

オンラインイベント

Clarif FileMaker Server 管理者が知っておきたい Infrastructure as Code

松尾篤

セッション ID :
O-G-15

 Clarif Engage Japan 2020

今回の話題

- Infrastructure as Code (IaC) とは
- Claris FileMaker Server for Linux
- サーバーの構成管理やテストを行うツール



デモ



デモについて

- サーバー構成管理ツール (Itamae) を使用したデモ
 - macOS Catalinaのターミナルアプリケーションで1つのコマンドを実行するだけでサーバーを自動構築
 - CentOS Linux 7がインストールされていてSSH接続できる状態になっているLinuxサーバーを用意
 - FileMaker Server for Linux 開発者プレビューをLinuxサーバーにインストール



Itamae



Claris Engage Japan 2020

～デジタルトランスフォーメーション～
2025年の崖を飛び越える



#ClarisEngageJP



松尾篤

株式会社エミック



自己紹介

松尾篤

- 株式会社エミック（東京都杉並区）代表取締役社長
- FileMaker 18 認定デベロッパ
- kintone認定 アプリデザインスペシャリスト
（2020年2月）
- kintone認定 カスタマイズスペシャリスト
（2020年3月）
- https://note.com/matsuo_atsushi





株式会社エミック

- 現在30期目
- Claris FileMaker 対応ホスティングサービス (FMプラン) を1998年から提供
- kintone導入支援・カスタマイズ開発
- <https://www.emic.co.jp/>

FMプラン



1. Infrastructure as Code (IaC) とは



次のようなお悩みを抱えていませんか

- サーバーの構築に時間がかかっている
- 設定ミスが発生することがある
- 担当者によって設定内容が違うことがある
- 時間の経過に伴いサーバーによって設定がバラバラになっている



Infrastructure as Code (IaC)

- ソフトウェア開発で役立つ手法や習慣などをシステムの運用管理に応用する考え方
- ソフトウェア開発で役立つ手法や習慣の例
 - 自動化
 - バージョン管理
 - テスト
 - 継続的インテグレーション



Infrastructure as Codeの利点

- 自動化で環境構築にかかる時間を短縮
- 作業の自動化・簡素化で人為的ミスを低減
- 属人化を解消
- 常に一定の品質を保つことができる



コードとして管理することで変化

- ITインフラの構築や運用に関わる作業をコード化することで自動化を実現
 - 作業手順書→コード
 - 手作業でセットアップ→自動構築
 - 手動テスト（目視で確認）→自動テスト
 - Excelを使って設定変更の履歴を管理→バージョン管理システム（Git）の利用



2. Claris FileMaker Server for Linux



Claris FileMaker Server

- オンプレミス（自社サーバー）向けの Claris FileMaker サーバー製品
- 対応プラットフォーム
 - Windows Server
 - macOS
 - Linux（CentOS Linux） **New!**



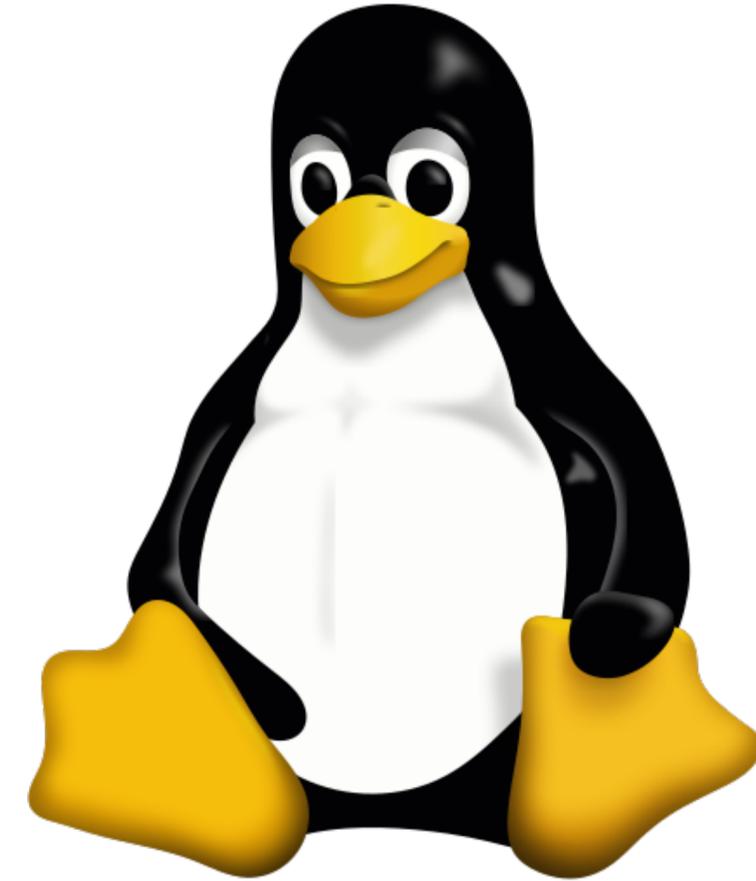
Claris FileMaker Server for Linux

- 現在開発者プレビュー版が公開されている
- 近日正式版が登場予定
- laCを導入する場合はLinux版がオススメ



Linux

- クラウドでは業界標準
- IaaSと親和性のあるプラットフォーム
 - ユーザー数が多い
 - 情報やツールが豊富



コマンドラインインターフェース (CLI) での操作が必須

- コマンドプロンプトもしくはPowerShell (Windows)
- ターミナルアプリケーション (macOS)



実際の運用時にはSSHについての理解も必要

- SSH (Secure Shell)
 - ネットワークを介してサーバーに接続するためのソフトウェア
 - 通信経路が暗号化される
- sshコマンドでリモートサーバーに接続
 - サーバー構成管理ツールでも使われる



3. サーバーの構成管理やテストを行うツール



今回紹介するツール

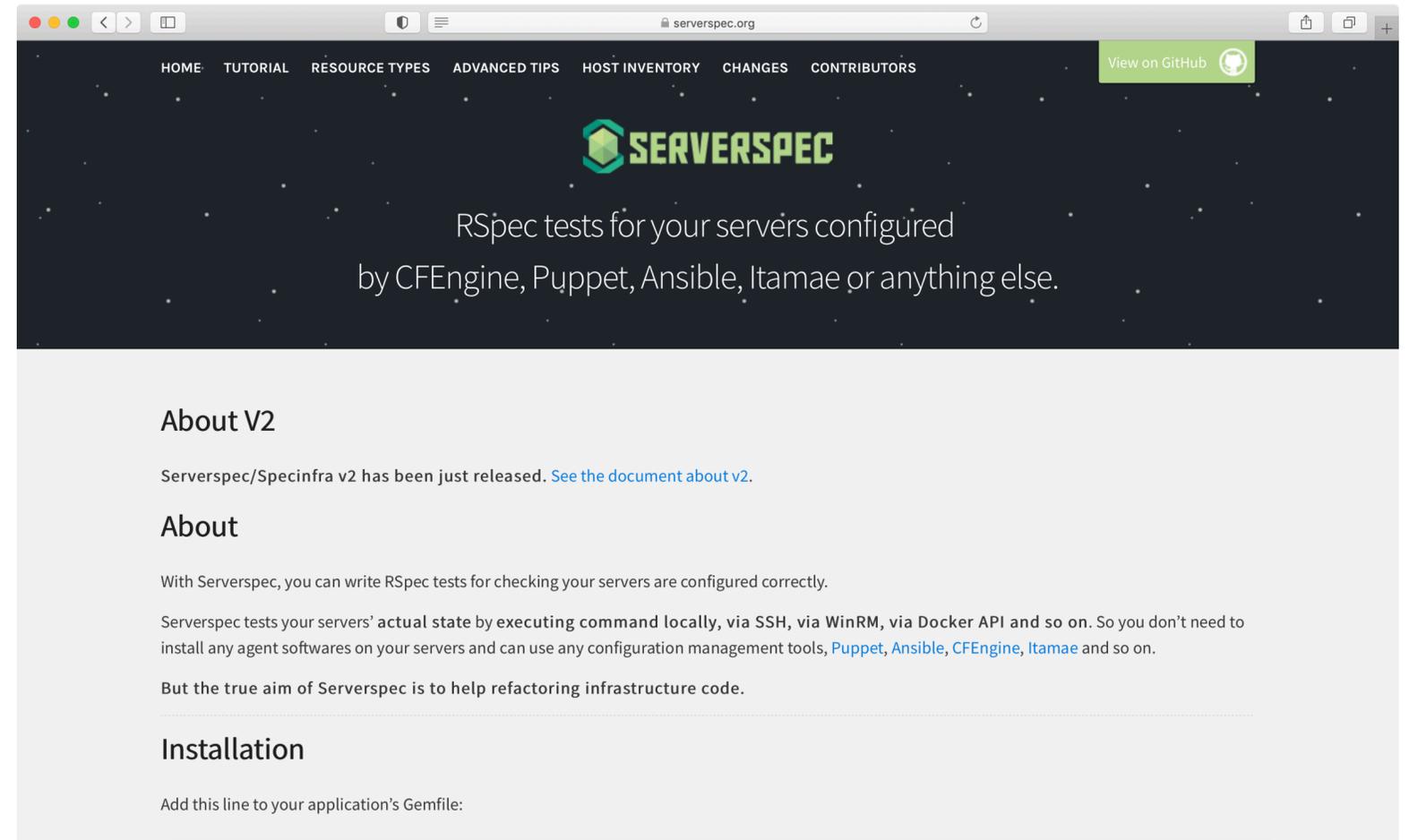
- サーバーテストツール (Serverspec)
- サーバー構成管理ツール (Ansible)



サーバーテストツール

- Serverspec

- <https://serverspec.org/>



Serverspecでの記述例

記述例

```
describe package('httpd') do
```

```
  it { should be_installed }
```

```
end
```

```
describe service('httpd') do
```

```
  it { should be_running }
```

```
end
```



Serverspecのインストール

■ CentOS Linux 7におけるインストール手順の例

```
% sudo yum install -y centos-release-scl↵
```

```
% sudo yum install -y rh-ruby27↵
```

```
% scl enable rh-ruby27 bash↵
```

```
% gem install serverspec rake↵
```



Serverspecの始め方

■ ローカルでの実行例

`% serverspec-init↵`

■ `1↵ (UN*X)`

■ `2↵ (Exec (local))`

`% rake spec↵`



サーバー構成管理ツール

- Ansible
- Chef
- Puppet
- Itamae



Ansibleの特長

- シンプル
 - Playbookという構成定義ファイルを記述
- エージェントレス
 - 操作対象サーバー側に専用ソフト（エージェント）をインストールする必要がない



Ansibleのインストール

■ CentOS Linux 7におけるインストール手順の例

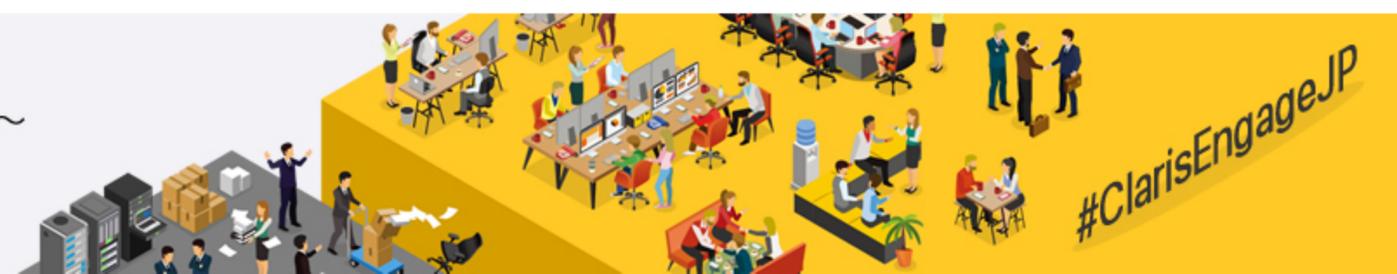
```
% sudo yum install -y epel-release↵
```

```
% sudo yum install -y ansible↵
```



Ansibleの始め方

- ローカルでの実行例
- ファイルを2つ作成（ファイル名は任意）
 - hosts : インベントリ
 - httpd.yml : Playbook（定義ファイル）



Ansibleの始め方

■ hostsファイルの中身

```
[localhost]
```

```
127.0.0.1 ansible_connection=local
```



Ansibleの始め方

■ httpd.ymlファイルの中身

- hosts: localhost

become: yes

tasks:

- yum: name=httpd state=installed

- service: name=httpd state=started enabled=yes



Ansibleの始め方

- ローカルでの実行例

```
% ansible-playbook -i hosts httpd.yml↵
```

- 実行後にServerspecでテスト

```
% rake spec↵
```



その他のツール

- バージョン管理システム

 - Git

- ソフトウェア開発プラットフォーム

 - バージョン管理システム、リポジトリの管理・共有や継続的インテグレーションなど

 - GitHub

 - GitLab



デモ



デモについて

- バージョン管理システムを活用したデモの一例
 - GitLabとDockerをインストールしておいたLinuxサーバー（CentOS Linux 7）を用意
 - gitコマンドでGitLabにデータを反映したタイミングで自動的に任意の処理を実行
 - Dockerコンテナ上でAnsibleとServerspecをインストールしてサーバー構成管理ツールおよびサーバーテストツールを自動実行



GitLab



docker



Claris Engage Japan 2020

～デジタルトランスフォーメーション～
2025年の崖を飛び越える



バージョン管理システム導入のメリット

- 変更履歴を共有して確認しやすくなる
- 課題に関連するコードの変更内容を確認しやすくなる
- コードを更新した際に自動テストを実行できるようになる
- コードレビュー後にサーバーに設定を反映させることも可能



Infrastructure as Code (IaC) 導入のメリット

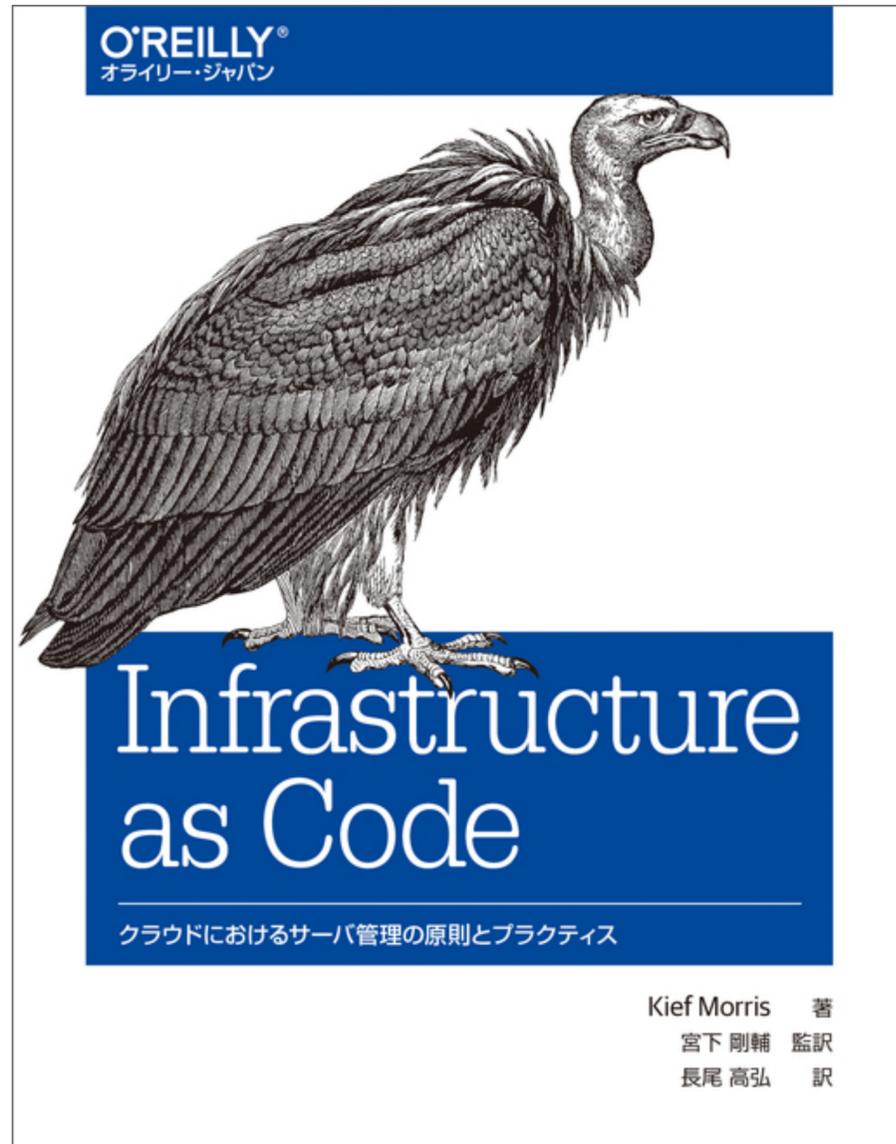
- 自動化で環境構築にかかる時間を短縮
- 作業の自動化・簡素化で人為的ミスを低減
- 属人化を解消
- 常に一定の品質を保つことができる



ツールを導入しただけでは終わりではない

- チームとして継続的な改善に取り組む組織の体制や風土も必要
- 一気にすべてをコード化しようとしな
 - 少しずつテストや構成管理用のコードを記述していくという方法でも良い
 - テストは財産になる
 - 無理のない範囲で始め改善していくことが大事





参考文献

Infrastructure as Code

発行：オライリー・ジャパン

発売：オーム社

ISBN：978-4-87311-796-6



参考) fmcsadmin

エミックで開発したコマンドラインツール

- リモート管理機能を備えたサーバー管理者向けコマンドラインツール
- Linux、macOSおよびWindowsで利用可能
- <https://www.emic.co.jp/products/fmcsadmin/>



Claris Marketplace

注目の製品

世界中の Claris のエキスパートが開発した製品をご覧ください。



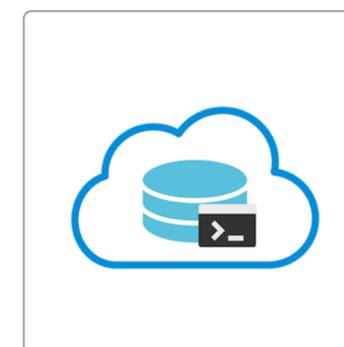
FileMaker 19 スーパーリファレンス

Windows & macOS & iOS対応 (基本からしっかり学べる)
- ソーテック社



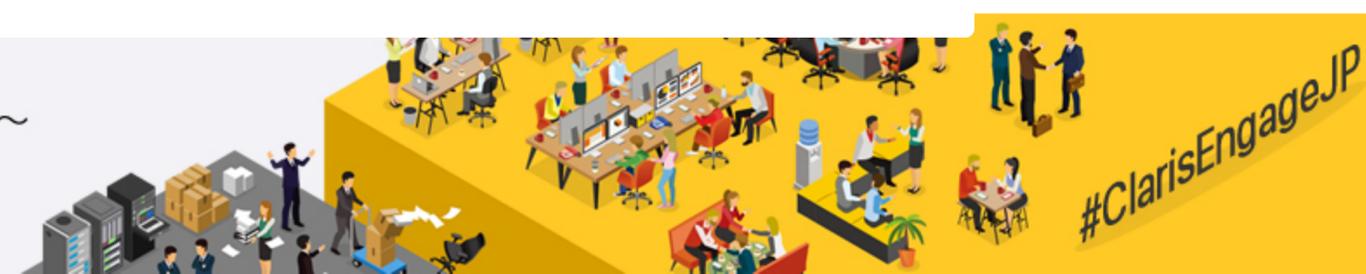
DayBack カレンダー

自分好みにカスタマイズできるカレンダー



fmcsadmin

Linux、macOSおよびWindowsで利用できる、リモート管理機能を備えたサーバー管理者向けコマンドラインツール





Claris Engage Japan 2020

