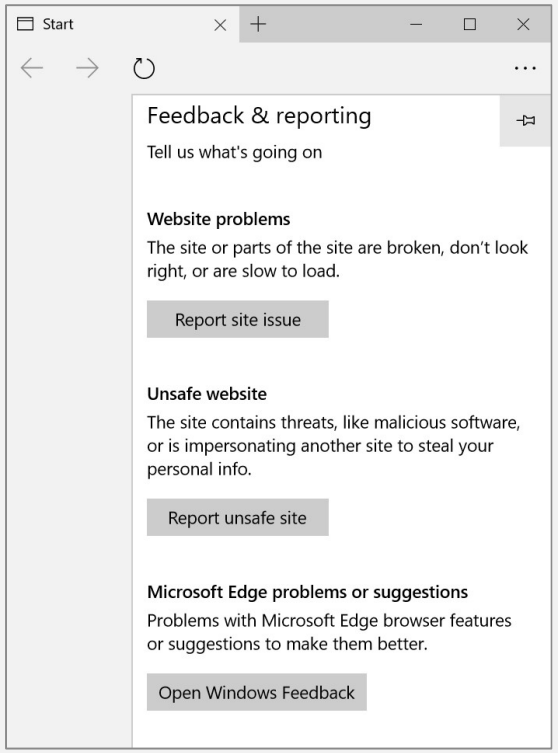
Enable developers to deliver consistently delightful user experiences across all form factors

# Listening to Feedback

## Customer Feedback: Edge executes YouTube html5player.js script slow on (a few sites)

## Stack walking improvements for real world sites

- Stack walk to identify the caller
- Relying on the OS APIs to do virtual unwind
- Data rich but costly + Not all data from OS APIs needed
- Now: JIT'ed functions linked via RBP chains; much faster stack walk



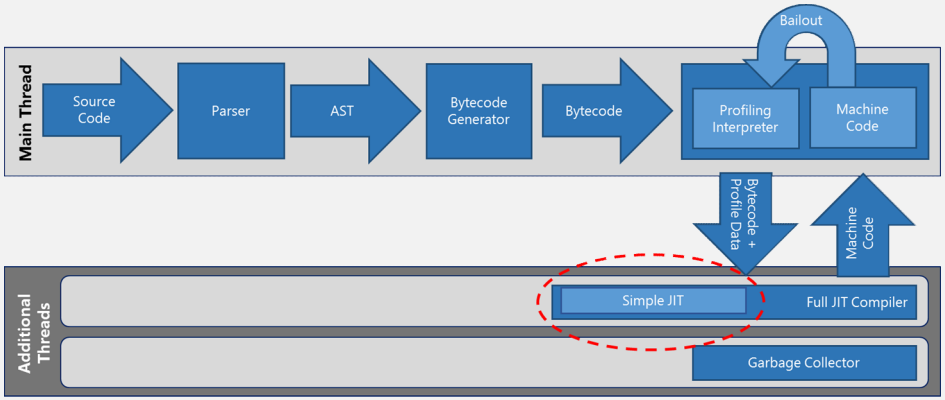

## Wins: 40-50% better page load times of specific sites

# Always optimizing the engine

## Startup, throughput, scalability, using the full power of the hardware

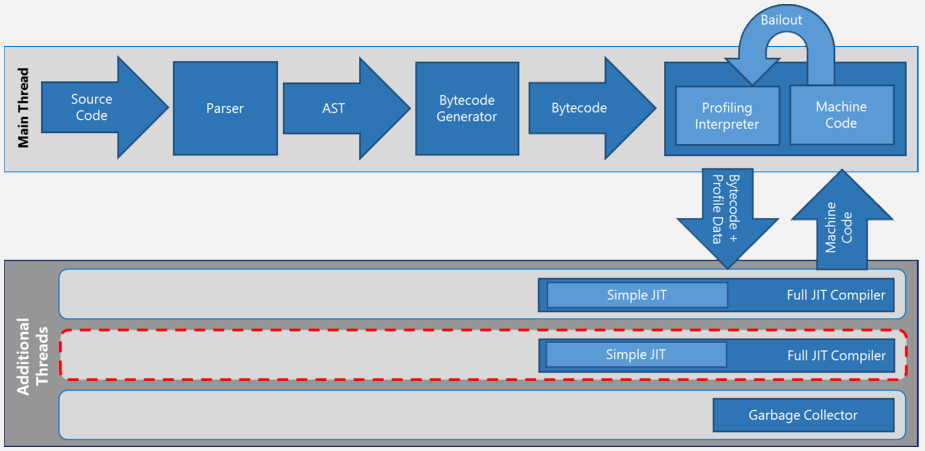
### "Simple JIT" Tier

- New non-optimizing JIT tier
- Increases code throughput by faster transition to compiled code




### Multiple Concurrent JITs

- Large functions block the JIT compiler
- Ability to spawns multiple JIT threads based on hardware profiles




## Wins: 30% improvement in TypeScript compiler's throughput

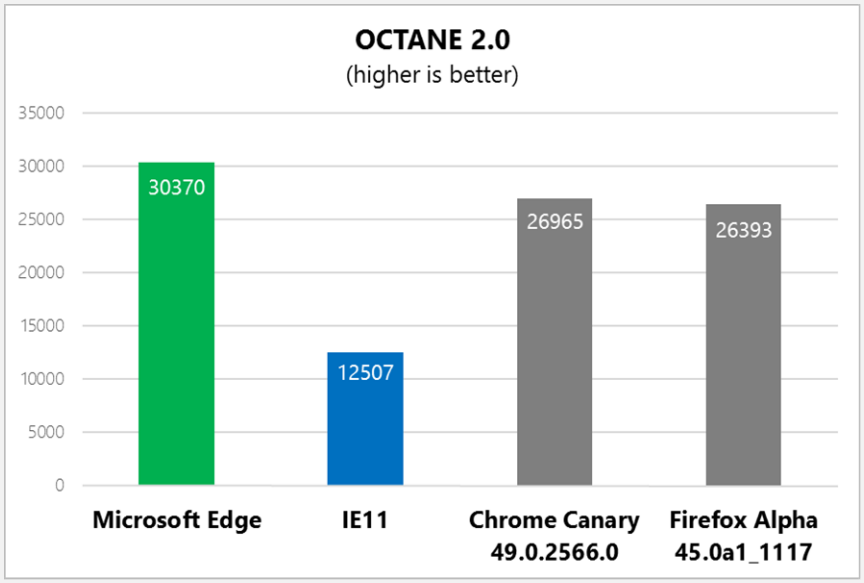
# Staying lean across a range of scenarios
Cortana

## Bailout Memory Reduction

- Cortana runs across a range of device profiles
- Memory footprint important for optimized user experiences
- Optimized JITs maintain bailout info to continue execution when type assumptions go wrong
- E.g. JS heavy payload in browser - 15MB JIT code to 50MB of bailout info/local offsets
- Optimized bailout info and local offsets – shared & tightly packed

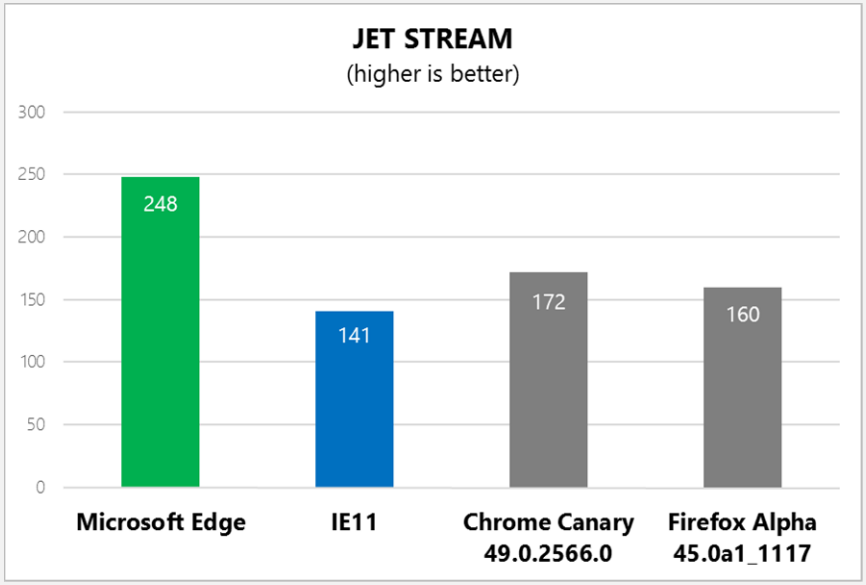
## Wins: Down from 50MB to 18MB – 1/3rd memory reduction

# But what about benchmarks?




Owned by: Google

Edge leads closest competitor by 10+%




Owned by: Apple

Edge leads closest competitor by 40+%

*System info: 64-bit browsers on Intel Core i5-34755 @ 2.90Ghz with 4.0GB RAM running Windows 10*

# It's a Journey: Road ahead

## ChakraCore – sources coming post holidays

PR to Node.js mainline
Interoperable debugging support
Time Travel Debugging
Cross platform support

## Committed to OSS

Engage, develop and learn

## Committed to JavaScript and it's evolution

Make JavaScript powerful – tools, VMs, libs, compilers and more…

# Thank you!

@gauravseth