Visualize TC39 Process

TC39によるECMAScript策定プロセス

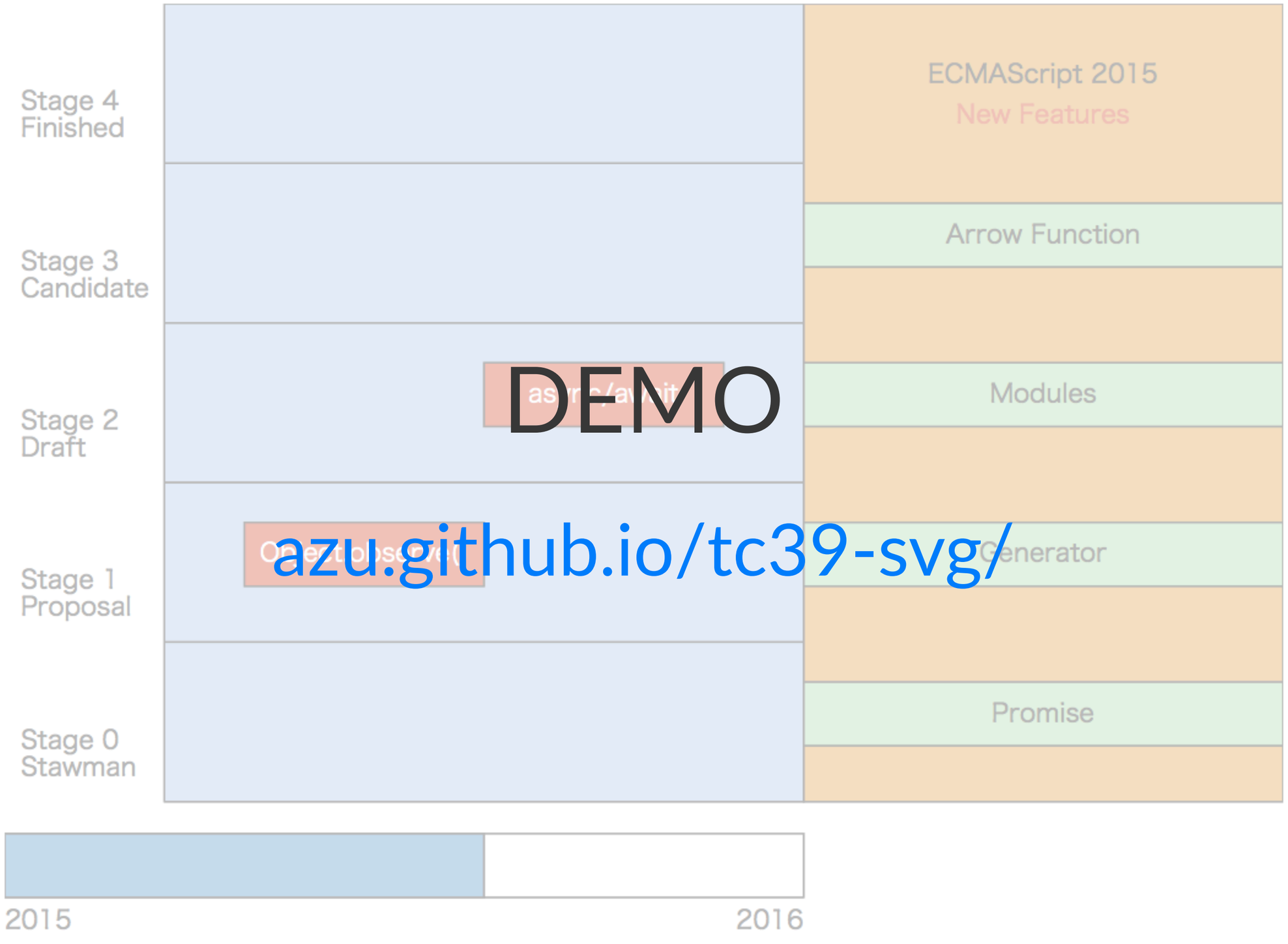
詳しくはTC39 Process - Google ドキュメントを参照

TC39 Process 🚃

Stageは5段階

- 0 Stawman
- 1 Proposal
- 2 Draft
- 3 Candidate
- 4 Finished

毎年 3月と9月にstage4なったものを取り入れたものを新しい


Stage 4
Finished
Stage 3
Candidate
Stage 2
Draft
Stage 1
Proposal
Stage 0
Stawman
ECMAScript 2015
New Features
Arrow Function
Modules
Generator
Promise
DEMO
azu.github.io/tc39-svg/
2015
2016

Stage 0: Stawman

- アイデア

Stage 1: Proposal

- プロポーサルの目的や解決方法を示す
- Polyfillやデモ等を用いて解説する

Stage 2: Draft

- いわゆるドラフト
- ECMAScript標準と同じルールでAPIや構文、セマンティックについて説明していなければならない
  - 仕様書と同じ記法でプロポーサルを書き直す

Stage 3: Candidate

- 設計は完了した状態
- 実装者からフィードバックを求めて改善をする状態
- レビューアはその仕様策定者以外ならだれでもなれるが専門的な知識を持っている必要がある

Stage 4: Finished

- ECMAScriptへ取り込まれる準備が完了したことを示す状態
- 2つの実装がテスト(tc39/test262)をパスしてる状態

Stage up

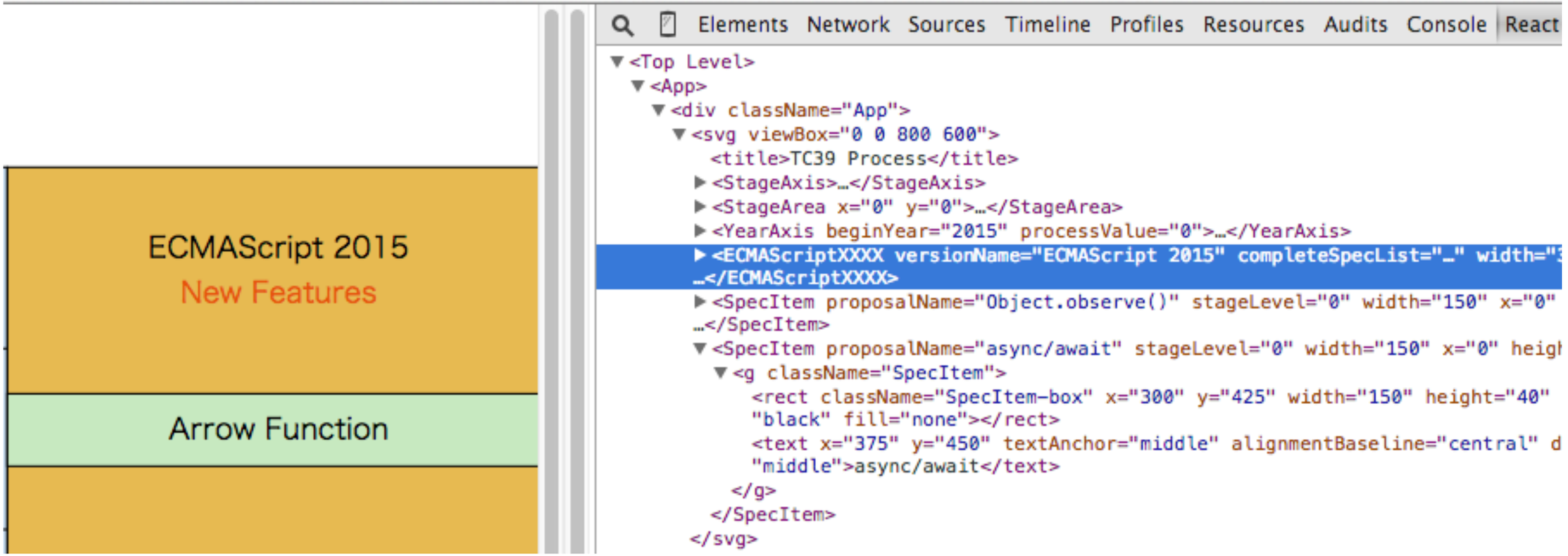
- Stage++するかは、2ヶ月ごとのTC39のミーティングで決まる
  - tc39-notes
- 逆算すれば、Stage 0のものがStage4になるまで8ヶ月かかる
  - Stage 1から始めるものもある

ES2016(想像上の話)

- おそらくES2016までにStage 4となるものはない
  - tc39/ecma262
- ES2015 == ES2016
- ES2016 == ISO/IEC 16262:2016
- 多分、perhaps

azu.github.io/tc39-svg/の中身

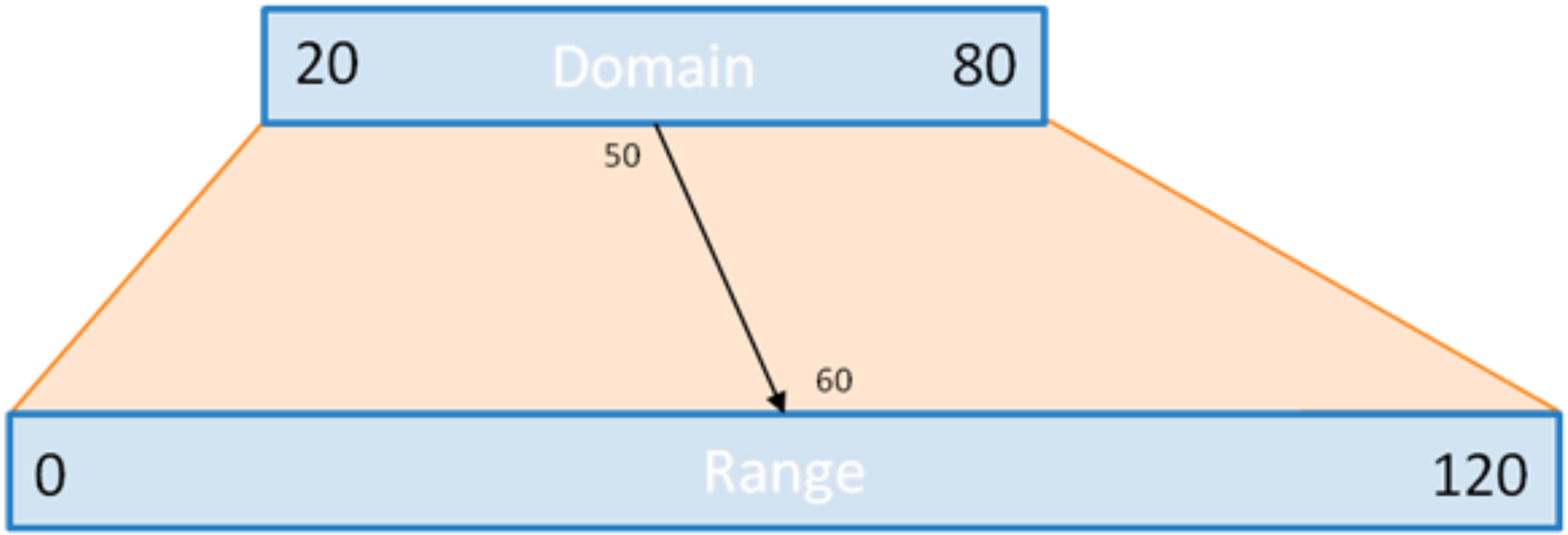
SVG + React


ECMAScript 2015
New Features
Arrow Function
Elements Network Sources Timeline Profiles Resources Audits Console React
<Top Level>
<App>

<svg viewBox="0 0 800 600">
<title>TC39 Process</title>
<StageAxis>…</StageAxis>
<StageArea x="0" y="0">…</StageArea>
<YearAxis beginYear="2015" processValue="0">…</YearAxis>
<ECMAScriptXXXX versionName="ECMAScript 2015" completeSpecList="…" width="
…</ECMAScriptXXXX>
<SpecItem proposalName="Object.observe()" stageLevel="0" width="150" x="0"
…</SpecItem>
<SpecItem proposalName="async/await" stageLevel="0" width="150" x="0" heigh
<g className="SpecItem">
<rect className="SpecItem-box" x="300" y="425" width="150" height="40"
"black" fill="none"></rect>
<text x="375" y="450" textAnchor="middle" alignmentBaseline="central" d
"middle">async/await</text>
</g>
</SpecItem>
</svg>

SVG on React

- SVGの要素をReact Componentに書ける
- 縮尺の問題はd3-scaleをコンポーネントに渡す



React + d3-scale

- コンポーネントがステートレスになる
- 幅、高さ その領域の相対座標計算機(range)を渡して描画

```
var ECMAXRange = linear().domain(TenPercent).range([500, 800]);
var ECMAYRange = linear().domain(TenPercent).range([0, 500]);
<ECMAScript versionName={`ECMAScript ${this.state.beginYear}`}
    completeSpecList={completeSpecList}
    width={300} height={500}
    xRange={ECMAXRange} yRange={ECMAYRange}/>
```

A Proposal

- ES nextの策定プロセスを分かりやすくまとめた記事・Issue #57
  ・azu/azu