ECMA SCRIPT

Krishna Mohan Koyya
krishna@glarimy.com | 9731423166
www.glarimy.com

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Schedule

- 1-day class room program with three breaks
  - Lunch break - 45 minutes
  - Refreshment Breaks - 2, each for 15 minutes
- Interspersed with theory and hands-on
  - concepts - 30%
  - demonstrations & exercises - 70%

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Etiquette

- Mute your phones
- Avoid bringing regular work to the class
- No cross discussions
- Stop me for any questions, any time
- Code examples are online, will be shared
  - https://bitbucket.org/glarimy/glarimy-resources/
- Slides are online, will be shared
  - https://bitbucket.org/glarimy/glarimy-resources/
- Attempt all code exercises

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Pre-requisites

- Exposure to basics of programming
  - basics of logic development
  - concepts of object oriented programming
- Exposure to web-ui development
  - basics of HTML
  - basics of CSS
  - basics of Javascript
  - basics of HTTP
- You already know Node and NPM? Wonderful!

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Environment

- Any IDE with Javascript support
  - Visual Studio Code
    - *https://code.visualstudio.com/download*
- Any browser with debugging support
  - Mozilla Firefox, Google Chrome and etc.,
  - Postman or any other REST Client extension
- NodeJS with NPM
  - https://nodejs.org/en/download/
  - Go for LTS

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Krishna Mohan Koyya

- **Technology Consultant**
- **Since 1997 in to the career**
  - With Wipro-Lucent, HP and Cisco till 2005
    - *as engineer and lead*
  - With Sudhari IT Solutions (P) Ltd till 2006
    - *as chief executive officer*
  - With Sasi Institute of Technology & Engineering till 2008
    - *as HoD, Department of Information Technology*
  - With Glarimy Technology Services since 2008
    - *as proprietor and principle consultant*
- **Holds M.Tech (Computer Science & Engineering)**
  - from Andhra University, Visakhapatnam, India
- **Lives in Bengaluru, India**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Your Introduction, please

- **Your name**
- **Your career span, in year**
- **Your regular work**
  - what you do
  - which technology stack you use more often
- **Your expectation**
  - what is your area of focus in the technology
  - how are you going to apply the learning

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Agenda

- Introduction
- Setup
- The Language System
- Logic Development
- Object Oriented Programming
- Modular Programming
- Functional Programming
- Asynchronous Programming
- UI Development
- Event Driven Programming
- AJAX and Beyond

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Introduction

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

History of ECMAScript

- Born as LiveScript
  - At Netscape Communications
  - By Brendan Eich
  - In 1995
- Became JavaScript
  - Because of a pact with Sun Microsystems
  - To counter Microsoft Internet Explorer
- Forgotten for a while
  - Non-standard browsers
  - Cumbersome for larger applications
  - Netscape dead, practically

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

History of ECMAScript

- **Resurrected again**
  - **By the community**
    - *Mozilla Foundation and Firefox Browser*
  - **DOM standardisation**
    - *by W3C, World Wide Web Consortium*
  - **Javascript Object Notation or JSON**
    - *by Douglas Crockford*
  - **Web 2.0**
    - *REST Services and AJAX*
  - **Server-Side adaptation**
    - *By NodeJS*
- **Subjected for multiple attempts**
  - **JScript, Silverlight, ActionScript, TypeScript and etc.,**
- **Standardised**
  - **As ECMA Script**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Versions and Features

- Before ECMA 2009 or ES5
  - Baseline features
- ECMA 2015 or ES6
  - Modules, Promises, Classes and Inheritance
  - Block scopes with let and const
  - Template literals
  - Generator functions
  - Spread operators
  - Arrow functions
- ECMA 2016 or ES7
  - Minor additions
- ECMA 2017 or ES8
  - Async and await
- ECMA 2018 or ES9
  - Minor additions

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Purpose and Applications

- **Started as a simple browser scripting language**
- **Became a full blown language**
  - **For desktop browser applications**
    - *event driven programming*
    - *primarily for DOM events*
    - *UI development with HTML and CSS*
    - *Server communication*
  - **For server applications**
    - *single threaded and event driven async programming*
    - *primarily for request processing*
    - *on platforms like NodeJS*
  - **For mobile/device applications**
    - compiled to native code
  - **For serverless applications**
    - *on platforms like AWS Lambda Functions*

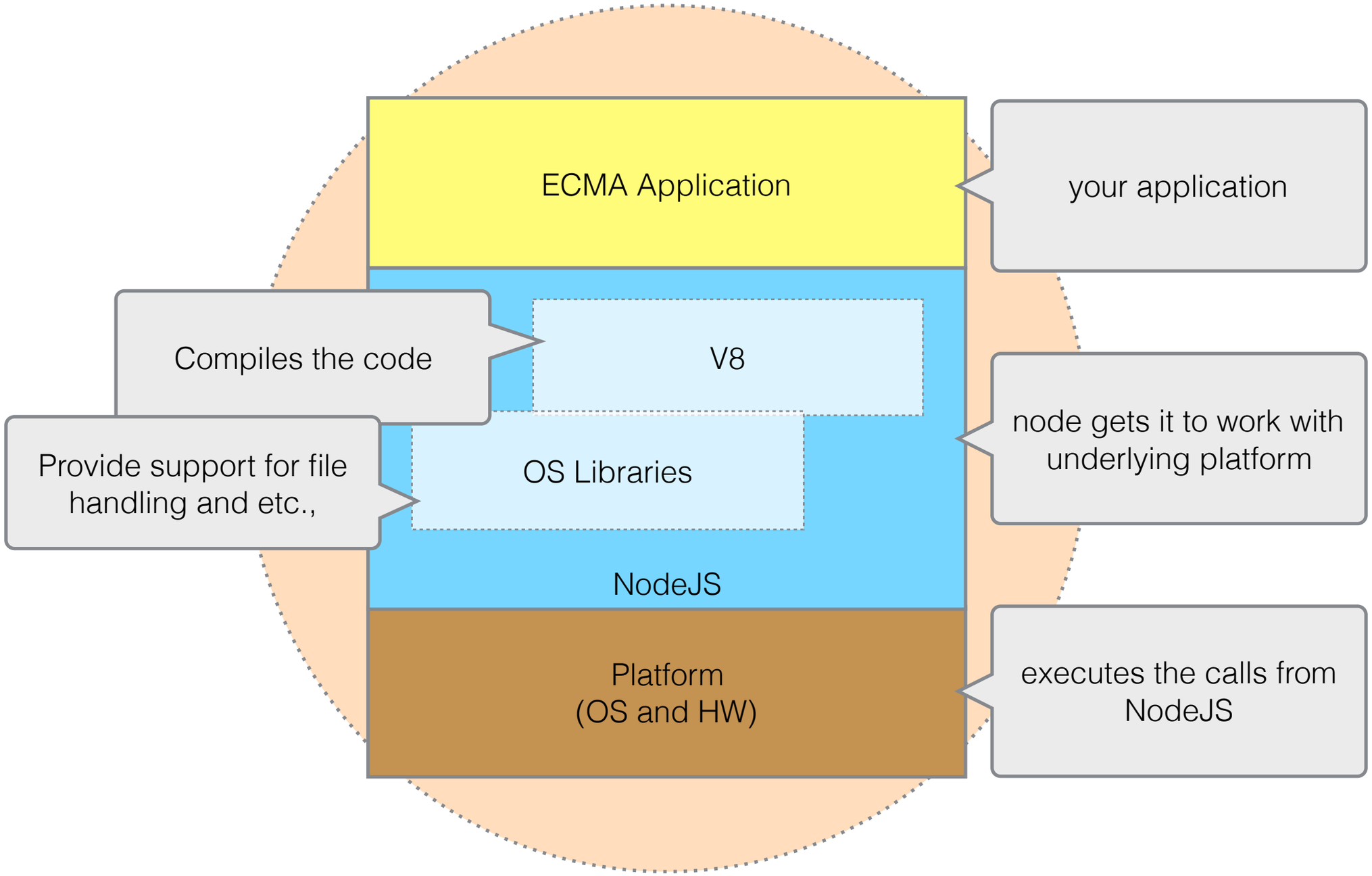
Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Salient Features

- **High-Level Language without full compilation**
  - **Pure Interpretation Engines**
    - *Old browser technology*
    - *Rhino (Mozilla), JScript (Microsoft)*
  - **JIT (Just-In-Time) Compilation Engines**
    - *New browser technology*
    - *V8 (Google), SpiderMonkey (Mozilla), Nashon (Oracle), Chakra (Microsoft)(*
- **Dynamic or Duck-Type Language**
  - **No way to declare types**
    - *type of variables are computed at runtime*
    - *type of variables can change anytime*
- **Object Oriented Language, almost**
  - **Classes, Objects, Inheritance**
- **Single Threaded (Asynchronous Programming Features)**
  - **Events, Promises, Async and Await**
- **Functional Programming Features**
  - **Closures and Arrow-Functions**

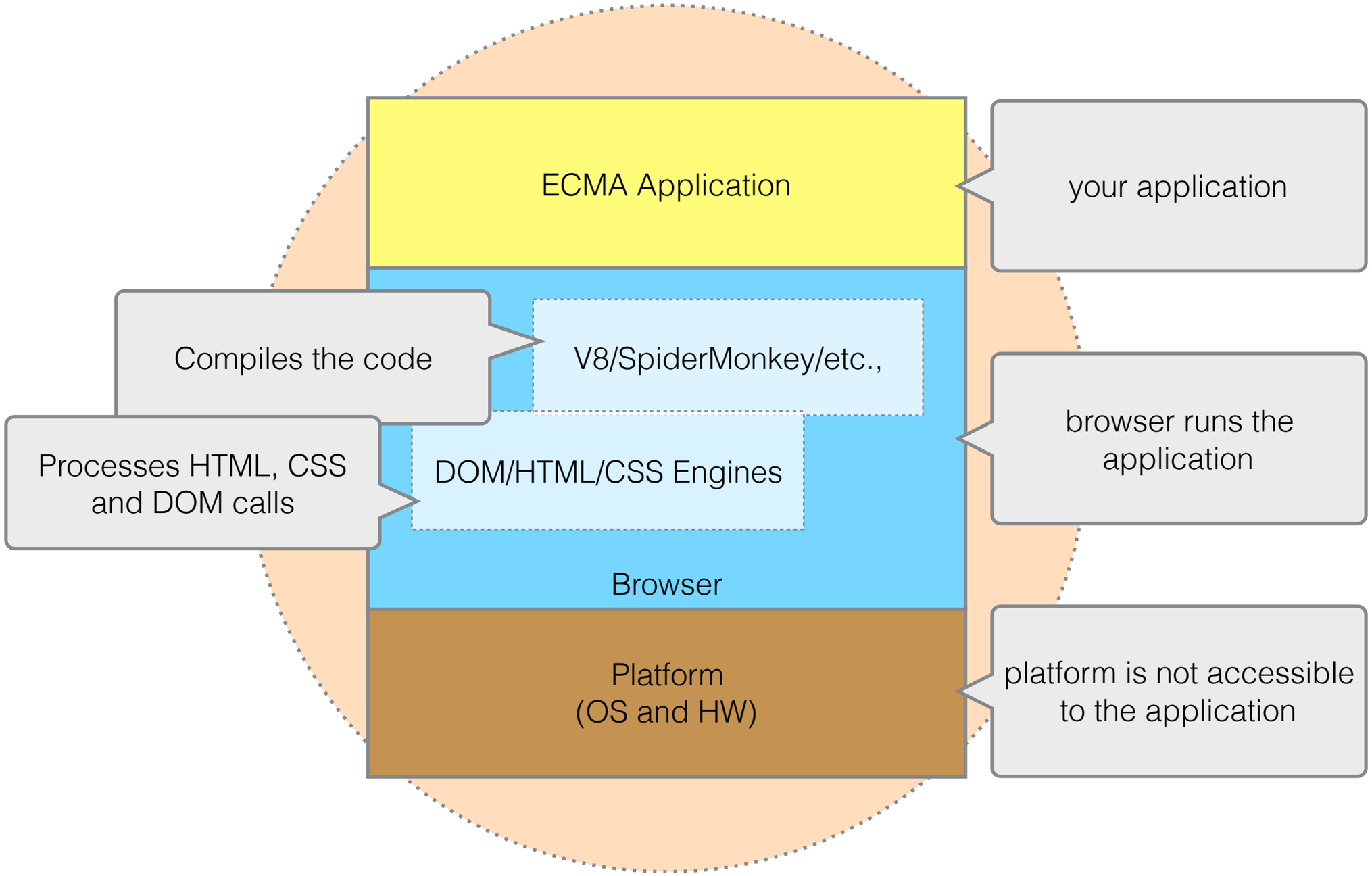
Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Technology Stack on Server



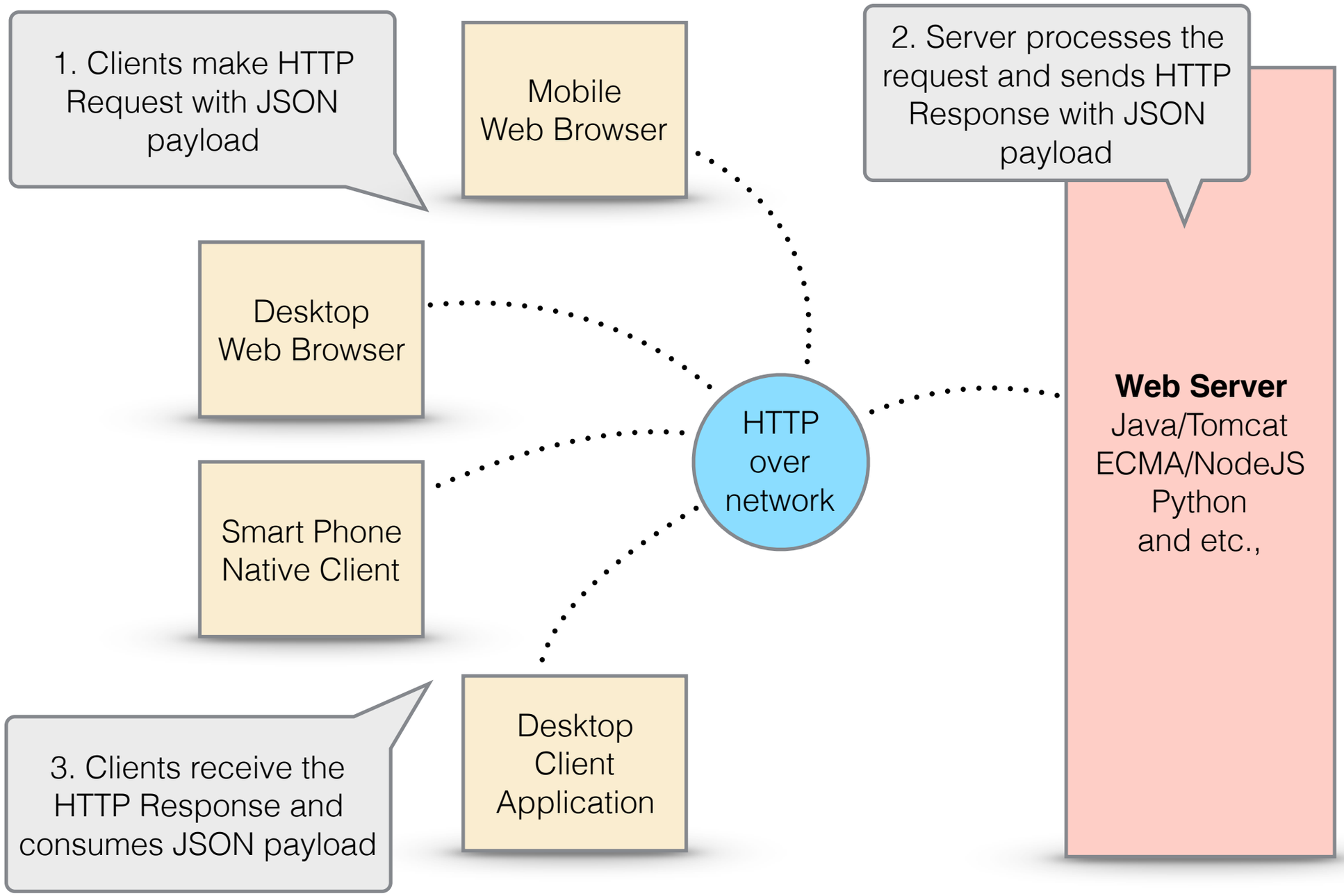
Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Technology Stack on Browser



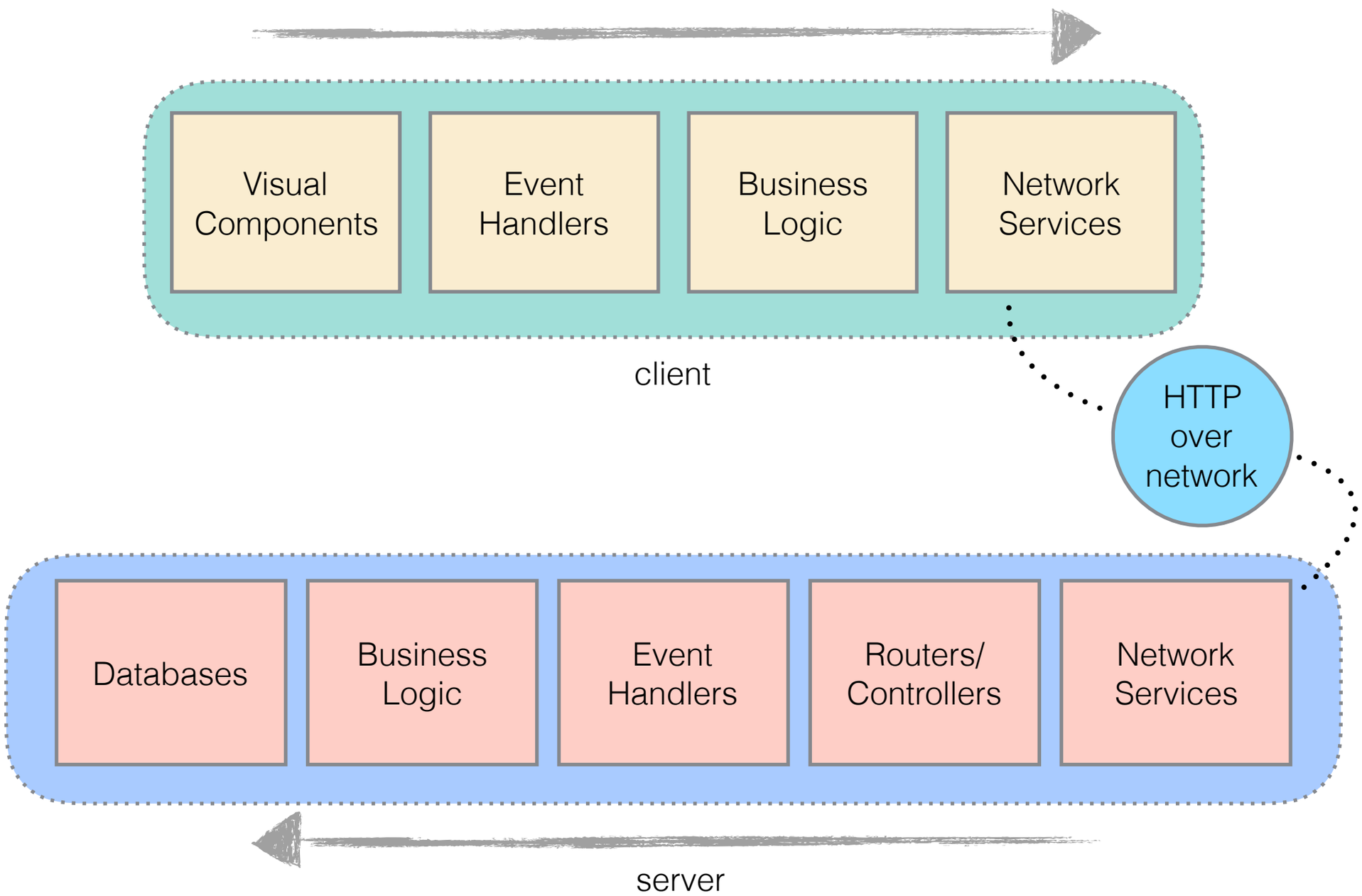
Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Client/Server Architecture



Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Application Architecture



Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Frameworks

- JQuery
  - A simple library to handle browser discrepancies
  - A wrapper on top of Javascript
  - One of the very early and popular libraries
- AngularJS
  - A 2-way binding framework from Google
  - Very popular at it's peak
- ReactJS with Redux
  - A simple component framework from Facebook
  - Tops the current trends
- NG or Angular
  - A re-vamped AngularJS From Google
  - Component framework with Typescript
- Dojo, Sencha ExtJS, GWT
  - Comprehensive frameworks
  - Passed their prime

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Questions

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Quickies

- **Javascript is a strongly typed language**
  - Yes
  - No
- **ECMA 2015 is also called ES6**
  - Yes
  - No
- **Google V8 engine runs on Chrome and NodeJS**
  - Yes
  - No
- **Most of the browsers do not support ES6 fully**
  - Yes
  - No

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Quickies

- **Javascript is a strongly typed language**
  - No, it's a dynamically typed language
- **ECMA 2015 is also called ES6**
  - Yes, it is called because it's 6th version released in 2015
- **Google V8 engine runs on Chrome and NodeJS**
  - Yes, V8 engine is open sourced JIT compiler
- **Most of the browsers do not support ES6 fully**
  - Yes, only ES5 works on most of the current browsers

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Thanks

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Setup

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Setup

- **Install NodeJS**
  - **Update the `path`, if needed!**
- **Open a terminal**
  - **Verify the installation**

    ```
    node -v
    npm -v
    ```
  - **Create a working folder**

    ```
    {/path/to/home}/ecma/labs
    ```
- **Install and Launch Visual Studio Code**
  - **Open integrated terminal**
  - **verify the versions of Node and npm**
  - **Open the working folder**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Hello World

- **Create a file** `hello.js`
  - **under the working folder**
- **Enter the code and save the file**

```
console.log("Hello World!");
```

- **Run the command from the terminal**

```
cd <working-folder>
node hello.js
```

- **Check the results**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Hello World

- **Create and move to folder** `hello`

  ```
  cd {workingfolder}
  mkdir hello
  cd hello
  ```

- **Run the command and answer the questions**

  ```
  npm init
  ```

- **Update** `package.json`
  - **Add the following under scripts**

  ```
  "start": "node index.js"
  ```

- **Run the command from the terminal**

  ```
  npm start
  ```

- **Check the results**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Hello World

- **Create a file** `hello.html`
  - **under the working folder**
- **Enter the code and save the file**

```
<html>
 <head>
  <script>
   console.log("Hello World!");
  </script>
 </head>
</html>
```

- **Open the file in a browser**
- **Verify the results in the browser console**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Hello World

- **Create a file** `hello.html`
  - **under the working folder**
- **Enter the code and save the file**

```
<html>
  <head>
    <script src="hello.js"></script>
  </head>
</html>
```

- **Open the file in a browser**
- **Verify the results in the browser console**

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Questions

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Quickies

- **The command `npm start` runs the `start` script in the `package.json`**
  - Yes
  - No
- **The function `console.log()` always writes on the server terminal**
  - Yes
  - No
- **The HTML code runs also on NodeJS**
  - Yes
  - No
- **The `npm` tool helps in creating, downloading, installing and running packages on NodeJS**
  - Yes
  - No

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Quickies

- **The command `npm start` runs the `start` under `scripts` in the `package.json`**
  - Yes, it's a short cut to run different scripts in a package
- **The function `console.log()` always writes on the server terminal**
  - No, it writes on the terminal of the current environment in which it is running
- **The HTML code runs also on NodeJS**
  - No, only javascript code runs on NodeJS
- **The `npm` tool helps in creating, downloading, installing and running packages on NodeJS**
  - Yes, the tool also handles dependency management

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Thanks

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

The Language System

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Object Oriented Programming

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Modular Programming

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Functional Programming

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Asynchronous Programming

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

UI Development

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

Event Driven Programming

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

AJAX and Beyond

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

ECMA SCRIPT

AJAX and Beyond

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.

Thanks

Copyright (C) 2019 Glarimy Technology Services. All rights reserved.