

Apache FlexJS Maven migration initiative



Christofer Dutz <christofer.dutz@codecentric.de>

ApacheCon EU 2016 - Seville - 2016-11-18

WHO AM I?

- Christofer Dutz
- Senior IT Consultant
- codecentric AG
- Apache Member
- Apache Flex Committer & PMC
- Flexmojos Lead Developer
- Open-Source enthusiast



AGENDA

- Developing FlexJS in 2015
- Why migrate to Maven
- Migration strategy
- Developing FlexJS in 2016
- `flexjs-maven-plugin`
- Ongoing migrations

DEVELOPING FLEXJS IN 2015

- Building with Ant only
 - Building FlexJS required
 - Cloning 2 GIT repositories
 - Manually installing multiple tools
 - Setting several environment variables ($x > 6$)
 - Requires a strict directory structure of all projects
- Only a hand full of developers working on FlexJS

DEVELOPING FLEXJS IN 2015

- Problems with the build
 - Relies on direct downloads
 - Dependencies not verified
 - Build scripts quite hard to understand
 - Circular dependencies
- CI feedback takes several hours
 - Frustrates people working on FlexJS

WHY MIGRATE TO MAVEN?

- WHY MIGRATE to Maven?
 - Simplify the project
 - Simplify the build process
 - Make the build process more robust
 - Get more people involved

WHY MIGRATE TO MAVEN?

- Why migrate TO MAVEN?
 - Maven makes it hard to be sloppy
 - Helps volunteers to get started
 - Repo-System makes the build robust

MIGRATION STRATEGY

- Update the directory structure
- Find cycles in the build
- Refactor to eliminate cycles
 - Extract code-generators into custom maven plugins
 - Extract dependencies of code-generators into modules
 - Refactor tests not to rely on cycles
 - Split up into 3 repositories
- Get third party libraries mavenized

MIGRATION STRATEGY

- Educating the community



The screenshot shows a video player interface. At the top left, there is a circular profile picture of Justin Mclean and the video title "ApacheCon 2016 Maven Training" in blue text. Below the title, it says "von Justin Mclean" and "@codecentric". The main content of the video is a presentation slide with a dark background and a green wireframe pattern. The slide features the Apache Flex logo, which consists of the word "apache" in a small font above the word "Flex" in a large, bold, white font. To the right of the text is a colorful logo made of four interlocking loops in red, green, blue, and orange. Below the logo, the text reads "Maven Introduction for Apache Flex". At the bottom of the slide, it says "Christofer Dutz <christofer.dutz@codecentric.de>" and "ApacheCon NA 2016 - Vancouver 2016-05-11". The video player controls at the bottom include a play button, a progress bar showing 14:32, and icons for HD and full screen.

<https://vimeo.com/167857327/da3e3239bc>

DEVELOPING FLEXJS IN 2016

- Set one environment variable
- Checkout 3 repositories
- Run the following on each of them

```
mvn -s settings-template.xml clean install
```

DEVELOPING FLEXJS IN 2016

- CI feedback in about 10 minutes

 	FlexJS Compiler (maven)	2 Stunden 40 Minuten - #314	1 Tag 0 Stunden - #310	4 Minuten 34 Sekunden	 
 	FlexJS Framework (maven)	2 Stunden 34 Minuten - #415	1 Tag 12 Stunden - #408	3 Minuten 20 Sekunden	 
 	FlexJS Pipeline	Nicht anwendbar	Unbekannt	Nicht anwendbar	 
 	FlexJS Pipeline » develop	19 Stunden - #21	10 Tage - #17	19 Minuten	 
 	FlexJS Typedefs (maven)	2 Stunden 36 Minuten - #141	4 Monate 3 Tage - #1	1 Minute 44 Sekunden	 

DEVELOPING FLEXJS IN 2016

- Jenkins Multibranch Pipeline build



The screenshot shows a Jenkins Multibranch Pipeline build table. At the top left, there are two buttons: 'All' and '+'. The table has six columns: 'S', 'W', 'Name', 'Letzter Erfolg', 'Letzter Fehlschlag', 'Letzte Dauer', and 'Console'. There are two rows of build data. The first row shows a successful build for the 'develop' branch, with the last success at '4 Tage 23 Stunden - #11' and the last failure at '26 Tage - #8', taking '35 Minuten'. The second row shows a failed build for the 'feature-autobuild/example-maven-dirs' branch, with the status 'Nicht anwendbar' and 'Unbekannt' for the last failure, also marked as 'Nicht anwendbar'. Each row includes a status icon (a blue sphere for success, a sun for failure) and a console icon.

S	W	Name	Letzter Erfolg	Letzter Fehlschlag	Letzte Dauer	Console
		develop	4 Tage 23 Stunden - #11	26 Tage - #8	35 Minuten	 
		feature-autobuild/example-maven-dirs	Nicht anwendbar	Unbekannt	Nicht anwendbar	 

- Full Stage View
- Job Config History
- Embeddable Build Status
- Test Results Analyzer
- Pipeline Syntax

Stage View

Average stage times:
(Average full run time: ~36min 42s)

	Wipe Workspace	Checkout Upstream Projects	Build FlexJS Compiler	Build FlexJS Typedefs	Build FlexJS Framework	Release Site Changes
Average stage times:	1min 22s	1min 24s	5min 23s	1min 55s	24min 17s	1s
#12 Oct 27 17:18						
#11 Oct 22 17:29	2min 16s	1min 12s	4min 49s	1min 47s	25min 0s	75ms
#10 Oct 13 13:26	1min 21s	1min 15s	5min 49s	2min 16s	30min 29s	93ms
#9 Oct 04 23:37	19s	1min 14s	4min 47s	1min 48s	23min 37s	49ms
#8 Oct 01 08:39	1min 51s	1min 18s	5min 9s	2min 5s	5min 50s <small>failed</small>	
#7 Sep 30 16:22	1min 54s	2min 19s	5min 44s	1min 53s	30min 10s	412ms

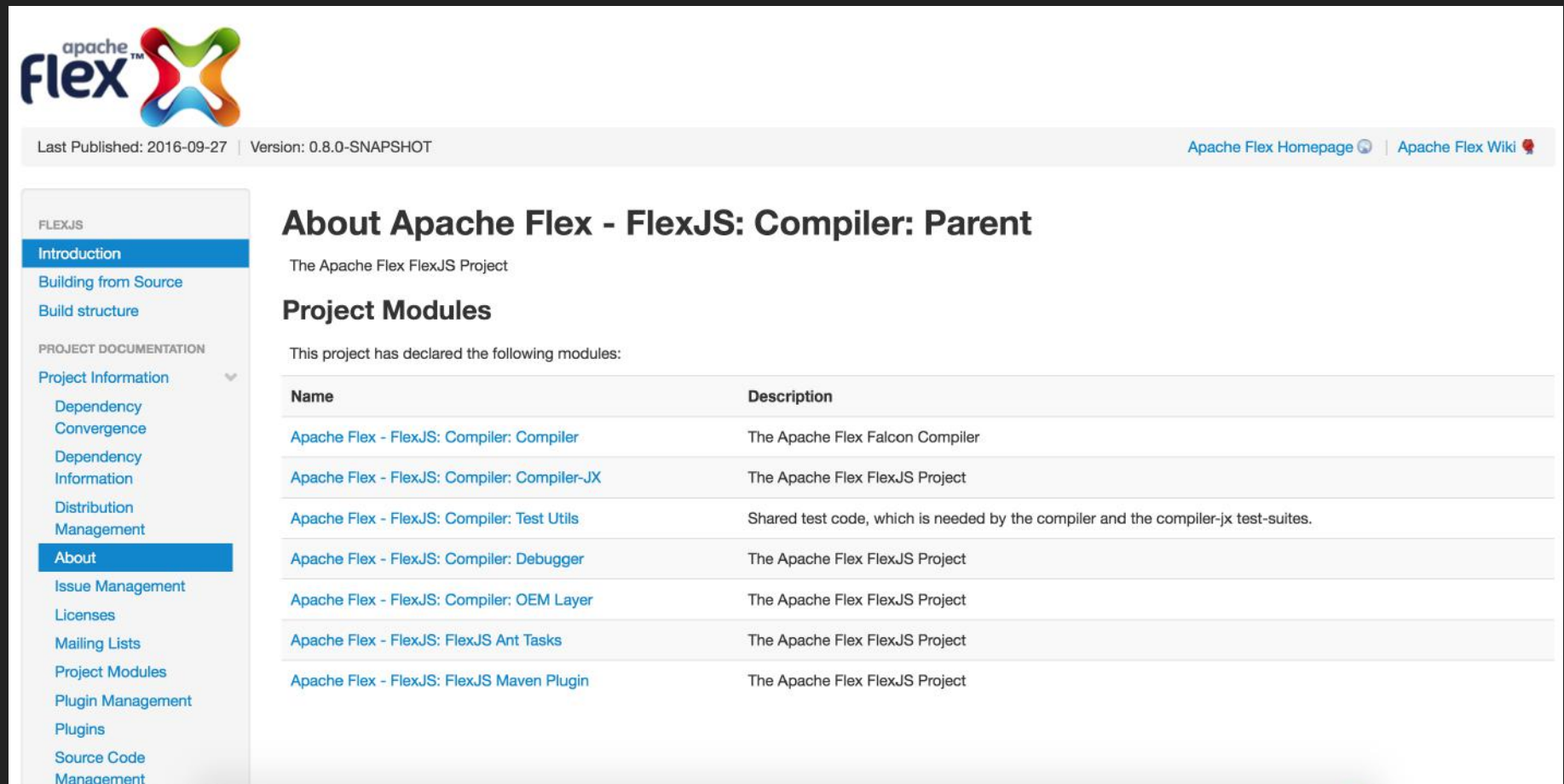
Build-Verlauf Trend

find


- #12 27.10.2016 15:18
- #11 22.10.2016 15:29
- #10 13.10.2016 11:26
- #9 04.10.2016 21:37
- #8 01.10.2016 06:39
- #7 30.09.2016 14:22
- #6 30.09.2016 07:48
- #5 30.09.2016 07:40
- #4 29.09.2016 21:56
- #3 29.09.2016 15:05
- #2 29.09.2016 12:56
- #1 29.09.2016 12:55

DEVELOPING FLEXJS IN 2016

- Automatic Site generation



The screenshot displays the Apache Flex FlexJS Project website. At the top left is the Apache Flex logo, and below it, the text 'Last Published: 2016-09-27 | Version: 0.8.0-SNAPSHOT'. On the top right, there are links for 'Apache Flex Homepage' and 'Apache Flex Wiki'. The main content area is titled 'About Apache Flex - FlexJS: Compiler: Parent' and includes a sub-header 'Project Modules'. Below this, a table lists the project's modules with their names and descriptions.

apache flex™ 

Last Published: 2016-09-27 | Version: 0.8.0-SNAPSHOT

[Apache Flex Homepage](#) | [Apache Flex Wiki](#)

FLEXJS

- Introduction
- Building from Source
- Build structure

PROJECT DOCUMENTATION

- Project Information
 - Dependency Convergence
 - Dependency Information
 - Distribution Management
 - About**
 - Issue Management
 - Licenses
 - Mailing Lists
 - Project Modules
 - Plugin Management
 - Plugins
 - Source Code Management

About Apache Flex - FlexJS: Compiler: Parent

The Apache Flex FlexJS Project

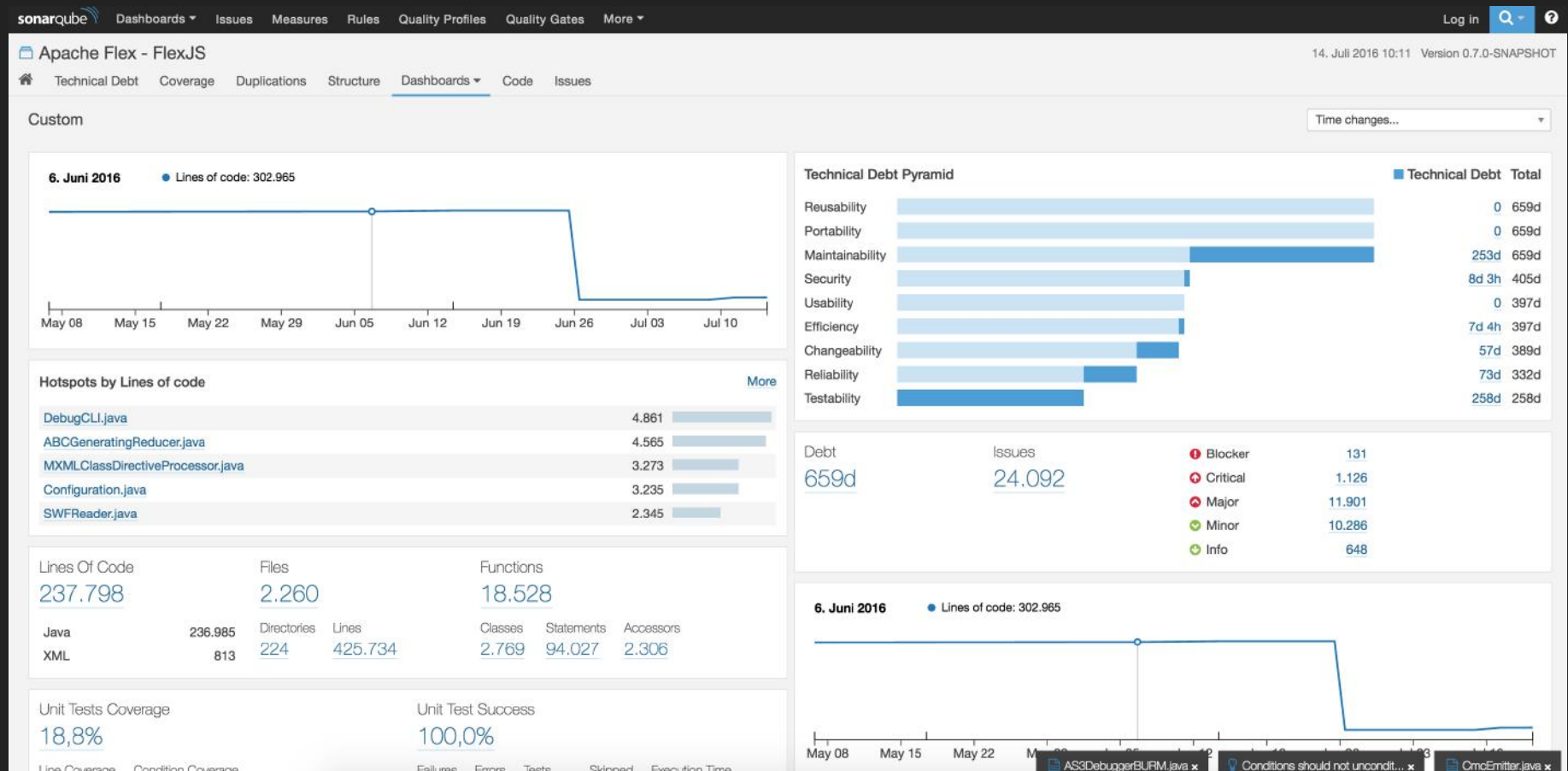
Project Modules

This project has declared the following modules:

Name	Description
Apache Flex - FlexJS: Compiler: Compiler	The Apache Flex Falcon Compiler
Apache Flex - FlexJS: Compiler: Compiler-JX	The Apache Flex FlexJS Project
Apache Flex - FlexJS: Compiler: Test Utils	Shared test code, which is needed by the compiler and the compiler-jx test-suites.
Apache Flex - FlexJS: Compiler: Debugger	The Apache Flex FlexJS Project
Apache Flex - FlexJS: Compiler: OEM Layer	The Apache Flex FlexJS Project
Apache Flex - FlexJS: FlexJS Ant Tasks	The Apache Flex FlexJS Project
Apache Flex - FlexJS: FlexJS Maven Plugin	The Apache Flex FlexJS Project

DEVELOPING FLEXJS IN 2016

- Static code analysis using SonarQube



COMPILING FLEXJS APPLICATIONS

- Flex compiler defined by config.xml file
- Settings in config.xml overridden by commandline parameters

COMPILING WITH ANT

- Manually create config.xml
- Additionally provide commandline as input to Ant target
- Usually base config.xml for common cases
 - reused and customized for individual builds
- Tricky to configure
- Hard to find out how the compiler was configured
- Relies on environment variables

FLEXJS-MAVEN-PLUGIN

- config.xml generated on the fly
- Templates provide the base content of config.xml
- Maven plugin defines parameters and default values
 - Can be overridden in pom.xml
- Compiler uses only options in generated config.xml
- config.xml located in `target` directory
- Only requires one environment variable for running unit-tests

ONGOING MIGRATIONS

- Website generated with `maven-site-plugin`
- Automatic download of Flashplayer-Debugger in the converter (for mac it requires implementing code to unpack dmg files)
 - FlexJS builds without any environment variables
- Release the `flex-sdk-converter`
- Extend the `flexjs-maven-plugin` to support unit-testing