Everything a Java EE Developer needs to know about the JavaScript Landscape

leanpub.com/
everythingjs

Everything a Java EE Developer needs to know about the JavaScript Landscape

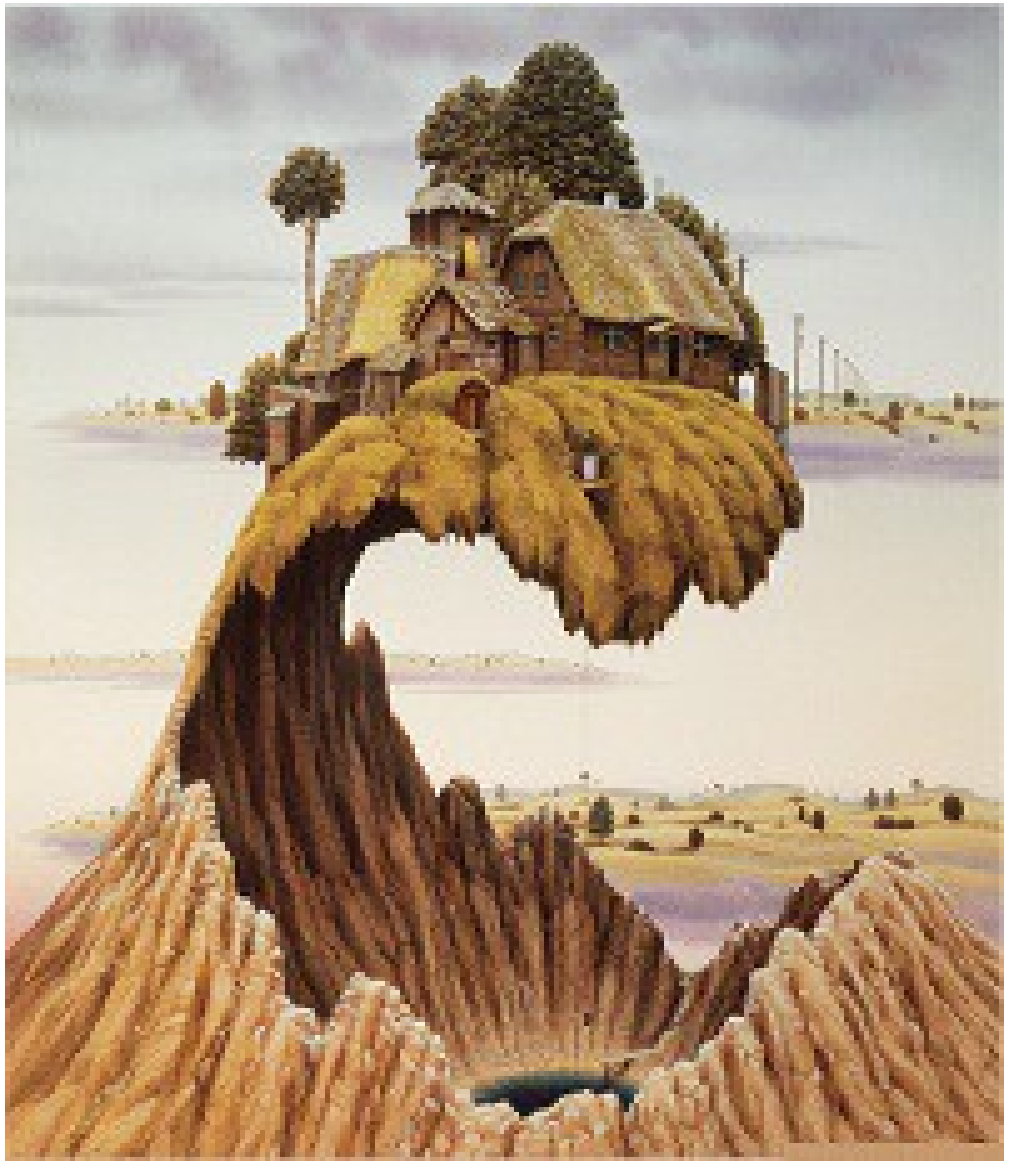
Geertjan Wielenga
@geertjanw

leanpub.com/
everythingjs

Everything a Java Developer needs to know about the JavaScript Landscape

Geertjan Wielenga
@geertjanw

leanpub.com/
everythingjs



leanpub.com/
everythingjs

Program Agenda

- JavaScript is the assembly language of the web
- Many, many, many libraries, frameworks, and tools
  - Core libraries
  - Module systems
  - Build tools
  - Application frameworks
  - Testing frameworks
  - Component suites
  - Supporting tools
- Maybe we should forget about Java altogether?

leanpub.com/
everythingjs

Program Agenda

- JavaScript is the assembly language of the web
- Many, many, many libraries, frameworks, and tools
  - Core libraries
  - Module systems
  - Build tools
  - Application frameworks
  - Testing frameworks
  - Component suites
  - Supporting tools
- Maybe we should forget about Java altogether?

leanpub.com/
everythingjs

JavaScript is the assembly language of the web.

leanpub.com/
everythingjs

```
; Example of IBM PC assembly language
; Accepts a number in register AX;
; subtracts 32 if it is in the range 97-122;
; otherwise leaves it unchanged.

SUB32  PROC          ; procedure begins here
       CMP   AX,97   ; compare AX to 97
       JL    DONE    ; if less, jump to DONE
       CMP   AX,122  ; compare AX to 122
       JG    DONE    ; if greater, jump to DONE
       SUB   AX,32   ; subtract 32 from AX
DONE:  RET           ; return to main program
SUB32  ENDP          ; procedure ends here
```

FIGURE 17. Assembly language

```
simple_loop:
# parameter 1: %rdi
..B1.1:                         # Preds ..B1.0
..___tag_value_simple_loop.1:   #2.1
        xorl      %eax, %eax    #3.19
        xorl      %edx, %edx    #5.8
        testq     %rdi, %rdi    #5.16
        jle       ..B1.5        # Prob 10% #5.16
                   #LOE rax rdx rbx rbp rdi r12 r13 r14 r15
..B1.3:                         # Preds ..B1.1 ..B1.3
        addq      %rdx, %rax    #6.5
        addq      $1, %rdx      #5.19
        cmpq      %rdi, %rdx    #5.16
        jl        ..B1.3        # Prob 82%   #5.16
..B1.5:                         # Preds ..B1.3 ..B1.1
        ret                     #8.10
        .align    2,0x90
```

An assembly language is a **low-level programming language** for a computer, or other programmable device, in which there is a very strong (generally one-to-one) correspondence between the language and the architecture's machine code instructions. Each assembly language is **specific to a particular computer architecture**, in contrast to most high-level programming languages, which are generally portable across multiple architectures, but require interpreting or compiling.

Wikipedia

machine architecture: **browser**

machine language: **JavaScript**

Netscape – JavaScript

- In 1995, JavaScript started at Netscape
- Java applets were cute – jumped on the bandwagon
- License from Sun Microsystems to use “Java”
- Uses a vaguely similar syntax
- In reality, nothing like Java
- Added JavaScript to the browser

“JavaScript is the assembly language of the web.”

– Scott Hanselman, 2011

leanpub.com/
everythingjs

“I said 'JS is the x86 of the web' a couple of years ago but I can't claim it's original.

The point is JS is about as low as we can go. But it also has higher level facilities.”

– Brendan Eich (inventor of JavaScript)

```
; Example of IBM PC assembly language
; Accepts a number in register AX;
; subtracts 32 if it is in the range 97-122;
; otherwise leaves it unchanged.

SUB32  PROC         ; procedure begins here
       CMP  AX,97   ; compare AX to 97
       JL   DONE    ; if less, jump to DONE
       CMP  AX,122  ; compare AX to 122
       JG   DONE    ; if greater, jump to DONE
       SUB  AX,32   ; subtract 32 from AX
DONE:  RET          ; return to main program
SUB32  ENDP         ; procedure ends here
```

FIGURE 17. Assembly language

```
simple_loop:
# parameter 1: %rdi
..B1.1:                         # Preds ..B1.0
..___tag_value_simple_loop.1:   #2.1
        xorl      %eax, %eax    #3.19
        xorl      %edx, %edx    #5.8
        testq     %rdi, %rdi    #5.16
        jle       ..B1.5        # Prob 10% #5.16
                  #LOE rax rdx rbx rbp rdi r12 r13 r14 r15
..B1.3:                         # Preds ..B1.1 ..B1.3
        addq      %rdx, %rax    #6.5
        addq      $1, %rdx      #5.19
        cmpq      %rdi, %rdx    #5.16
        jl        ..B1.3        # Prob 82%   #5.16
..B1.5:                         # Preds ..B1.3 ..B1.1
        ret                     #8.10
        .align    2,0x90
```

to act like they are the only company in the world?Edit <button id="po-z12hsf3b5s2xej3nw23vdlsazlrls5eyk04" name="eswidget" href="javascript:void(0);" class="esw eswd a-f-i-sb-e" g:entity="buzz:z12hsf3b5s2xej3nw23vdlsazlrls5eyk04" g:type="plusone" aria-pressed="false" title="Click to +1 this post"></button> - Comment - Share +16 by Laurent Bugnion and 15 others 2 shares - Behnam Yousefi and Felix M. Martinez 26 older comments from Larry Ruckman, Adrian Clark, Rafał Legiędź, Mendelt Siebenga, Scott Hanselman, mehrdad tat and John Hostile <img src="https://lh5.googleusercontent.com/-1JLP6eQk7mk/AAAAAAAAAAI/AAAAAAAAAAA/fbqGdArgmik/photo.jpg?sz=32" alt="Ernest Stormann's profile photo" class="a-f-i-q a-b-f-i-q a-f-i-W-Zb-z"> Ernest Stormann - Competition makes everyone better, but Darwin never said you wouldn't be tired.   <button id="po-z12hsf3b5s2xej3nw23vdlsazlrls5eyk04#1309992166066000" name="eswidget" href="javascript:void(0);" class="esw eswd a-f-W-sb-e" g:entity="comment:z12hsf3b5s2xej3nw23vdlsazlrls5eyk04#1309992166066000" g:type="plusone" title="Click to +1 this comment"></button> Add a comment... <h3 class="a-za">Scott Hanselman</h3><img src="//lh5.googleusercontent.com/-73Eb7tLaAEM/AAAAAAAAAAI/AAAAAAAAAG4/-J_WZn_VXQs/photo.jpg?sz=48" alt="Scott Hanselman's profile photo" class="a-f-i-q a-b-f-i-q"> Scott Hanselman -   - Mobile - Public I hope the g+ iPhone app is decent because this Safari nonsense, while technically impressive, is a jittery mess. HTML5 my ass. <button id="po-z12tdltjdrreslamw23vdlsazlrls5eyk04" name="eswidget" href="javascript:void(0);" class="esw eswd a-f-i-sb-e" g:entity="buzz:z12tdltjdrreslamw23vdlsazlrls5eyk04" g:type="plusone" aria-pressed="false" title="Click to +1 this post"></button> - Comment - Share +14 by Robert Scoble, John Sheehan, Laurent Bugnion and 11 others 1 share - Gianluca Gravina 11 older comments from Jonathan Sampson, Daniel Treadwell, M. Edward (Ed) Borasky, Michael Neale, Scott Hanselman and Rob

leanpub.com/
everythingjs

JavaScript is machine language for the browser.

leanpub.com/
everythingjs

Program Agenda

- JavaScript is the assembly language of the web
- Many, many, many libraries, frameworks, and tools
  - Core libraries
  - Module systems
  - Build tools
  - Application frameworks
  - Testing frameworks
  - Component suites
  - Supporting tools
- Maybe we should forget about Java altogether?

leanpub.com/
everythingjs

Core Libraries

leanpub.com/
everythingjs

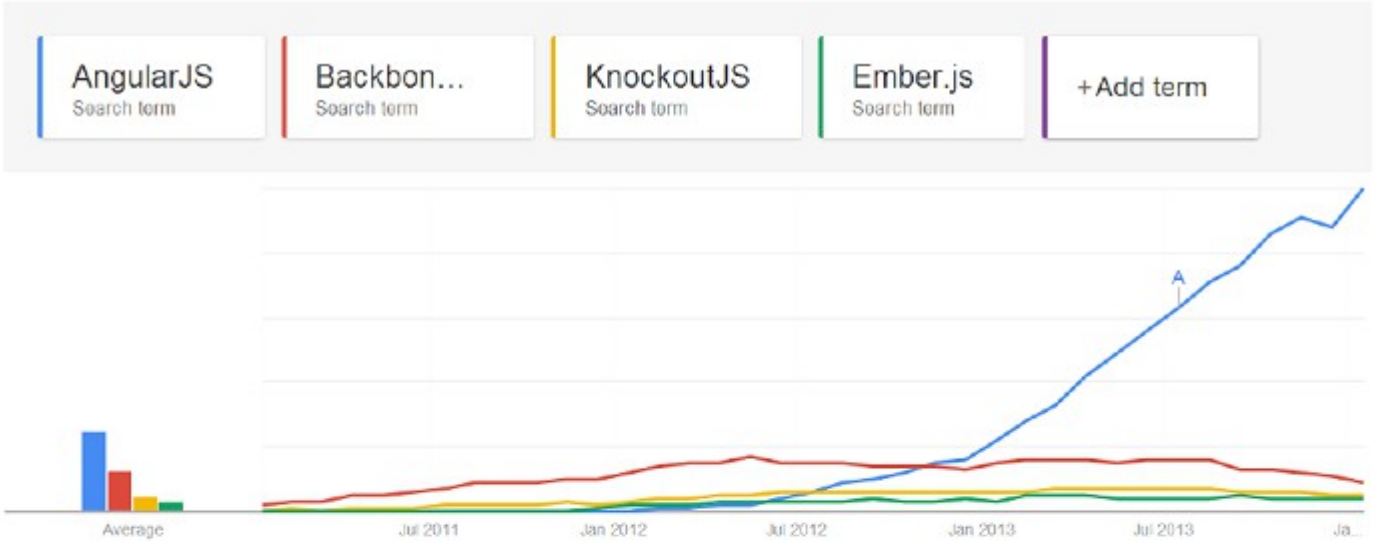
Module Systems

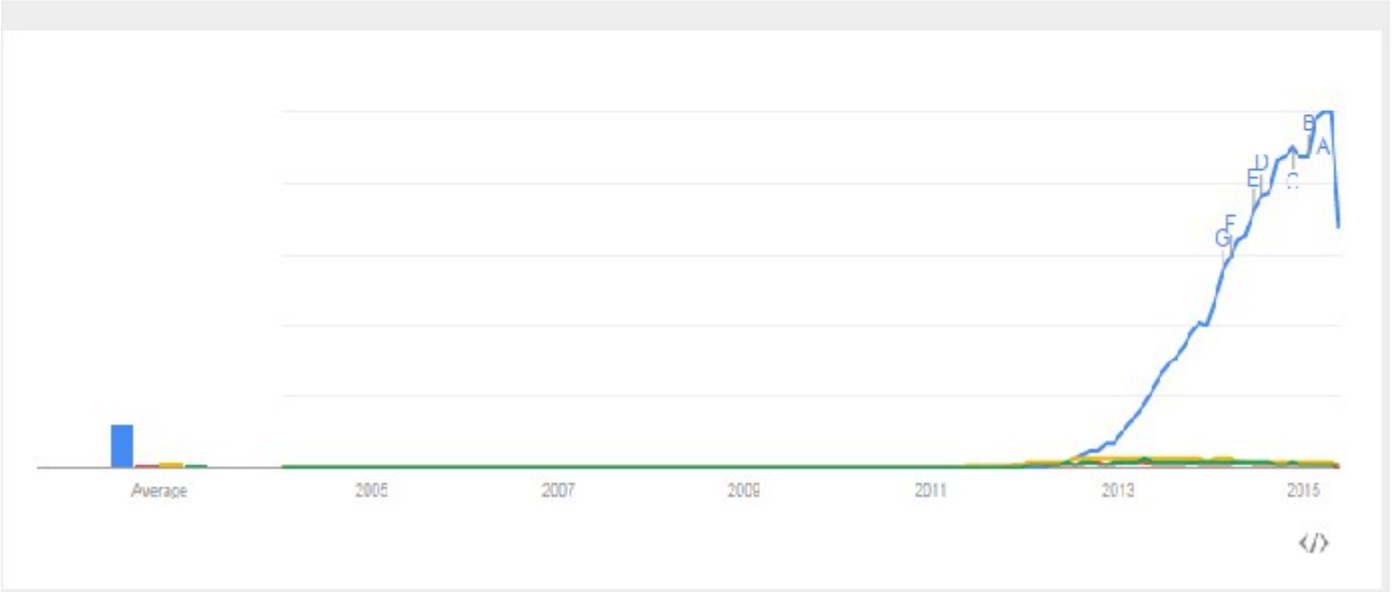
Build Tools

leanpub.com/
everythingjs

Application Frameworks

leanpub.com/
everythingjs





leanpub.com/
everythingjs

Enterprises on AngularJS

http://blog.backand.com/are-enterprises-migrating-to-angularjs

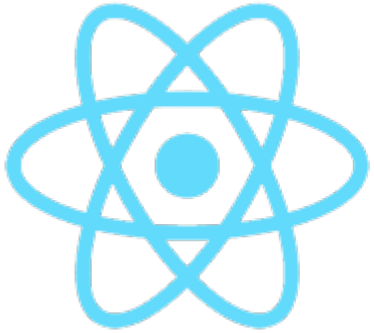
- Google – DoubleClick
- ING – SpectINGular
- IBM – MobileFirst Platform Foundation

Demo

leanpub.com/
everythingjs

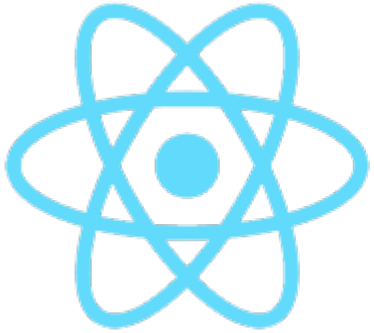
New Kid On The Block...

leanpub.com/
everythingjs



React

leanpub.com/
everythingjs



React

- Om – ClojureScript interface to React
- Mori – Persistent data structures
- Cortex – Centrally manage data
- Mercury – Module system
- Mithril – Similar framework

WONTA

leanpub.com/
everythingjs

~~Write Once, Run Anywhere~~

Write **O**nce, **N**ever **T**ouch **A**gain

leanpub.com/
everythingjs

Testing Frameworks

leanpub.com/
everythingjs

Module Suites

Data grids, menus, sliders, and other widgets.

Supporting Tools

YEOMAN

android

Program Agenda

- JavaScript is the assembly language of the web
- Many, many, many libraries, frameworks, and tools
  - Core libraries
  - Module systems
  - Build tools
  - Application frameworks
  - Testing frameworks
  - Component suites
  - Supporting tools
- Maybe we should forget about Java altogether?

leanpub.com/
everythingjs

Maybe we should forget about Java altogether?

leanpub.com/
everythingjs

Transpilers

Source-to-source compiler

From Wikipedia, the free encyclopedia

A **source-to-source compiler**, **transcompiler**, or **transpiler** is a type of compiler that takes the source code of a programming language as its input and outputs the source code into another programming language. A source-to-source compiler translates between programming languages that operate at approximately the same level of abstraction, while a traditional compiler translates from a higher level programming language to a lower level programming language. For example, a source-to-source compiler may perform a translation of a program from Pascal to C.

https://github.com/jashkenas/coffeescript/wiki/List-of-languages-that-compile-to-JS

Benefits of Transpilers

- Use a “better” (different, more familiar) language
- Make use of different language features
- Integrate with JavaScript landscape
- Same language on frontend and backend

Transpilers

- CoffeeScript
- TypeScript
- Dart
- GWT, Errai, Vaadin
- DukeScript

CoffeeScript

```
# Assignment:
number   = 42
opposite = true

# Conditions:
number = -42 if opposite

# Functions:
square = (x) -> x * x

# Arrays:
list = [1, 2, 3, 4, 5]
```

leanpub.com/
everythingjs

CoffeeScript

```coffeescript
# Objects:
math =
  root:   Math.sqrt
  square: square
  cube:   (x) -> x * square x

# Splats:
race = (winner, runners...) ->
  print winner, runners

# Existence:
alert "I knew it!" if elvis?

# Array comprehensions:
cubes = (math.cube num for num in list)
```

leanpub.com/
everythingjs

CoffeeScript

- Classes, Inheritance, and Super
- Lexical Scoping and Variable Safety
- String Interpolation
- Block Strings and Block Comments
- Golden Rule: “It's Just JavaScript”

leanpub.com/
everythingjs

TypeScript

```
class Student {
    fullname : string;
    constructor(public firstname, public middleinitial, public lastname) {
        this.fullname = firstname + " " + middleinitial + " " + lastname;
    }
}

interface Person {
    firstname: string;
    lastname: string;
}

function greeter(person : Person) {
    return "Hello, " + person.firstname + " " + person.lastname;
}

var user = new Student("Jane", "M.", "User");
```

leanpub.com/
everythingjs

TypeScript

- Superset of JavaScript
- Optionally typed
- Declaration files can apply types to existing libraries
- Type inference
- Modules
- Generics
- Mixins

leanpub.com/
everythingjs

Dart

```dart
import 'dart:math' show Random;         // Import a class from a library.

void main() {                           // The app starts executing here.
  print(new Die(n: 12).roll());         // Print a new object's value. Chain method calls.
}

class Die {                             // Define a class.
  static Random shaker = new Random(); // Define a class variable.
  int sides, value;                     // Define instance variables.

  String toString() => '$value';        // Define a method using shorthand syntax.

  Die({int n: 6}) {                     // Define a constructor.
    if (4 <= n && n <= 20) {
      sides = n;
    } else {
      throw new ArgumentError(/* */);  // Support for errors and exceptions.
    }
  }
  int roll() {                          // Define an instance method.
    return value = shaker.nextInt(sides) + 1; // Get a random number.
  }
}
```

leanpub.com/
everythingjs

Dart

- Optionally typed
- Functions
- Interfaces
- Mixins
- Libraries
- Generics

Dart

- Metadata (annotations)
- Integration with AngularJS
- Integration with Polymer
- Transpiles to JavaScript or the Dart VM
- Run in Dartium, special build of Chromium
- Maintained by Google

GWT, Errai, Vaadin

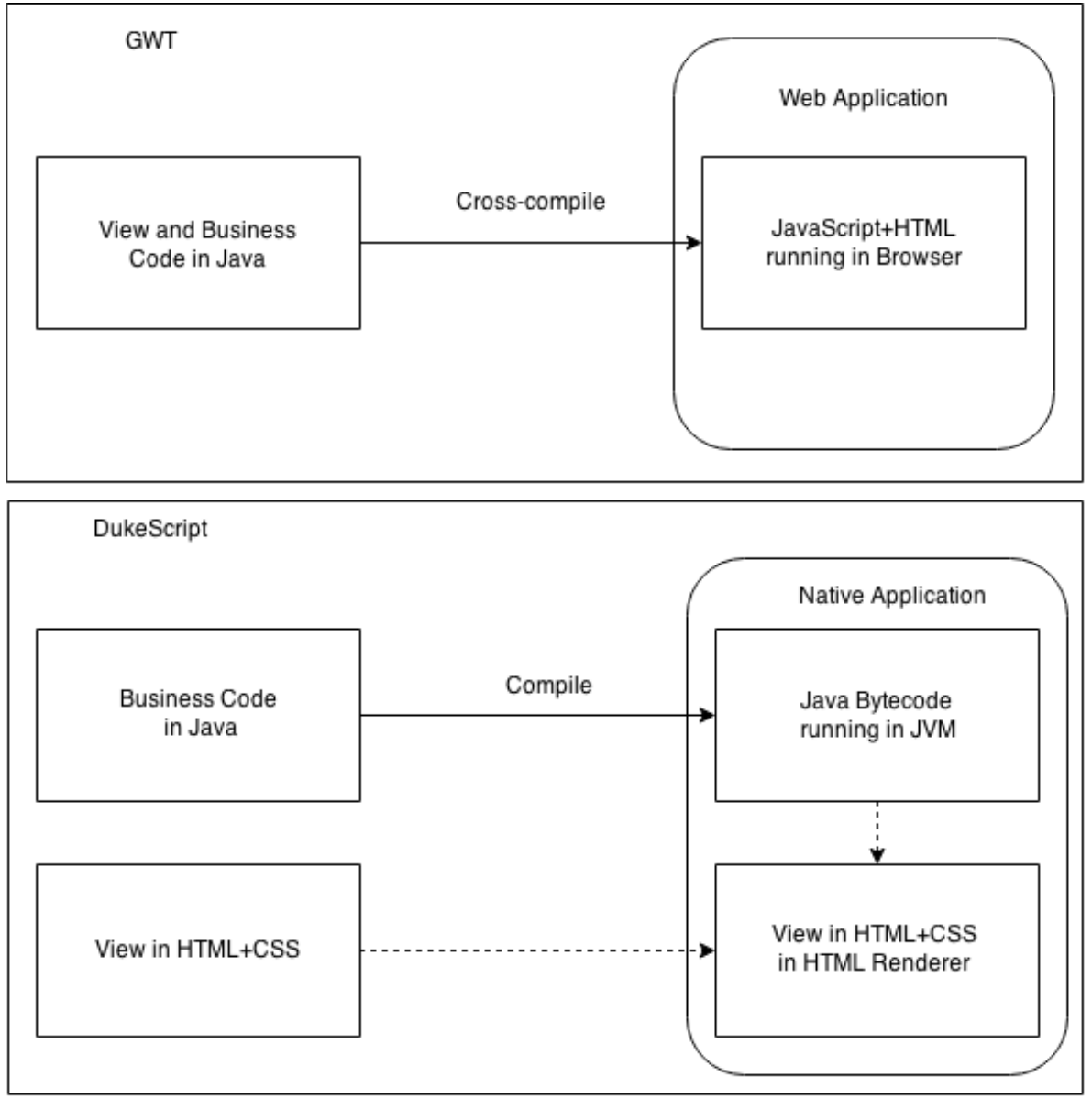
- Google Web Toolkit
- Compiler from Java to JavaScript
- Tons of other features
  - Build applications
  - Facilities for doing REST
  - UI components
- Errai: communications framework on GWT
- Vaadin: component framework on GWT

DukeScript

- Framework for creating cross-platform **mobile, desktop and web applications**.
- **Plain Java applications** that internally use HTML5 technologies and JavaScript for rendering.
- Write clean Java code and leverage the latest developments in modern UI technology.

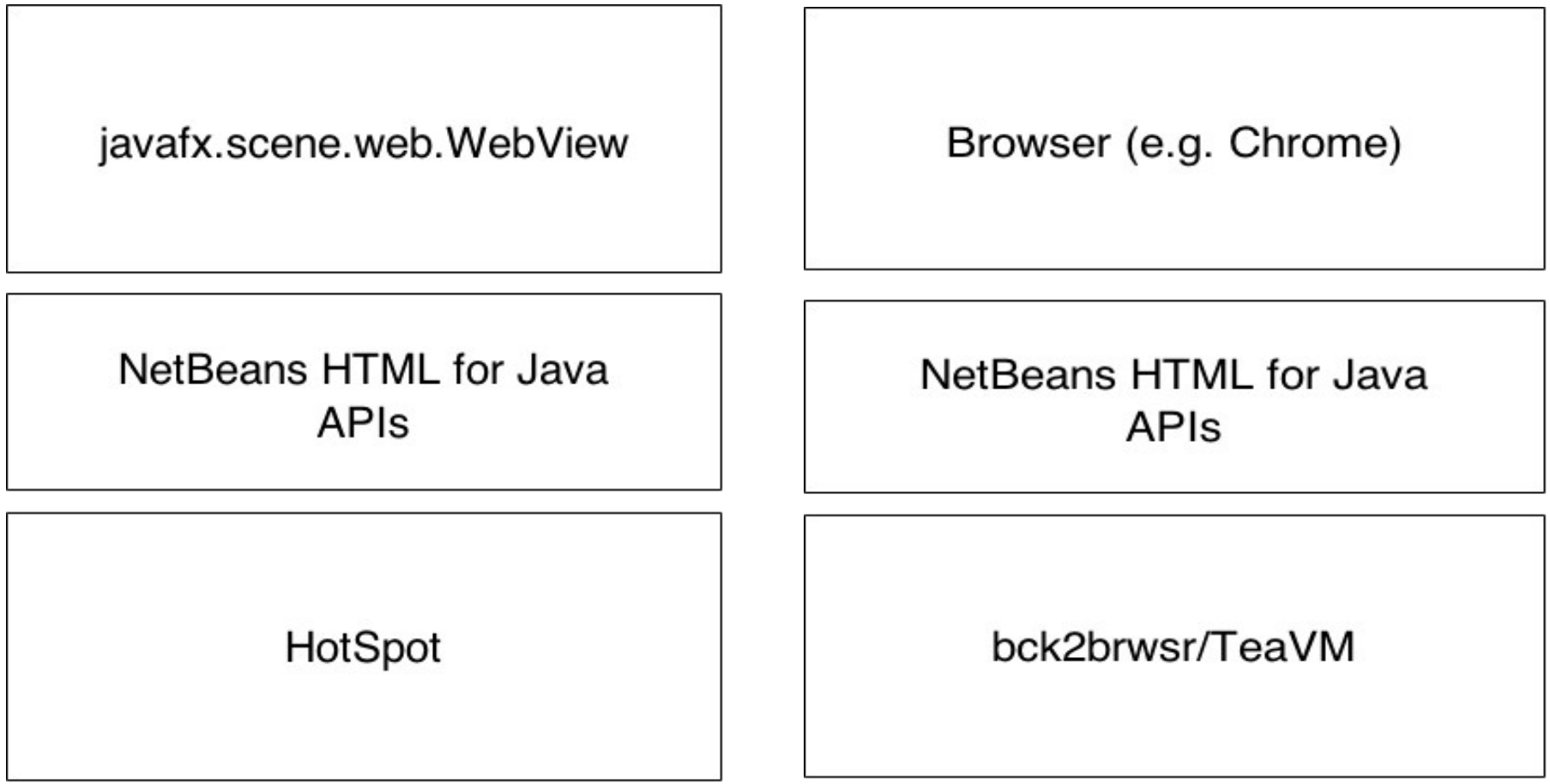
leanpub.com/everythingjs

DukeScript



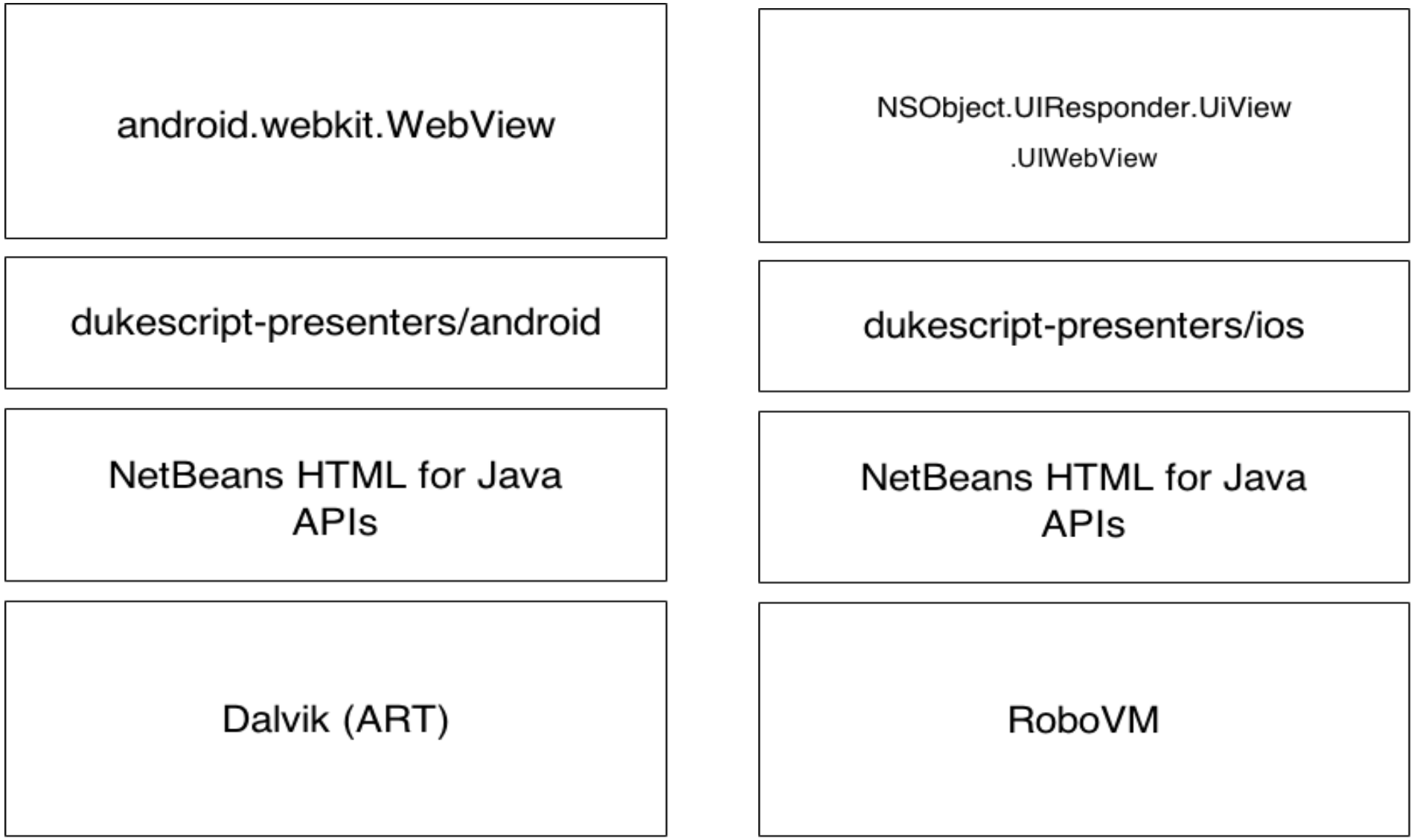
leanpub.com/everythingjs

DukeScript



leanpub.com/everythingjs

DukeScript



leanpub.com/everythingjs

Demo

leanpub.com/
everythingjs

Conclusions

JavaScript is the assembly language
for modern web applications.

The browser is the target platform.

Either write JavaScript
or write Java and transpile to JavaScript
or stick to traditional server-side approach.

Choose wisely.

leanpub.com/
everythingjs