

SaaSによるアジリティ獲得



- ・ エンジニアの役割変化
 - ・ 個別にオンプレ業務システムをカスタマイズしていたSlerの仕事は減っていく
 - ・ 情シスは、SaaSによるxOpsを支援する新しい役割へ
 - ・ 「SaaS Ops (仮)」
 - ・ 今回は詳しく話さないが重要

ここまでのまとめ

- ・ DXとは、従来型企業が「デジタルネイティブな組織」に変わること
 - ・ そのためには、「ITを活用してアジリティを上げること」が重要
- ・ SaaSの登場で、非エンジニアだけでシステム構築が完結できるケースが増える
 - ・ 様々な業務のアジリティを上げる
- ・ つまり、DXの第一歩は「SaaSを活用してアジリティを上げること」

参考: SaaSを使うことがDXの第一歩

もっとシンプルな話だと私は捉えていて、たとえばSaaSを使うことが、DXの第一歩だと思います。

DXを本気で進めたいのなら、デジタルのスピード感覚に慣れ、デジタル的な感覚のビジネスリテラシーをもって意思決定し、トライ&エラーを繰り返して学習する組織や意思決定できる構造を、ちゃんと根付かせることのほうがよほど重要だと思います。

「日本のDX、難しく考えすぎでは? LayerX代表 福島氏が語るDX推進の処方箋」より

- ・ 現場メンバーが自らSaaSの導入や運用を経験することで、デジタルを前提とする仕事の仕方に少しずつ慣れ、それが徐々に組織文化として根付いていく

SaaSと利用者の間にある「2つのギャップ」



- ・ **エンジニアリングスキルのギャップ**
 - ・ 多くのSaaSは汎用性が高く、業務に合わせる部分は利用者に委ねられている
 - ・ しかし、利用者の多くは非エンジニア
- ・ **業務知識のギャップ**
 - ・ 多くのSaaSには、業界や職種の業務ロジックが埋め込まれている
 - ・ しかし、実際の業務は日々変化する

補足: NoCodeという幻想

- ・ 全てのSaaSは「NoCode」の要素を含んでいる
- ・ 非エンジニアでもシステムやアプリケーションの構築ができる
- ・ 誰でも簡単に操作できるツールであっても、それを使って複雑な現実に対処するには相応のスキルが必要
- ・ NoCodeとは「超高水準プログラミング言語」でしかない
- ・ GUIエディタとリッチな関数を備えた "NoCode" という言語を使ってCodeを書いているに過ぎない

