

ClojureScript – Clojure in the browser

Tomasz Pewiński

Programming Technologies Lecture

Poznań, 7 Jan 2016

Table of Contents

Introduction

- What is ClojureScript?

- Rationale

Comparison

- Similarities to Clojure

- Differences with Clojure

Usage

- Compilation to JavaScript

- JavaScript interoperability

- Tooling and libraries

Pros and Cons

- Advantages of ClojureScript

- Disadvantages of
ClojureScript

Reality check

- Technology adoption

- How to begin?

- Example applications

Discussion

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of
ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

What is ClojureScript?

- ▶ It's a compiler
- ▶ Compiles Clojure code to JavaScript code
- ▶ Clojure in the browser and the server

What is ClojureScript?

- ▶ It's a compiler
- ▶ Compiles Clojure code to JavaScript code
- ▶ Clojure in the browser and the server

What is ClojureScript?

- ▶ It's a compiler
- ▶ Compiles Clojure code to JavaScript code
- ▶ Clojure in the browser and the server

Short history

- ▶ Announced in 2011 by Rich Hickey, creator of Clojure¹
- ▶ Latest version 1.7.170, Nov 5 2015
- ▶ Tooling, libraries, community

¹<http://blip.tv/clojure/rich-hickey-unveils-clojurescript-5399498>

Short history

- ▶ Announced in 2011 by Rich Hickey, creator of Clojure¹
- ▶ Latest version 1.7.170, Nov 5 2015
- ▶ Tooling, libraries, community

¹<http://blip.tv/clojure/rich-hickey-unveils-clojurescript-5399498>

Short history

- ▶ Announced in 2011 by Rich Hickey, creator of Clojure¹
- ▶ Latest version 1.7.170, Nov 5 2015
- ▶ Tooling, libraries, community

¹<http://blip.tv/clojure/rich-hickey-unveils-clojurescript-5399498>

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of
ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Rationale

- ▶ JavaScript runs everywhere
- ▶ Client-side browser applications
- ▶ JavaScript is fast
- ▶ JavaScript may not be adequate
- ▶ Clojure ecosystem

Rationale

- ▶ JavaScript runs everywhere
- ▶ Client-side browser applications
- ▶ JavaScript is fast
- ▶ JavaScript may not be adequate
- ▶ Clojure ecosystem

Rationale

- ▶ JavaScript runs everywhere
- ▶ Client-side browser applications
- ▶ JavaScript is fast
- ▶ JavaScript may not be adequate
- ▶ Clojure ecosystem

Rationale

- ▶ JavaScript runs everywhere
- ▶ Client-side browser applications
- ▶ JavaScript is fast
- ▶ JavaScript may not be adequate
- ▶ Clojure ecosystem

Rationale

- ▶ JavaScript runs everywhere
- ▶ Client-side browser applications
- ▶ JavaScript is fast
- ▶ JavaScript may not be adequate
- ▶ Clojure ecosystem

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Similarities to Clojure

- ▶ Clojure semantics
- ▶ Standard library
Persistent collections, Seqs, Namespaces ...
- ▶ Macros

Similarities to Clojure

- ▶ Clojure semantics
- ▶ Standard library
Persistent collections, Seqs, Namespaces ...
- ▶ Macros

Similarities to Clojure

- ▶ Clojure semantics
- ▶ Standard library
Persistent collections, Seqs, Namespaces ...
- ▶ Macros

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Differences with Clojure

- ▶ Primitives implemented using JavaScript
`(= 0 0.0) ; => true`
- ▶ Support for concurrency
JavaScript VM runs single thread
Atoms and `core.async` are available

Differences with Clojure

- ▶ Primitives implemented using JavaScript
`(= 0 0.0) ; => true`
- ▶ Support for concurrency
JavaScript VM runs single thread
Atoms and `core.async` are available

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Compilation to JavaScript

- ▶ **.cljs file extension**
- ▶ Standalone compiler, commandline
- ▶ leiningen plugins
- ▶ Supports automatic recompilation

Compilation to JavaScript

- ▶ .cljs file extension
- ▶ Standalone compiler, commandline
- ▶ leiningen plugins
- ▶ Supports automatic recompilation

Compilation to JavaScript

- ▶ .cljs file extension
- ▶ Standalone compiler, commandline
- ▶ leiningen plugins
- ▶ Supports automatic recompilation

Compilation to JavaScript

- ▶ .cljs file extension
- ▶ Standalone compiler, commandline
- ▶ leiningen plugins
- ▶ Supports automatic recompilation

Compilation to JavaScript

ClojureScript is meant to be used with Google Closure optimizer.

- ▶ Hello World without optimizations: 1.9MB
- ▶ Hello World with optimizations: 99KB
- ▶ Dead code elimination

Compilation to JavaScript

ClojureScript is meant to be used with Google Closure optimizer.

- ▶ Hello World without optimizations: 1.9MB
- ▶ Hello World with optimizations: 99KB
- ▶ Dead code elimination

Compilation to JavaScript

ClojureScript is meant to be used with Google Closure optimizer.

- ▶ Hello World without optimizations: 1.9MB
- ▶ Hello World with optimizations: 99KB
- ▶ Dead code elimination

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of
ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

JavaScript interoperability

- ▶ use `js/` syntax to call JavaScript functions
e.g. `js/alert`, `js/jquery`
- ▶ ClojureScript collections are not exactly JS objects
watch out when integrating with external libraries
- ▶ Converter functions: `js->clj`, `clj->js`
- ▶ Source maps

JavaScript interoperability

- ▶ use `js/` syntax to call JavaScript functions
e.g. `js/alert`, `js/jquery`
- ▶ ClojureScript collections are not exactly JS objects
watch out when integrating with external libraries
- ▶ Converter functions: `js->clj`, `clj->js`
- ▶ Source maps

JavaScript interoperability

- ▶ use `js/` syntax to call JavaScript functions
e.g. `js/alert`, `js/jQuery`
- ▶ ClojureScript collections are not exactly JS objects
watch out when integrating with external libraries
- ▶ Converter functions: `js->clj`, `clj->js`
- ▶ Source maps

JavaScript interoperability

- ▶ use `js/` syntax to call JavaScript functions
e.g. `js/alert`, `js/jquery`
- ▶ ClojureScript collections are not exactly JS objects
watch out when integrating with external libraries
- ▶ Converter functions: `js->clj`, `clj->js`
- ▶ Source maps

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Tooling and libraries

- ▶ **lein-cljsbuild**

- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Tooling and libraries

- ▶ lein-cljsbuild
- ▶ figwheel – live coding
- ▶ domina – DOM manipulation
- ▶ om – reactjs wrapper
- ▶ schema – data validation
- ▶ cljs.test – testing
- ▶ many many more online

<https://github.com/clojure/clojurescript/wiki#libraries>

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of
ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Advantages of ClojureScript

- ▶ All the benefits of Clojure itself
- ▶ Clojure's ecosystem
- ▶ Single language for the front-end and back-end

Advantages of ClojureScript

- ▶ All the benefits of Clojure itself
- ▶ Clojure's ecosystem
- ▶ Single language for the front-end and back-end

Advantages of ClojureScript

- ▶ All the benefits of Clojure itself
- ▶ Clojure's ecosystem
- ▶ Single language for the front-end and back-end

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of
ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Disadvantages of ClojureScript

- ▶ Inaccessible Java/JVM interop
(recently introduced reader conditionals address it)
- ▶ Awkward interoperability with JavaScript libraries
(partially mitigated by CLJSJS)

Disadvantages of ClojureScript

- ▶ Inaccessible Java/JVM interop
(recently introduced reader conditionals address it)
- ▶ Awkward interoperability with JavaScript libraries
(partially mitigated by CLJSJS)

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Technology adoption

- ▶ Mature project, used in production
- ▶ Actively supported
- ▶ <https://github.com/clojure/clojurescript/wiki/Companies-Using-ClojureScript>

Technology adoption

- ▶ Mature project, used in production
- ▶ Actively supported
- ▶ <https://github.com/clojure/clojurescript/wiki/Companies-Using-ClojureScript>

Technology adoption

- ▶ Mature project, used in production
- ▶ Actively supported
- ▶ <https://github.com/clojure/clojurescript/wiki/Companies-Using-ClojureScript>

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

How to begin?

- ▶ Interactive Koans (<http://clojurescriptkoans.com>)
- ▶ ClojureScript Tutorial
(<https://www.niwi.nz/cljs-workshop>)
- ▶ Download a project template and start coding
(<https://github.com/clojure/clojurescript/wiki#project-templates>)

How to begin?

- ▶ Interactive Koans (<http://clojurescriptkoans.com>)
- ▶ ClojureScript Tutorial
(<https://www.niwi.nz/cljs-workshop>)
- ▶ Download a project template and start coding
(<https://github.com/clojure/clojurescript/wiki#project-templates>)

How to begin?

- ▶ Interactive Koans (<http://clojurescriptkoans.com>)
- ▶ ClojureScript Tutorial
(<https://www.niwi.nz/cljs-workshop>)
- ▶ Download a project template and start coding
(<https://github.com/clojure/clojurescript/wiki#project-templates>)

Table of Contents

Introduction

What is ClojureScript?

Rationale

Comparison

Similarities to Clojure

Differences with Clojure

Usage

Compilation to JavaScript

JavaScript interoperability

Tooling and libraries

Pros and Cons

Advantages of ClojureScript

Disadvantages of

ClojureScript

Reality check

Technology adoption

How to begin?

Example applications

Discussion

Example applications

- ▶ CircleCI frontend
(<https://github.com/circleci/frontend>)
- ▶ jinteki.net (<https://github.com/mtgred/netrunner>)
- ▶ Minkowski distance visualization
(<https://github.com/pewniak747/minkowski>)

Example applications

- ▶ CircleCI frontend
(<https://github.com/circleci/frontend>)
- ▶ jinteki.net (<https://github.com/mtgred/netrunner>)
- ▶ Minkowski distance visualization
(<https://github.com/pewniak747/minkowski>)

Example applications

- ▶ CircleCI frontend
(<https://github.com/circleci/frontend>)
- ▶ jinteki.net (<https://github.com/mtgred/netrunner>)
- ▶ Minkowski distance visualization
(<https://github.com/pewniak747/minkowski>)

Summary

ClojureScript follows the “compile-down” pattern of many languages.

Higher-level languages compile to runtime representations:

- ▶ C++ → machine code
- ▶ clojure, java → JVM bytecode
- ▶ ClojureScript, Opal (Ruby), ScalaJS, PyJS, SharpKit (C#)
→ JavaScript

Discussion

Thank you for your attention.
I'll be happy to answer any questions.