NetBeans support for EcmaScript 6

Kostas Saidis

@saikos

NetBeans Day Athens 2017

April 21, 2016

About Me

- Software and Data Architect

- Adjunct Lecturer @ [di.uoa.gr](di.uoa.gr) (University of Athens) teaching Software Engineering

- Managing Director @ niovity.com

- Early Java enthusiast (since 1997)

- Diving into JavaScript since 2010

The JavaScript Ecosystem

Brief history

1993: First web browser
Mosaic by NCSA, University of Illinois

1994: Netscape browser
Netscape Navigator (code named Mozilla)

1995: Brendan Eich develops JavaScript
Netscape Navigator 2.0 & Netscape Enterprise Server

The core of JavaScript

A dynamically typed, interpreted language

- that was intentionally designed to look like Java

- supporting Scheme-like functions (functional paradigm)

- and Self-like prototypes (object-oriented paradigm)

Douglas Crockford, "JavaScript: The good parts", 2008, O'Reilly Media

Example

https://github.com/monsur/jscache

```javascript
function Cache(maxSize, debug, storage) {
  this.maxSize_ = maxSize || -1;
  this.debug_ = debug || false;
  this.storage_ = storage ||
                  new Cache.BasicCacheStorage();

  this.fillFactor_ = .75;

  this.stats_ = {};
  this.stats_['hits'] = 0;
  this.stats_['misses'] = 0;
}
```

The first 20 years

1996

- Microsoft JScript engine (in IE 3.0 & IIS)

- Mozilla SpiderMonkey JavaScript engine (Netscape Navigator 3.0)

- Submission of JavaScript to ECMA International

- Opera Browser

- NetBeans IDE!

The first 20 years

1997

- Dynamic HTML (IE 4)

- ES1 (ECMA-262 standard)

The first 20 years

1998

- Netscape releases the source of Navigator

- Mozilla Foundation

- Gecko HTML engine

- Rhino JavaScript engine (Java)

- ES2 (ISO/IEC 16262 standard)

The first 20 years

- AJAX, XMLHTTPRequest (IE 5)

- Macromedia Flash

- ES3

    - Regular expressions

    - Exceptions, try/catch blocks

    - `do...while`, `in`, `instanceof`

    - Enhancements in arrays, strings, numbers

10

The first 20 years

2000

- ES4 (Never released, abandoned)

- Adobe ActionScript, ES4 derivative (classes, static typing, etc)

- ES6 contains many ES4 features

The first 20 years

2001

- Douglas Crockford
  - [The World's Most Misunderstood Programming Language](#)
  - JSON

The first 20 years

2002

- Mozilla Suite 1.0

- Firefox

- Outlook Web App (AJAX goes mainstream)

- JSON.org

The first 20 years

2003

- Safari Web Browser

- WebKit & Nitro engines (forks of KDE HTML/JS engines)

The first 20 years

2004

- Google Mail (AJAX for the masses)

- Dojo toolkit

The first 20 years

2005

- Jesse James Garrett, Ajax: A New Approach to Web Applications

- prototype.js, the dollar function $()

The first 20 years

2006

- jQuery

- Yahoo! User Interface library

The first 20 years

2007

- PC World, Netspace Navigator: the best tech product of all time

The first 20 years

2008

- Google Chrome

- V8 JavaScript engine

- Chromium Project

The first 20 years

2009

- ES5 (ES3.1 essentially)

- node.js (server-side JS)

- CommonJS (JS modules)

- AngularJS

- PhoneGap, hybrid JS apps (now Apache Cordova)

The first 20 years

2009

- CoffeeScript, the first language to transpile to JS

- underscore.js, functional programming in JS (superseded by lodash.js)

The first 20 years

2010

- Backbone.js, MVC library

- Express.js, Web framework for node.js

- Knockout.js, MVC library

- PhantomJS, Headless Webkit

The first 20 years

2011

- Data-driven documents wit D3.js

- Asynchronous Module Definitions (AMD) with require.js

- Ember.js, MVC library

The first 20 years

2012

- Oracle releases Nashorn JavaScript engine (Java), the default JavaScript engine in Java 8

- Meteor.js, web framework (node.js, hybrid, reactive)

- TypeScript by Microsoft (transpiled to JS)

The first 20 years

2013

- React framework by Facebook

- asm.js by Mozilla

- ElectronJS by Github (atom-shell)

The first 20 years

2014

- HTML5 - APIs everywhere

- The (unofficial) death of XML?
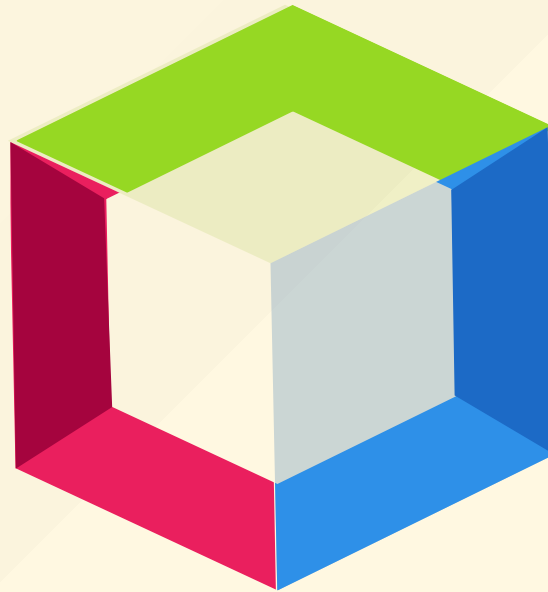
The first 20 years

2015

- ES6 (Harmony, ES2015)

- React Native (hybrid apps)

- WebAssembly (assembly for the web)

JavaScript is here to stay!

- JavaScript *is the lingua franca* of the Web rapidly turning into a general-purpose programming language with a massive usage base

- ES2016 (minor additions to ES6)

- ESNext (next features, like async/await)

28

Apache NetBeans

is here to stay, too!

JavaScript Support in NetBeans 8.2

- Built-in support for JS and a multitude of tools, libraries and frameworks

- Support for ES6 and beyond

Built-in JS Support

- Syntax highlighting

- Code folding

- Code completion and analysis

- Refactoring

- Structural navigation

- Lints/Hints

- Integrated debugger

And more

- Bracket matching

- Go to declaration

- JSDoc

- JSON

- JS embedding (HTML, PHP, etc)

Developing JS with Netbeans

- Embedded web server (run your apps from within the IDE)

- Automagical update in the web browser (Embedded Webkit or Chrome with plugin)

Tools and Libraries

- Inherent support for JS libs/tools at the source level

- Node.js and npm

- Grunt, Gulp, Bower, Require.js, Karma, Express.js, Babel, Webpack, …

- LESS, SASS

- Apache Cordova

- Oracle JET

- NB Plugins

ES6 Features

- Modules with `import` and `export`

- Constants with `const`

- Block scoped variables with `let`

- Arrow functions with `=>`

- Classes with `class` and `extends` and `super`

- String interporation with `` `${var}` ``

ES6 Features

- Default parameter values, rest & spread operators

- Desctructuring of objects and arrays

- Promises

- Enhanced object literals

- New collections (Set, Map, WeakSet, WeakMap)

Not covered here

- Generators & iterators

- Various other goodies, such as

  - Symbols

  - Metaprogramming (Proxies & Reflection)

  - Math, String, Number enhancements

  - Binary and octal literals

  - and more…

Example

References

A brief history of JavaScript

A re-introduction to JavaScript

ECMAScript 6 New Features: Overview & Comparison

NB Wiki

ECMAScript 6, Modules, Babel, Webpack, and NetBeans IDE

Thank you!

@saikos

Linkedin