

A Journey into the Compiler and Tooling

Florian Verdonck

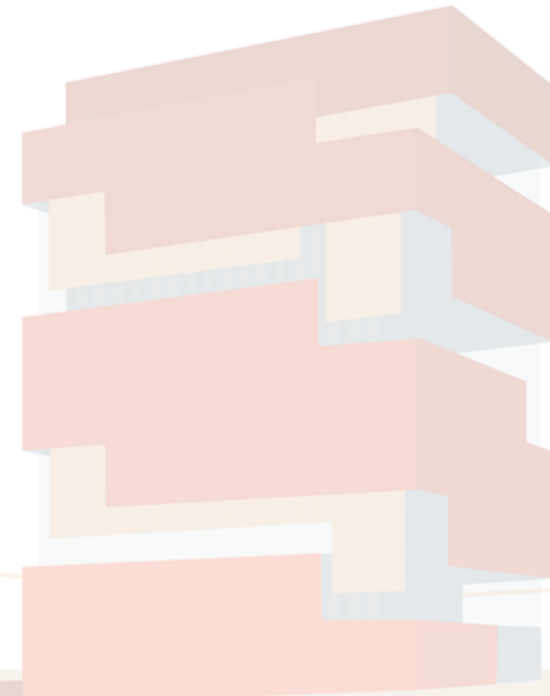
Agenda

- From F# to JS
- Fable Clients
- What's next



About this talk

- Existing Fable users
- Deeper understanding
- Heroes journey
- F# scripts
- Demo's





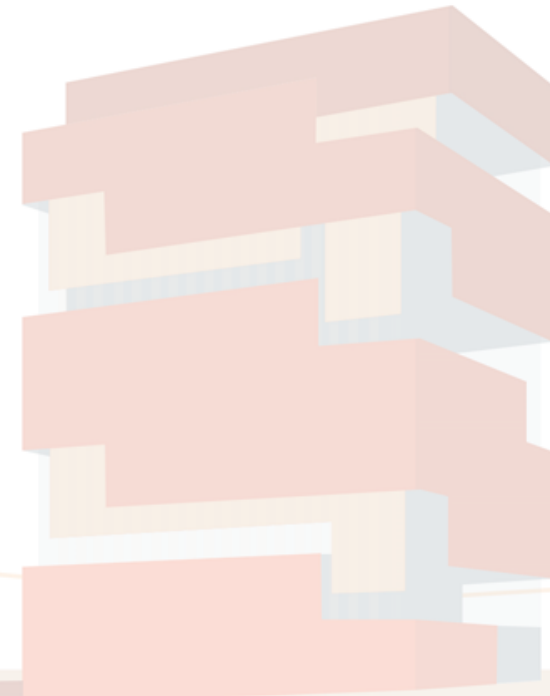
From F# to JS: Fable

From F# to JS



From F# to JS: FCS

- FSharp Compiler Service
- Fork of dotnet/fsharp
- Re-packaged and on Nuget
- [TAST](#)



From F# to JS: Fable AST

- Intermediate format
- Interop ability
- Replacement



From F# to JS: Babel AST

- JSON
- Babel toolchain





Barebone Fable

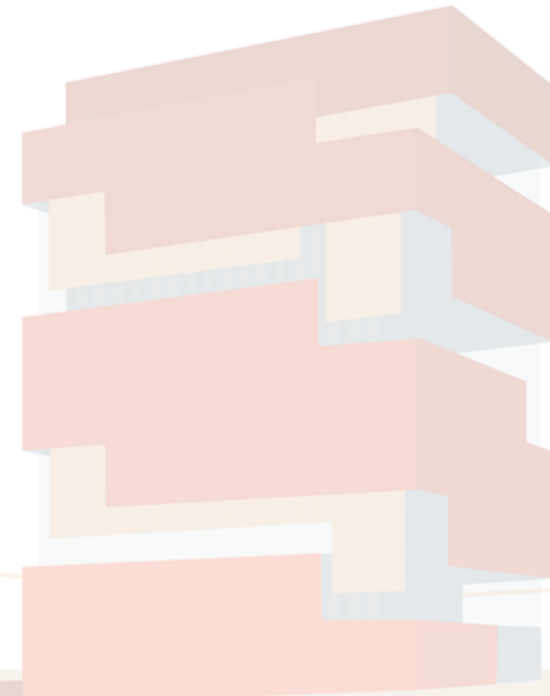
Fable Clients

- fable-compiler impractical
- Incremental compilation
- Compile on file change
- fable-splitter
- fable-loader
- ...



Fable Clients: splitter

- Individual files
- NodeJS
- Unit tests
- Easy cli
- Less setup

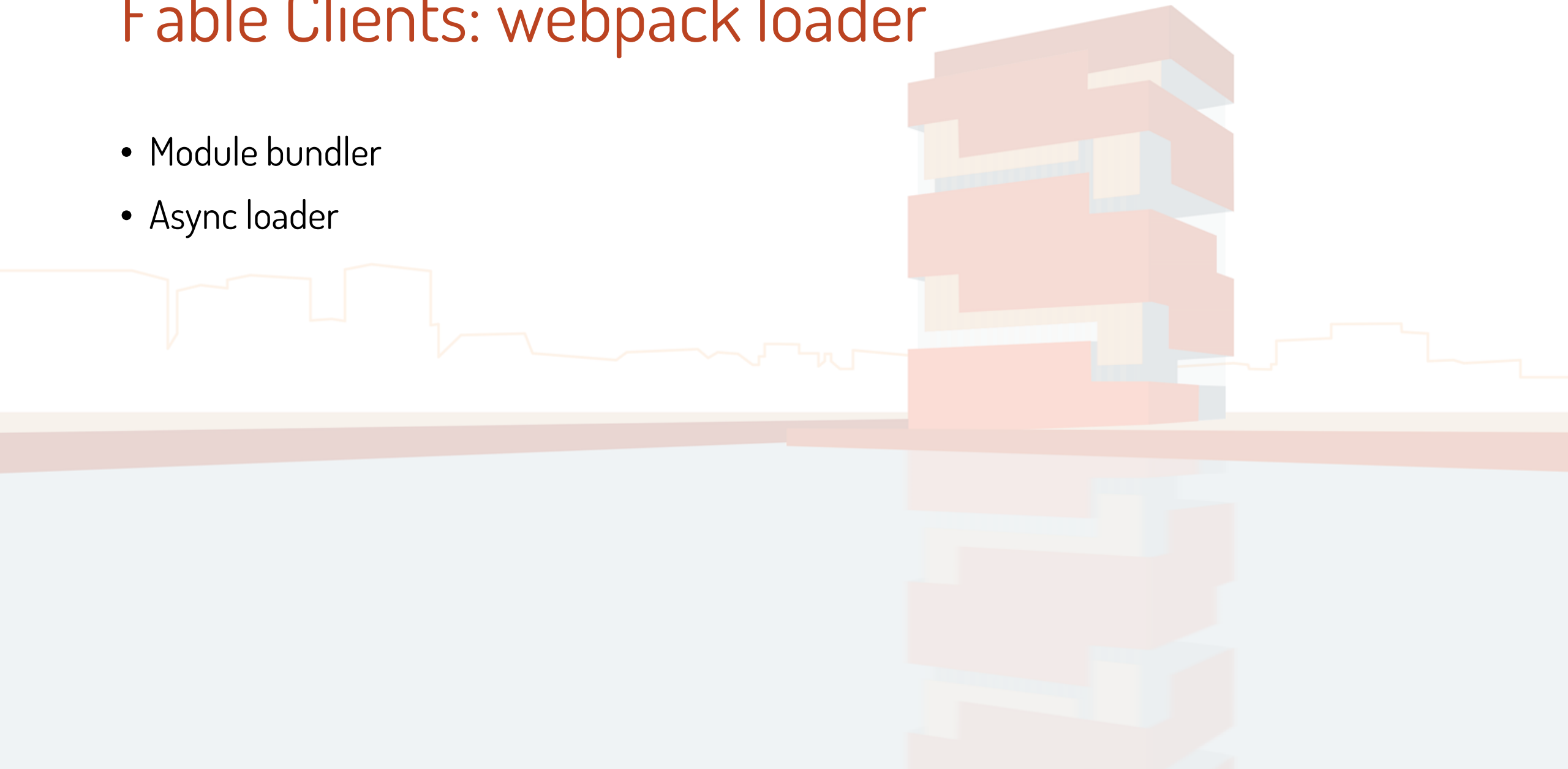


The image shows a close-up, top-down view of a large pile of chopped firewood. The wood is cut into irregular, roughly triangular or wedge-shaped pieces. The color of the wood varies from a light tan or yellowish-brown to a dark, almost black charred surface, particularly on the outer edges and bark. The texture appears rough and fibrous. The pieces are packed closely together, filling the entire frame. In the center of the image, the text "MVC & Pika" is overlaid in a clean, white, sans-serif font. The text is centered both horizontally and vertically, with a bounding box of approximately [414, 468, 570, 536].

MVC & Pika

Fable Clients: webpack loader

- Module bundler
- Async loader



A close-up photograph of a spider on its web. The spider has a yellow and black patterned body and legs. The web is a complex, spiral structure made of fine, silken threads. The background is a soft, out-of-focus green, suggesting foliage. The text "Lazy / Suspense" is written in a white, sans-serif font, centered horizontally and slightly below the vertical center of the image.

Lazy / Suspense

**YOUR
SCRIPT
ARGUMENT
SUCKS**

STAR WARS PODCAST PUBLIC SERVICE ANNOUNCEMENT
Daily podcasts from Star Wars Celebration Europe & San Diego Comic-Con 2015
Follow us on www.StarWars.com, Twitter / Insta your sticker to @StarWars

Fable F# scripts

- Personal choice
- 1 fsx, 1 js, 1 command
- Sorta needs Paket
- FAKE like integration?
- F# lang suggestion 542



Fable F# scripts: FAKE

```
1:  #r "paket:  
2:  nuget Fake.Core.Target prerelease"  
3:  #load "../.fake/myscript.fsx/intellisense.fsx"
```

Fable F# scripts: F# lang 542


[fsharp](#) / [fslang-suggestions](#)

[Watch](#) 84 [Star](#) 185 [Fork](#) 9

[Code](#) [Issues 286](#) [Pull requests 0](#) [Projects 0](#) [Security](#) [Insights](#)

Package references in F# scripting #542

[Open](#) ctaggart opened this issue on Feb 22, 2017 · 89 comments



ctaggart commented on Feb 22, 2017 • edited by dsyme

[edited by @dsyme to be a more comprehensive guide to this design question]

[Latest implementation is here: <https://github.com/dotnet/fsharp/pull/4042>]

Package references in F# scripts

There has been a long standing desire to add the ability to reference nuget packages from F# scripts. Originally this was conceived as basic fully qualified nuget references like `packages.config`. Lately this has evolved into integrating "dotnet" or "paket" or "npm" package specifications and dependency management tools as part of the toolchain, or providing generic hooks to allow this.

Assignees

No one assigned

Labels

[approved-in-principle](#) [needs rfc](#) [proposed-priority](#) [started](#)

Projects

None yet

```
#package "Microsoft.Azure.WebJobs@2.0.0-*"
```

What's next?

- Fable compiles Fable
- JS only
- REPL



Florian Verdonck
@verdonckflorian

Coollest part of my day: I've compiled [#fsharp](#) code to JavaScript in the browser on an airplane!
[@FableCompiler](#) rocks! [#fableconf](#)

[Tweet vertalen](#)

3:35 p.m. · 30 jul. 2019 · [Twitter for Android](#)



fable-compiler-js



Florian Verdonck

.NET consultant @ [Axxes](#)



[@verdonckflorian](#)



[@nojaf](#)

