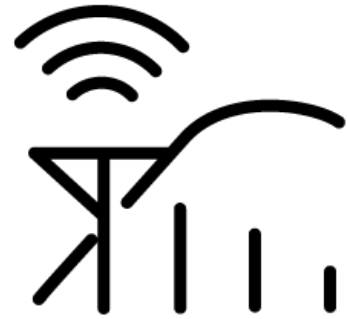


Errata

- スライド17
実演できませんでした. また, 表記されているGitHub リポジトリは公開されていませんでした. 現在は有効なリポジトリです.
- スライド20
2017年2月16日に更新とありますが, 更新しませんでした. しばらくは更新する予定はありません.



Compilation

Akihiko Odaki <akihiko.odaki.4i@stu.hosei.ac.jp>





whoami

- 某大学の2年 (来年度3年生なのでavailable for hire!)
- C言語おいしい. Webの開発歴は半年以下なので優しくしてください.
- 東京理科大学の神楽坂一丁目通信局というサークルにお邪魔しています
<https://kaguchou.net/>
(メールのドメインと大学名が違うとか気づいた勘のいい人はすごーい!)
- 局内で使っている名簿等のシステム, TsuboneSystemをフルスクラッチで書き直しています (Webpackを使用済)
<https://github.com/kaguchou/tsubonesystem3>

メール確認

確認メールを再送信する

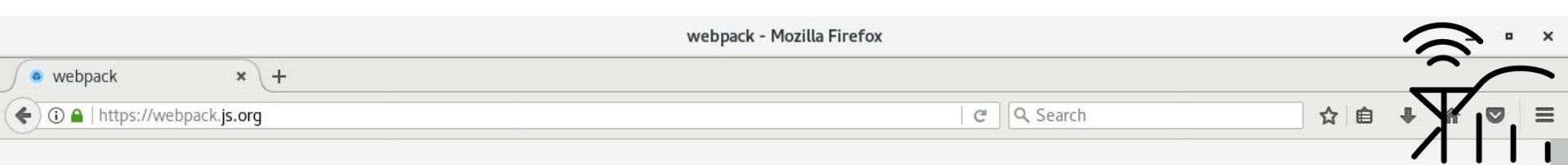
電話番号

000-000-001

性別

男

☒ Prog部



Bundle with webpack.

webpack.config.js

```
module.exports = {  
  entry: './app.js',  
  output: {  
    filename: 'bundle.js'  
  }  
}
```

page.html

```
<html>  
  <head>  
    ...  
  </head>  
  <body>  
    ...  
    <script src="bundle.js"></script>  
  </body>  
</html>
```

Then run `webpack` on the command-line to create `bundle.js`.

It's that simple.

[Get Started](#)

すごーい!



h?

page.html



```
<html>
  <head>
    ...
  </head>
  <body>
    ...
    <script src="bundle.js"></script>
  </body>
</html>
```

Then run `webpack` on the command-line to create `bundle.js`.



HTMLどっからきたの？
これじゃあWebpackじゃなくてJspackじゃ
ん!*

* JavaScriptからインポートされたassetsは扱えます



作戦立案

1. それっぽいHTMLを出力するloaderやpluginを作る

例: html-webpack-plugin

<https://www.npmjs.com/package/html-webpack-plugin>

pros: 実装するの簡単

cons: 汎用性は低い (JavaScriptのエントリーポイント以外のassetsをHTMLに直接適用するのはどうすればいいんですか?)



作戦立案

2. html-loaderとentry pointを作るloaderの合わせ技

pros: webpack-contribにあるhtml-loaderの汎用性を活用できる

cons: 複数のentrypointに正しく対応できない (CommonsChunkPlugin等が死ぬ)

```
<#list list as list>
```

```
  <a href="{list.href!}" />{list.name}</a>
```

```
</#list>
```

```

```

```
<div>{require('./components/gallery.html')}</div>
```



作戦発動

TsuboneSystemでは後者の案を採用

-> entry pointを作るloaderの開発に臨む!

-> これが厄介だった...



entrypointと普通のmodulesの違い

普通のmodules

transpile等はされるものの至って普通

```
api.js
~/go/src/github.com/kaguchou/tsubonesystem3/webapp/src/private/client

1 /**
2  * @file api.js implements a WebAPI interface.
3  * @author Akihiro Odaki <akihiro.odaki.41@stu.hosei.ac.jp>
4  * @copyright 2017..{@link https://kaguchou.net/|Kaguchou}
5  * @license AGPL-3.0+
6  */
7
8 /** @module private/client/api */
9
10 import Promise from "../promise";
11
12 /**
13  * ajax returns fetched and decoded JSON.
14  * @private
15  * @param {!String} uri -- The URI.
16  * @param {!String} method -- The method.
17  * @param {?String} [token] -- The token.
18  * @param {?} [data] -- The data.
19  * @returns {!module:private/promise}.A.promise resolved with fetched JSON.
20  */
21 function ajax(uri, method, token, data) {
22   const xhr = new XMLHttpRequest();
23
24   const opts = {
25     accepts: {json: "application/json; charset=UTF-8"},
26     data, dataType: "json", method,
27     xhr() {
28       return xhr;
29     },
30   };
31
32   /**
33    * RFC 6750 -- The OAuth 2.0 Authorization Framework: Bearer Token Usage
34    * 1.. Introduction
35    * https://tools.ietf.org/html/rfc6750#section-3
36    * The Bearer authentication scheme is intended primarily for
37    * server authentication using the WWW-Authenticate and
```



entrypointと普通のmodulesの違い

entrypoint

modulesをまとめ、実行するランタイムがある!
どうしよう?

```
bundle.js
1 /*****(function(modules){ //webpackBootstrap
2 /*****/. //The module cache
3 /*****/. var installedModules = {};
4
5 /*****/. //The require function
6 /*****/. function _webpack_require_(moduleId) {
7
8 /*****/. //Check if module is in cache
9 /*****/. if(installedModules[moduleId])
10 /*****/. return installedModules[moduleId].exports;
11
12 /*****/. //Create a new module (and put it into the cache)
13 /*****/. var module = installedModules[moduleId] = {
14 /*****/. i: moduleId,
15 /*****/. l: false,
16 /*****/. exports: {}
17 /*****/. };
18
19 /*****/. //Execute the module function
20 /*****/. modules[moduleId].call(module.exports, module, module.exports, _webpack_require_);
21
22 /*****/. //Flag the module as loaded
23 /*****/. module.l = true;
24
25 /*****/. //Return the exports of the module
26 /*****/. return module.exports;
27 /*****/. }
28
29
30 /*****/. //Expose the modules object. (_webpack_modules_)
31 /*****/. _webpack_require_.m = modules;
32
33 /*****/. //Expose the module cache
34 /*****/. _webpack_require_.c = installedModules;
35
36 /*****/. //Identity function for calling harmony imports with the correct context
37 /*****/. _webpack_require_.i = function(value) { return value; };
```



Compilation



Compilation

Disclaimer: 文書化されていないAPIを使用しています! 製品で使わないでください!

- loaderはthis._compilation, pluginはthis.plugin("compilation")でアクセス可能.
- Webpackの内部のほぼ全てへのアクセスを可能にする -> すごい!!!
- ソースコードがドキュメンテーション
(Webpackのソースコードは読みやすいし大丈夫ですよね!?)
- Compilation#createChildCompilerっていう良さげな関数がある
<https://github.com/webpack/webpack/commit/ee01837d66a44f1dd52fd1e174a6669e0d18dd55#diff-7a14ca420cc4fa7cf55f788246117e7cR207>
これでentrypointをloader/pluginから生成できる!



Webpackの自作loader/plugin開発は
Compilationで決まり!!!1



応用

これを使えばHTML内でJavaScriptをインラインで展開できたりしちゃう
-> HTTPリクエストが1個減ったすごい!!!1*

* インラインなスクリプトは同期的にパース、実行されます。また、HTMLを肥大化させることは全体的な遅延の増加を引き起こします。

```
11,2 index: 1}.crossfade element  
{visibility:hidden}.crossfade-visible{z-  
index: -2}</style> <title>TsuboneSystem</  
title> </head> <body> <div id=container>  
<script>!function(e){function t(i){if(n  
[i])return n[i].exports;var a=n[i]={i:i,l:!  
1,exports:{}};return e[i].call  
(a.exports,a,a.exports,t),a.l=!0,a.exports}  
var n={};return t.m=e,t.c=n,t.i=function(e)  
{return e},t.d=function(e,n,i){t.o(e,n)||  
Object.defineProperty(e,n,{configurable:!  
1,enumerable:!0,get:i})},t.n=function(e){var
```




実演

```
<!DOCTYPE html>
<html>
  <head><style>${require("./loader.js?css!./index.js")}</style></head>
  <script>${require("./loader.js!./index.js")}</script>
</html>
```

<https://github.com/kagucho/entry-webpack-plugin/>



課題

- アレなAPIを使っている
webpack-contribのloaderも同じことをしているし正規のAPIを用意すべきでは？
- 複数のentrypointに正しく対応できない
- それでもやはりWebpackはJavaScriptへのバンドリングが基本
Webpackという割にはWebページ全体を構築するツールとしては筋が悪い. バンドル先 (JavaScript, CSS, HTML, etc.) に応じて形を変えるような仕組みが必要.
- つかバンドリングツールでHTMLを扱う必要って本当にあるの？ バンドリングツールなんかで凝ってどうするんですかねえ.



ご清聴ありがとうございました!
語れ!



参照

- webpack/webpack: A bundler for javascript and friends. Packs many modules into a few bundled assets. Code Splitting allows to load parts for the application on demand. Through "loaders," modules can be CommonJs, AMD, ES6 modules, CSS, Images, JSON, Coffeescript, LESS, ... and your custom stuff. (2017年2月16日閲覧.)
<https://github.com/webpack/webpack>
- Loader API (2017年2月16日閲覧.)
https://webpack.js.org/api/loaders/#_compilation
- tsubonesystem3/js.js at master · kaguchio/tsubonesystem3
(JavaScriptのentrypointをインラインするloader. 2017年2月16日最終更新.)
<https://github.com/kaguchio/tsubonesystem3/blob/master/webapp/loader/js.js>



利用許諾

私, 小田喜陽彦はこのスライドの利用をを以下の文章に基づき許諾します.

Creative Commons — Attribution-ShareAlike 3.0 Unported — CC BY-SA 3.0

<https://creativecommons.org/licenses/by-sa/3.0/legalcode>

Webpackのロゴは次の文書に沿った利用が許諾されています.

media/LICENSE at 37b4686e9e6b4725813167e9e5605464870d0ecb ·

webpack/media

<https://github.com/webpack/media/blob/37b4686e9e6b4725813167e9e5605464870d0ecb/LICENSE>