



NSO技術者の育成について

CTC 渡部

NW 自動化

- 渡部友也(わたなべともや)
 - 東京都足立区出身
 - 伊藤忠テクノソリューションズ所属(通称CTC)
 - 2010年より現職で通信キャリア様を担当するエンジニア
 - NSO歴は2014年～なので10周年です
 - NSO連携Orchestrator開発
 - NED開発
 - BrownField Service Discovery



- JANOGでも活動しています
 - JANOG48
[IoT-GWプロビシステムの裏側](#)
 - JANOG52
[NW自動化開発 × アジャイル/スクラム開発](#)
 - JANOG53
[StreamlitはNWエンジニアにとって Web開発の銀の弾丸となり得るか？](#)

NW自動化

• NW/インフラに付随する自動化関連開発チーム

- 自動化オーケストレータWeb(OpS)の設計・開発・維持管理
- 自動化エンジン(NSO, Terraform, Ansible等)のライセンス販売 + 操作用モジュールの設計・開発
- NWの検証自動化(Robot Framework)、出荷検査自動化開発および関連OSSの開発

Mission

Config EngineのAPIを統括する
Webオーケストレータ等の開発

NSOやAnsible/Robot Framework等の
モデル/PlaybookやAPI開発

Config Engineとの実機結合開発
NW検証の自動化(Robot Framework等)

Product



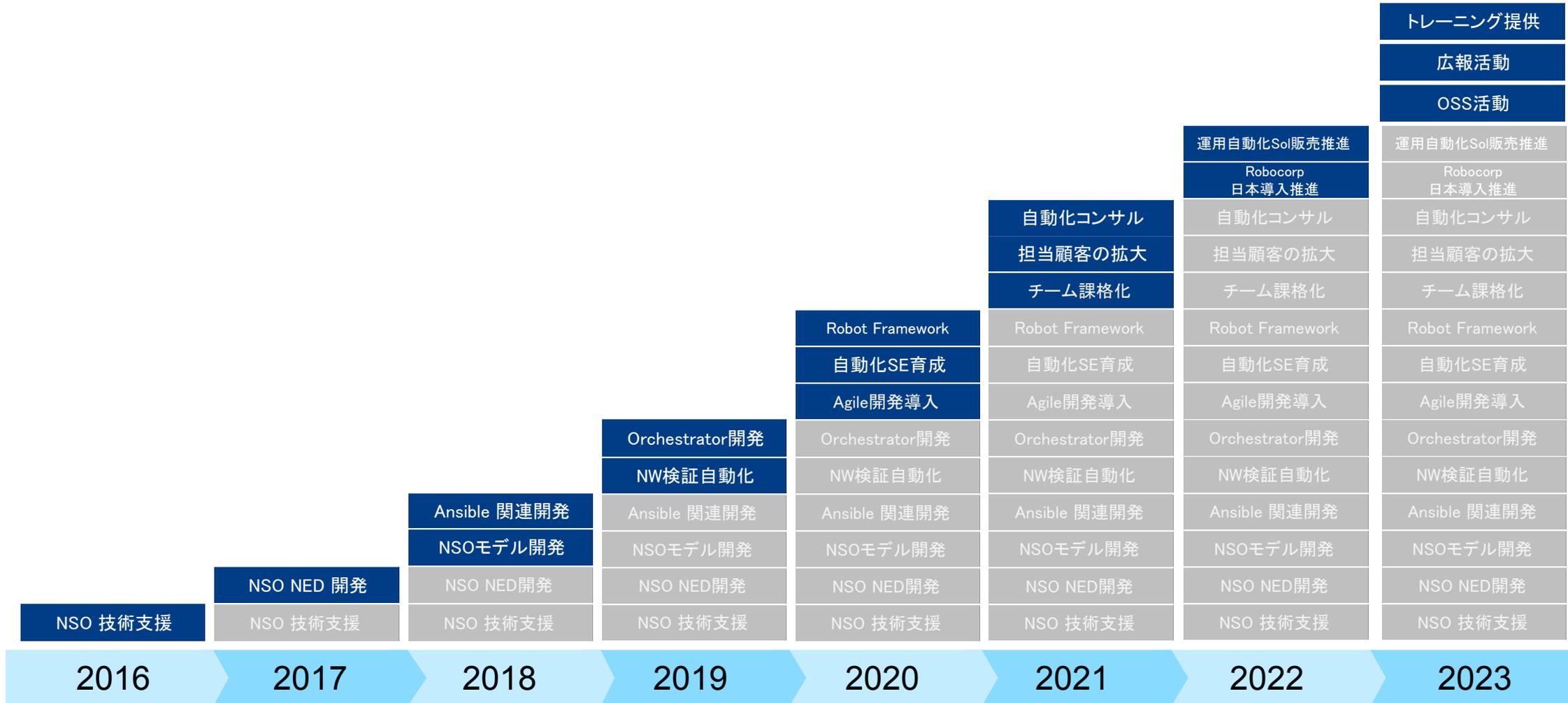
Value.2

要件整理支援および業務改善支援
OpS開発および工事業務のDX支援

コンフィグエンジンの技術サポート提供
自動化関連開発手法やナレッジの提供

マルチベンダーでの自動化
リグレッション試験の高度化

NW自動化



NSOだとこういうふうに
設計・設定すればいいよね



なににってるの？

なるほど、
NSOだから
こういう動作をするのか

だから、
なにをいってるの？

NW自動化

Why NSO is difficult?

- YANG
- CDB
- FastMap
- Backpointer/RefCount
- NED



- 概要

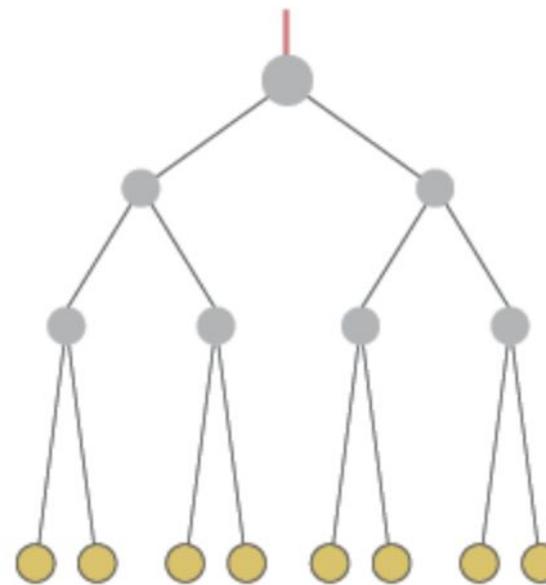
- RFC6020にて定義されるデータ定義言語
- ネットワークに関連する情報をモデリングするのに適している

- 課題

- ニッチな言語になってしまうので、ナレッジが検索してもあまりでてこない
- RFC6020を読むはめになってしまう

- 重要性

- ServiceModel/NEDに関連
- CDBを理解するために必要



- 概要
 - NSOのコアデータベースであり、XML/jsonベースのDB
- 課題
 - 多層に渡る意識を持つ必要がある
 - サービスモデルレイヤーのCDB
 - デバイスモデルレイヤーのCDB
 - 実機のConfig(CDBではないが・・・)
- 重要性
 - NSOはCDBがマスターデータであることの実感が必要
 - 運用するときには、より深い知識が必要になる



- 概要

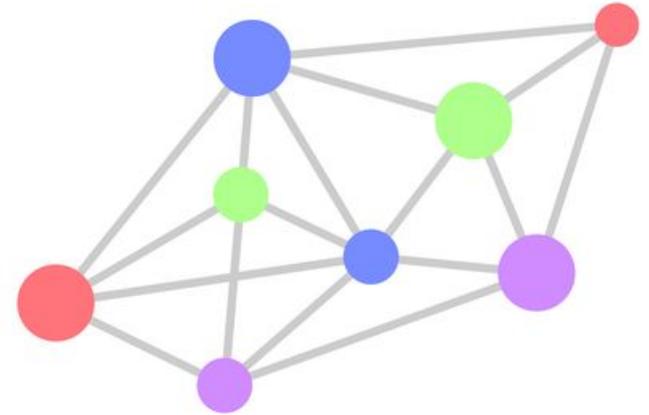
- CDBの内容から、実機に投入するConfigモデルを生成する仕組み

- 課題

- 動作仕様のDocumentは詳しいことは公開されていない
- CreateCallbackに基づくNSOの思想を理解する必要がある
- templateの動きや、diff-set等ほかにも必要な知識が多岐にわたる

- 重要性

- サービスモデルの開発時には挙動を理解して開発する必要がある
- 運用の際にも、設定を変更する際にはFastMapの動きを理解する必要がある



- 概要

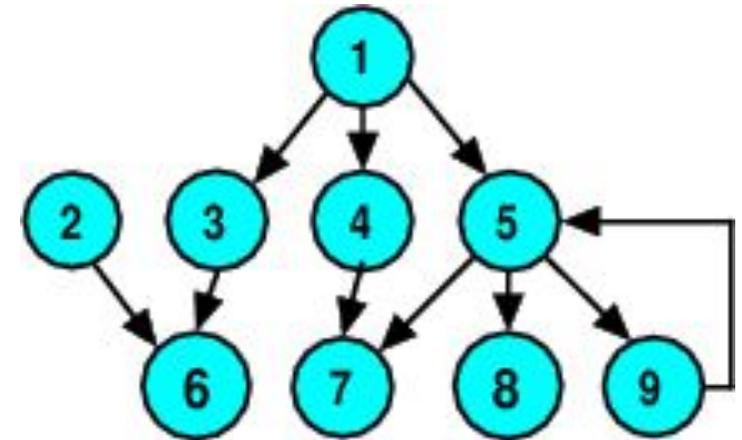
- サービスモデルで投入されたインスタンスとConfigの相関関係を持つ
- これらの値により、複数のインスタンスで共通利用されているConfigを消したり、消さなかったりという挙動の根幹にある仕組み

- 課題

- Documentはでてこない
- 意図しないrefcount等がついてしまい、reconcileする必要がある
- reconcileは万能ではないケースもある

- 重要性

- サービスモデルの追加開発・改修時に意識する必要がある



- 概要

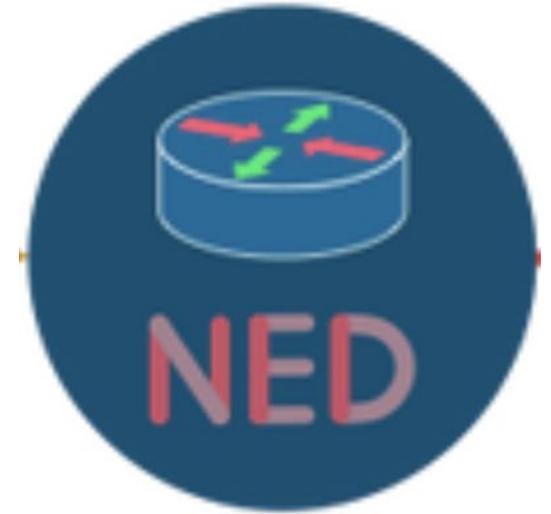
- NSOが装置を操作するためのDriver

- 課題

- 装置ごとに異なるNEDとなるため、若干の癖がある
- CLIだけでなくさまざまな種類がある
- NEDの挙動を理解しないと、トラブルシューティングできない

- 重要性

- そこまで理解することは難しくない。コードも見ることができる。
- ただし、全てに関連するもののため、最も重要な知識ともいえる
 - NEDから理解することが近道？

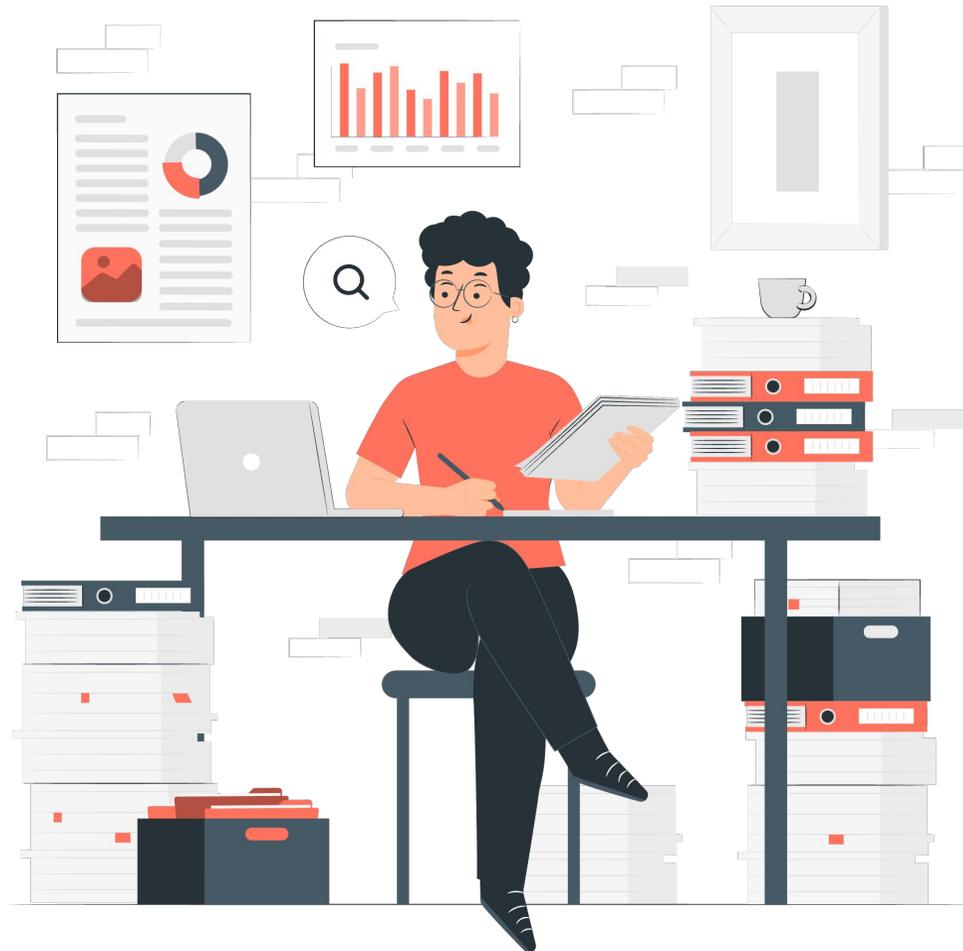


- maapi
- commit operation(no-networking/no-deploy/re-deploy/etc・・・)
- action
- Reactive FastMap
- premodification/postmodification
- servicepoint
- service discovery

- これらが絡み合っているので、すべてを理解して、初めて理解が進むものになる
- 点と点が線になって、面になるイメージ



- このほかに、そもそもNWのCLIやオペレーションの知識も必要
- それらを理解した上で、習熟コストを払うべき。それ以上の運用効果はある



- ただ、ニッチな技術なのでDocが本当に少ない…



- とりあえず、NSO解体新書というものを社内ナレッジで作りました
 - 作るのに1年かかりました・・・
- 課題
 - そもそも、内容がまだ足りてない
 - メンテナンスをCTC内部だけにするのは大変だし、効率が良くない
 - オープンなナレッジの蓄積の仕組みを作りたい
- 草案
 - GitHub Pages使って作りませんか？
 - 乗ってくれる人がいたら、言い出しっぺの我々がMaintainerやります

- みなさんどうしていますか

CTC

▼ *Challenging Tomorrow's Changes*

NW 自動化