

インフラから見たDevOps

8/19 ODC

野口 洋明

devopsが産まれた理由

もっと早くリリースをしたい

部署が違くとチケットや工数・予算確保の問題が出て来る

専属ではないので依頼をしても他のプロジェクトの事情が優先されてしまう

課題があっても目の前の依頼が優先されがち

部署間の諍い

工数管理でチケットでオーダーしろと面倒

まず devops の定義をしなくてはならない

単純には運用と開発を
一緒にすることだが。。。。

定義を調べていてなるほどと思った
devopsを一番簡単な表現をするのは

業務上の組織が障壁にならないよう回すもの

つまりdevopsは文字の綴りを超えた
会社組織そのものを定義している

会社変えるなんて無理

と言っても仕方ないので経験を元に話を

愚者は経験に学び、賢者は過去に学ぶ



いや、俺そんなこと言ってないよ

「愚者だけが自分の経験から学ぶと信じている。
私はむしろ、最初から自分の誤りを避けるために、
他人の経験から学ぶのを好む。」
(オットー・フォン・ビスマルク)

過去に学ぶ賢者にみんななれない。
過去を尊重する人は2割いればいい方
過去事例の方が説得力を与えられることもある

って書こうと思ったら同じ思いだった

1. うまくいっていたチーム@某ゲーム会社

タイトルごとに部署になっており

構築運用基盤は某ゲーム会社のものを使用

AWSでの自社タイトルだけ基盤から構築

成功理由

基盤が他社提供であったことからインフラ工数がそもそも少ない

技術者も半数が積極的に会議で話す人であり

会議はいつも長かったが実りがあった

技術力がある人間がそれなりにいたため課題解決がなった

2. プログラムとインフラチームが同じ部署だけど別チーム@とある狭い業界の会社

チームで会話していて部署として会話は共有という形

また、フロント側の開発が別部署、別の階だったため

コントロール下に置けず不思議な事も多かった。

そもそもインフラチームが無口な人が多かった

過去の遺産も、、、

失敗理由

一見同じ部署でもチームが違えばdevopsにはならない

ここがdevopsになるには長い年月が必要でしょう。

一人ブリッジで空回り気味の人がいたので彼の頑張り次第？

3. プログラムとインフラチームが別部署@某Web会社 前期

プログラマ2名とインフラエンジニア(私)が食事をよくしていて

お互いのスケジュールや悩みを相談

部署が違ったがブリッジがしっかりできていたので

うまく回っていると上司も認識していた

また、基盤が古く課題があったものの

それはインフラチームが計画を練って改修する予定だった

前期？当然後期があります

うまく回っていると認識していた上司が

とある人物の提案でdevopsチームということに

4. プログラマとインフラが同じチーム。ただしプロジェクトごとではない@某Web会社後期

全く回らなくなった

理由1。課題だった基盤をやる部署がなくなり誰がやるのとなった

理由2。プロジェクトごとに分けなかったためにリーダーが工数調整にひたすら追われた

理由3。障害が増える一方のため障害対応チームという犠牲者が出る仕組みがあった

理由4。うまく回らないのを予見していた提案者がブリッジをしなくなった

色々早すぎた。そしてもう一つ

色々早すぎた。そしてもう一つ

上が構造を変えても積極的にそれを支えようとする人がいないdevopsは失敗する

devopsの良さは人が変わらないと出てこない

devopsは上も下もやろうと一致しない限り
決してうまく回らない

逆をいうと献身的な人間がいれば
2.みたいに回ることもある
とは言っても今のうちじゃできないよ
という思いがある人

今私は挑戦をしています

5. プログラマとインフラが別部署、プログラマの部署がフリーランスのインフラエンジニアを雇用

理由:

部署が違うので完全にはうまく回りきっていない感がある

インフラチームは他のプロジェクトを見なくてはいけない

メイン担当はいるが専属ではない？一人では工数が足りないのはどうする？

などの諸々の問題解決に

インフラエンジニアなのにしゃべる自分が試験雇用された

求められていること

インフラチームとのブリッジをしつつ、

プログラマの課題を解決する

このコンテンツをよくすることだけを考えてます

本職のインフラでなくてもブリッジもできるプログラマを雇うことで

devopsの連携を他に見せつけ会社の仕組みが変わればそれは素晴らしいこと

話は戻って
つまりdevopsに必要な人間は壁をなくせる人

クラウドでLAMP構築で十分なんて環境は多い
それって今時PGだって半数はできる
ならインフラエンジニアは何をするのか

ここではコードに関わる仕事を中心
ではない人をインフラエンジニアと定義

ならインフラエンジニアがブリッジすべきで
はないのか

クラウドでインフラが楽になった分
Devopsではブリッジエンジニアを今後求められ
優秀なインフラエンジニアに求められるものと違う

ぶっちゃんけ仕事はどんどんなくなっていく

今後インフラエンジニアに新しく求められること

プログラマとクラウドの橋渡し(クラウド向きな設計とか操作教えるとか)

devopsチームに所属していて基盤チームとの橋渡し

プログラマが悩んでいる時に積極的に相談しプログラム以外でのアプローチ提案

他の部署との折衝

残念ながら無口な人間には向かない
そういう人は技術力を高めるしかない

優秀なインフラエンジニアなら求められるものと違う

基盤チームとdevopsな人間は分けないとい
けない

ん？基盤チーム

基盤チームの役割

各devopsチームが使う基盤整備、例外ルール作成

優秀な人間が研究しそれをブリッジエンジニアに展開する

また、ブリッジエンジニアが困ったときに専門家として手助けできる体制を作る

たまにブリッジエンジニアを基盤チームに編入し技術力の底上げを図る

新人を教育し最低限の技術力をつける

残念ながら事務作業も

優秀な人を便利屋みたいな役回りをさせるべきではない

なので基盤チームとdevopsな人材は別に分けるべき

基盤チームでもプログラマがいるのではないか？

優秀な人達の集団ではブリッジするだけの人間はすぐに限界が来る

技術力がある人間は基盤ないしは大規模、
研究に配属されるべき

最後にプログラマの方は比較的社交的なインフラエンジニアと一緒に食事をしましょう。
インフラエンジニアはブリッジを目指すか一つのこと
で登壇するほど詳しくなりましょう

チケットでタスクをやり取りするのではなく
コミュニケーションでタスクをやり取りする

一旦終わり

もちろん管理する側のチームビルディングも大事

ドクトリン

一人の人間が完全に管理できるのは自分
含め5-8人？平均7人

理想は7人のチームで区切る

となると6人のリーダーと管理職で43人が
チームとして機能する限界ではないか

最後にもう一度

Devopsの目的はあらゆる壁をなすこと