自己紹介

- Name : **azu**
- 🐦 Twitter : @azu_re
- 📝 Blog: Web scratch, JSer.info
- 📚 Book: JavaScript Primer
- 🏗 Develop: textlint, secretlint, HonKit

OrbStack

- OrbStack · Fast, light, simple Docker & Linux on macOS
- Drop in replace Docker for mac
- 早い、ファンを回さない、便利
- GUIがまとも

OrbStackの特徴

- Portを開けなくてもContainerに直接アクセスできる
  - Container側portをexposeしなくても、macOS側からアクセスできる
  - `node --inspector`のようなデバッガーを外から接続するのに便利
- macOS側にOrbStackのContainer/ImageがNFSとしてマウントされる
  - Finderで普通にImageの中身などを確認できる
- See OrbStack 1.0: Fast, light, easy way to run Docker containers and Linux

Docker for macOS to OrbStack

- v0.15.0からDocker for MacのImageをまるっと移行できるようになった
  - "Migrate"ボタン押すだけで終わる
  - 事前にDocker for macは最新にしておく方が良い
  - Docker Desktop使いがOrbStackでハマったところ
- `/var/run/docker.sock`の参照先を切り替えることでDocker for Macとは共存可能(一応戻れる)

OrbStackとSpeed

- Public Betaあたりから触って、v0.13.0 (6月25日) あたりでDocker for Macを使うのをやめた
- バグ報告をすると、素早く反応して修正してくれる
  - 主にRosetta側のバグにぶつかったことが多かったけど、なんとか回避する修正をしてくれていた
- 素早い反応ができるプロダクトは、期待値が高くなりやすい

Fast First Response

Bun

- Drop-in replacement for Node.js
- コアメンテナーのJarred Sumnerは、反応から修正までが早い
- 書いてるコード量も多い

Bunに報告したIssue

- `bun build --compile` cause segmentation fault · Issue #3764 · oven-sh/bun
  - 2分で反応、15分で修正
- v0.7.2+ regression: TypeError: undefined is not an object (evaluating '_*classPrivateFieldGet(this, _Client*fetch).call') · Issue #4029 · oven-sh/bun
  - 1分で反応、2時間で修正
- `bun run <missing.ts>` should exits with non-0 · Issue #4011 · oven-sh/bun
  - 10分で反応、1日で修正

バグと素早い反応

- Bunのバグはリグレッション的なものが多かった
- 何度も起きると困るけど、ちゃんと報告するとちゃんと直してくれるのでバグ報告がしやすい
- オープンソースでは多いが、素早くレスポンス返せるプロダクトは成長しやすい

6to5

- 6to5: Babelの元の名前
- 最初に6to5が公開された時にバグを見つけたので、報告したらすぐ修正された
  - 6to5-node throw error · Issue #9 · babel/babel
- 当時のメンテナーであるsebmckは、素早いレスポンスと異常なコード量を書いていた

Fast First Response

- 素早いレスポンスができると、期待値が上がる、成長しやすい
- 一方で成長すると、素早いレスポンスが難しくなる

なぜ?

- スタジアムモデル[working in public]
- 多くのプロダクトはユーザーは増えても、Contributorは線形には増えない

working in public Stripe Press — Working in Public

メンテナーが利用可能なAttentionを管理するパターン

- 初期費用を減らす: CI/Bot/Lint/Check List
- 利用可能なAttentionを制限する: Issueを閉じる
- コストの分配: モデレーション、ユーザーサポートを任せる
- 利用可能なAttentionの総量を増やす: コントリビューターを増やす、収益を増やす

内発的動機と外発的動機

- 多くのプロダクトのスタートは内発的動機から始まる
- また、成長中は評判や認識という形で外因性の報酬を受け取ることある
- 一方でプロダクトが成熟すると、評判のメリットがなくなっていき(増えなくなる)、義務感やコミュニティの側面が強くなっていく
- 内部と外部のバランスが変わっていく

🔥 BurnOut

- バランスが崩れていくと、BurnOutしてしまいやすい

  — ~2015 in review. This started off as a generic year in… | by Sebastian McKenzie | Medium

まとめ

- 素早いレスポンスは、期待値を上げる
- 成長により鈍化して見えるのは、バランスの変化
- バランスの変化を無視すると、BurnOutに繋がりやすい

参考

- Stripe Press — Working in Public
- bradfitz/issue-tracker-behaviors
- CMUSTRUDEL/toxicity-detector