

Q

A

JavaScript情報っ
てなんだっけ？



「情報」って何？

「情報」って何？


『知る』ということの実体化。われわれが、あるものについて『知る』
ということは、何かしらの得たこと、何かを頭の中に取り込んだこと
である。その『何かしら』を、われわれは情報と呼ぶのである

— 高橋秀俊



なんで情報が欲しいの？

何で情報が欲しいの？

- >> 新しい技術が面白そうだから
- >> 開発で楽しみたいから
- >> 話題だから
- >> 選択肢を持ちたいから 





情報って既に沢山
あるのでは？



ありすぎるのが
逆にこまる

情報が多いと何が問題な

ってくるのか

少し過去を振り返る

2011年

世界のJavaScript情報を読もう

>> 古いものと最近のものが混ざって見つけにくい

Be the first to clip this slide

現在を知るためには...

- 現実的な問題
 - JavaScript情報が溢れすぎ
 - 古い情報が混じっている
 - jQuery語がじわじわと
- ごちゃごちゃしてる中から見つける必要性

• 情報の取捨選択が必要になる



2012年

































世界のJavaScriptを読もう @ 2012

- >> 古いものと最近のものがまざってる
- >> 最近のものを整理するWeekly、ニュースサイトが増えた

乱立するまとめサイト

- ブラウザのアップデート頻度が増加
- Webの変化が急速化
- Webの情報量の増加
- SBMがまとめ役を保てなくなった
 - まとめ役が求められた
 - => Weekly系、小さくまとめるソーシャルニュースサイトが乱立

Frequency of Update
Daily
Weekly
Monthly
Other
Github
View on Github
View on Github

Website	Updated	Keywords	Twitter
JavaScript Weekly 	 weekly	JavaScript Library Tools	@JavaScriptDaily
Dart Weekly	 weekly	Dart JavaScript	
HTML5 Weekly 	 weekly	HTML5	
Node Weekly	 weekly	Node JavaScript	
Web Tools Weekly 	 weekly	HTML CSS JavaScript Tools	@WebToolsWeekly
Open Web Platform Daily Digest 	 daily	Web Spec HTML CSS JavaScript	@simevidas
JS Central	 other	Twitter JavaScript	@jscentral
The Morning Brew 	 daily	Node JavaScript	
JSter 	 monthly	JavaScript Library	
Web Design Weekly 	 weekly	Design CSS	@wdweekly
angular-js.in 	 daily	AngularJS JavaScript	@ng_directives
ng-newsletter	 weekly	AngularJS JavaScript	@ngnewsletter
The WebKit Blog 	 weekly	Webkit Safari Browser	
JSer.info 	 weekly	HTML CSS JavaScript Browser Tools	@jsr_info
JavaScript Jabber 	 weekly	JavaScript Podcast	@JSJabber
CSS WG Blog 	 weekly	CSS Spec	
W3C News 	 other	Spec	@W3C
EcoJS 	 daily	JavaScript HTML5 Library Tools	@ecojs

Meta Weekly

2014年

世界のJavaScriptを読もう @ 2014

- >> 急速にライブラリの数が増えてきた
- >> ユーザーがライブラリの選択で悩むようになってきた
- >> 現在に対する選択肢の増加

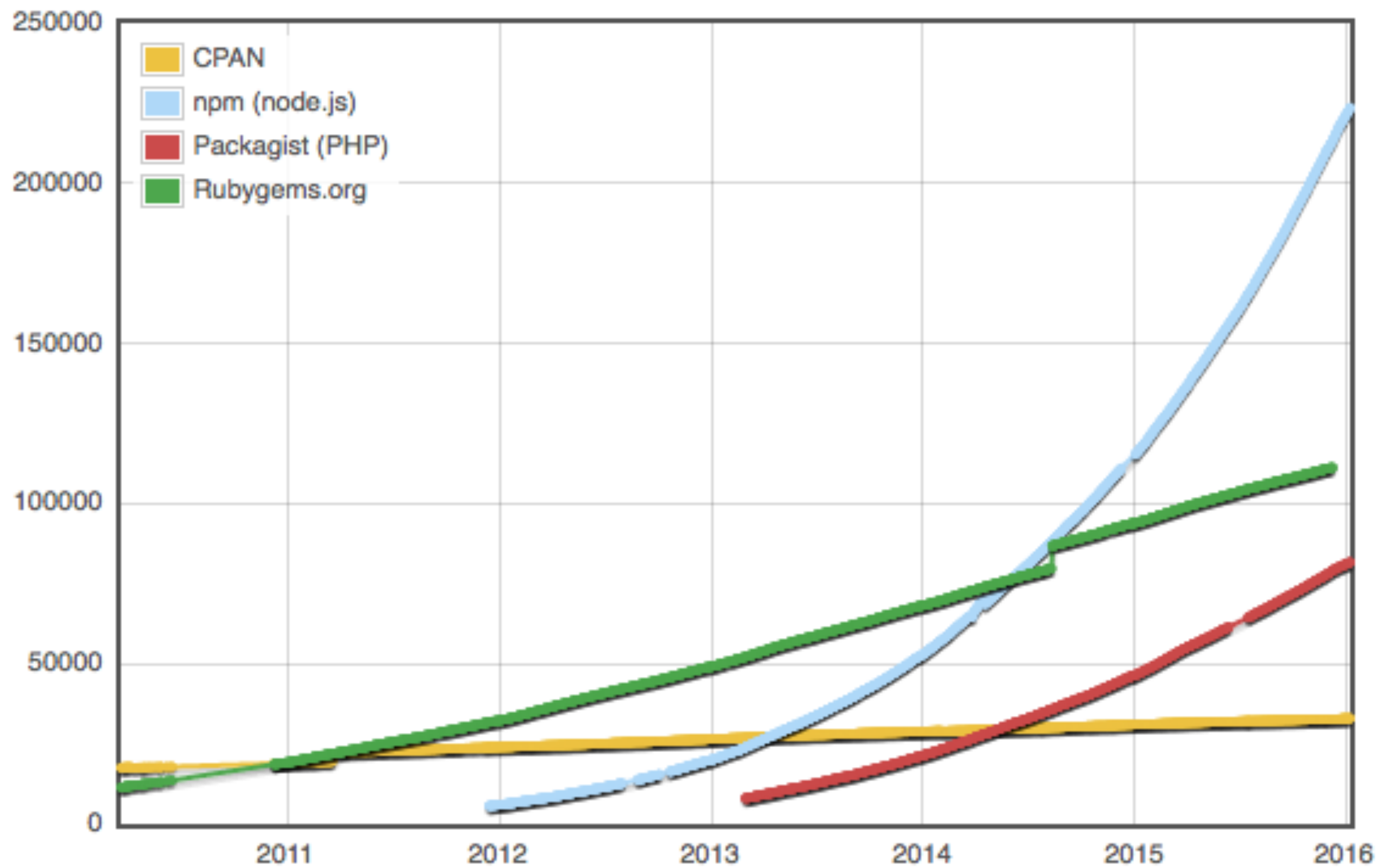
ウェブの変化の早さ

- ウェブの変化に対応するためにブラウザのリリース速度は上がった
- ライブラリも変化/トレンドに対応するためリリース速度は上がってる

ライブラリ数の推移



Module Counts



2015年 - 去年

- >> 古いものと最近のものがまざってる
- >> 現在に対する候補が多い
- >> 未来に関する悩みが増えた ★
- >> ブラウザの情報が整理された



未来に関する悩み
って？




未来に関する悩み？

- >> 不確実な新しすぎるものが増えた
- >> Transpilerによって未来の仕様が身近になった

例えば、将来


- >> ECMAScript Proposalの機能が全て使えるわけではない
- >> cssnextにある機能が全て使えるわけではない



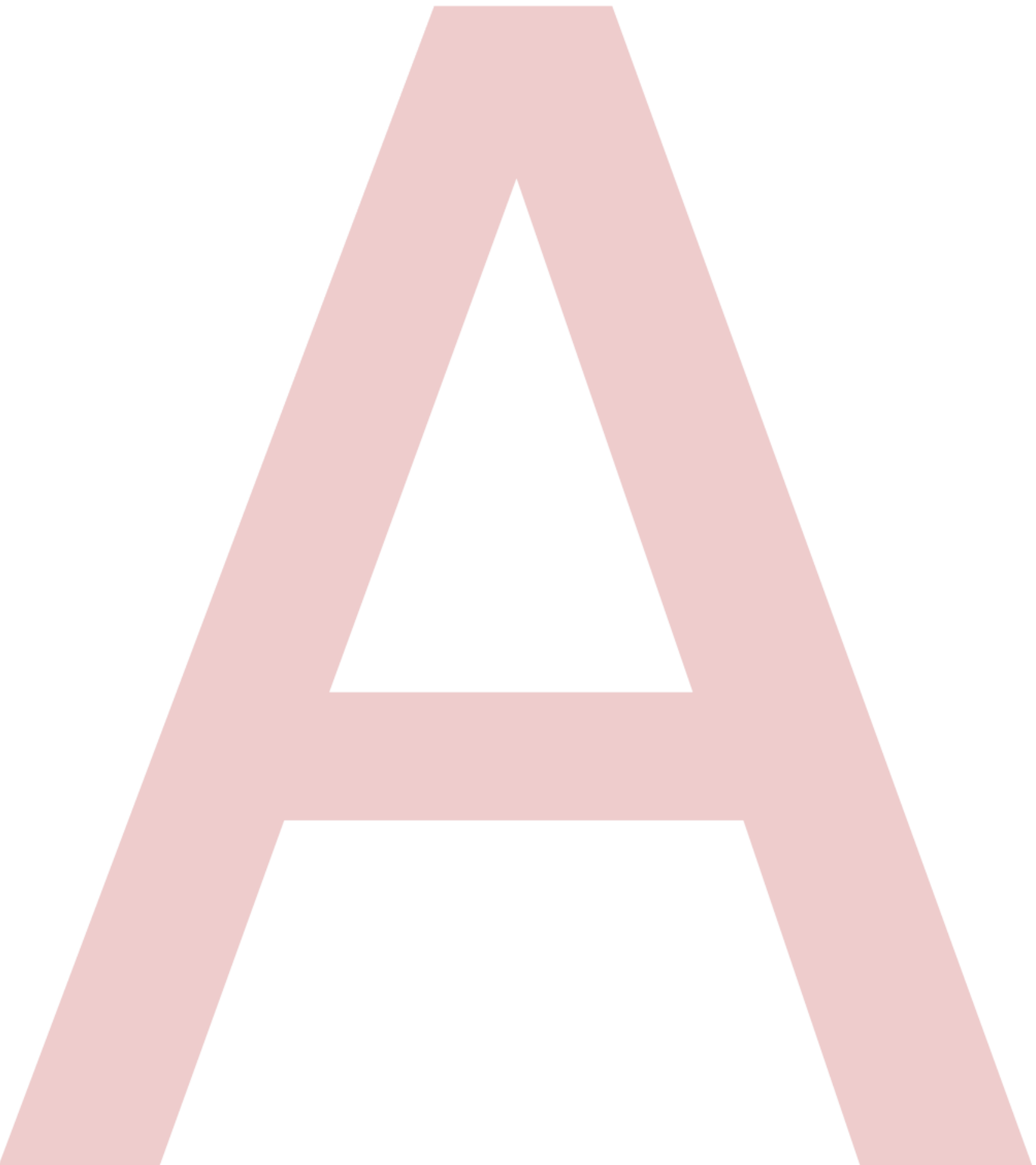
未来の悩みって今
悩まなくてもいい
のでは？

未来と現在の混同

- >> BabelがES6の仕様だと思ってる人がいる
- >> AngularJSやReduxのように不確定な仕様を推奨するものがある
- >> 🚀 安定した仕様とそうでないものを見分ける必要がある



安定した/していない
仕様の見分け方
は？

A large, stylized, light pink letter 'A' is positioned on the left side of the image, serving as a background element.

仕様の策定プロセス
を知ろう

仕様 – ECMAScript

- >> 大きく分類

- >> **Stable** – ECMAScript 20XX

 - >> ES6、ES2016... 1年ごとリリース

- >> **Unstable** – Proposal

 - >> 仕様のProposal、ドラフト

仕様 – Proposal

- >> ES 2016からは機能毎のProposal
 - >> ES.nextと策定プロセス | ECMAScriptとは何か？
- >> Proposalには5段階のStageがある
- >> Stage 4以外は基本的にUnstableと考える

仕様 - Stage


- >> Stage 0 Strawman - アイデア
- >> Stage 0.5 🚀 - Championが付きアイデア
- >> Stage 1 Proposal - 提案
- >> Stage 2 Draft - ドラフト
- >> Stage 3 Candidate - 仕様書と同じ形式
- >> Stage 4 Finished - 策定完了 => 仕様書へマージ

Stage 0.5 🚀

- >> (公式用語ではない)
- >> Stage 0 ProposalsにはChampionが付いてるかどうかの違いがある
- >> Champion 🚀 : 仕様を最後まで進めるTC39のメンバー
- >> 例) BabelはChampion 🚀ついてないStage 0は実装しない
 - >> New plugin for pipeline operator proposal #3159

仕様の策定プロセスまとめ

- >> ECMAScriptはProposalを見れば安定したものなのか分かる
 - >> tc39/ecma262
- >> まだProposalなもの = 安定してない
- >> 仕様に入ったもの = 安定してる



仕様っていつブラ
ウザでいつ使える
の？



ステータスページ
を見よう

ステータスページ

>> [azu/browser-javascript-resource](#)

>> 主要なブラウザ(エンジン)のステータスページが用意された！

>> [status.modern.IE](#)

>> [Firefox Platform Status](#)

>> [Chromium Dashboard](#)

>> [WebKit Web Platform Status](#)



Browser Platform Status Tracker

2016-01-08

Chromium

- **Changed:** Reflect (ES6)
Status: In development -> Behind a flag
ShippedAndroidMilestone: (n/a) -> 49
ShippedMilestone: (n/a) -> 49
ShippedOperaAndroidMilestone: (n/a) -> 36
ShippedOperaMilestone: (n/a) -> 36
- **Added:** addEventListenerOptions.rssiv
Status: In development

2016-01-07

Chromium

- **Changed:** WebAssembly
Status: Proposed -> In development
- **Changed:** Deprecate some nonstandard uses of RTCPeerConnection legacy methods.
Status: Enabled by default -> Deprecated
ShippedMilestone: 23 -> 49
ShippedOperaAndroidMilestone: (n/a) -> 36
ShippedOperaMilestone: (n/a) -> 36
- **Changed:** Destructuring (ES6)
Status: In development -> Enabled by default
ShippedAndroidMilestone: (n/a) -> 49
ShippedMilestone: (n/a) -> 49
ShippedOperaAndroidMilestone: (n/a) -> 36

Browser Platform Status Tracker



じゃあ今はどうすれば
いいんですか？



そもそも何でこんなにライブラリって多いんですか？


- >> 標準ライブラリがない(言語自体が提供するものは少ないデザイン)
- >> UIViewControllerやActivityのようなライフサイクルの元がなかった
- >> UIKitのようなUIデザインの元が薄い
- >> 色んな環境で動く



どうやってライブ
ラリを選ぶの？

>> ここで必要になるのが情報

>> やつと本題



ライブラリの生死
ってどうやって見
るの？

>> pluse



全てをカバーする
ライブラリはない
よ

- >> その選ぶ手段を手に入れたいから
- >> 何かしらの基準と情報を照らし合わせる
 - >> 基準は様々だね



基準ってどういう
ものがあるの？

>> issue


>> リリースプロセス

>> コミュニティ




コミュニティって
図れるものなの？

- >> とっても難しい
- >> そもそもコミュニティが難しい
- >> 多くなるほど難しい
- >> CODEOFCONDUCTやポリティカル・コレクトネスな問題も最近が多い



じゃあ結局どうや
って選択すればいいの？


>> 一つでダメなら複数で



情報って事前に集
める必要がある
の？

>> いいえ

>> 集め方の方が大事だと思います。




結局、詳しい人に
聞けばいいので
は？

>> 正解

>> 会場にいるひとに聞いてみて

>> 誰が何に興味を持っているかって大事



選択できても1年
後には変わるんで
しょ？

>> はい



エコシステム？

コミュニティから標準へ

結局全部つながってるん

じゃね

終わり