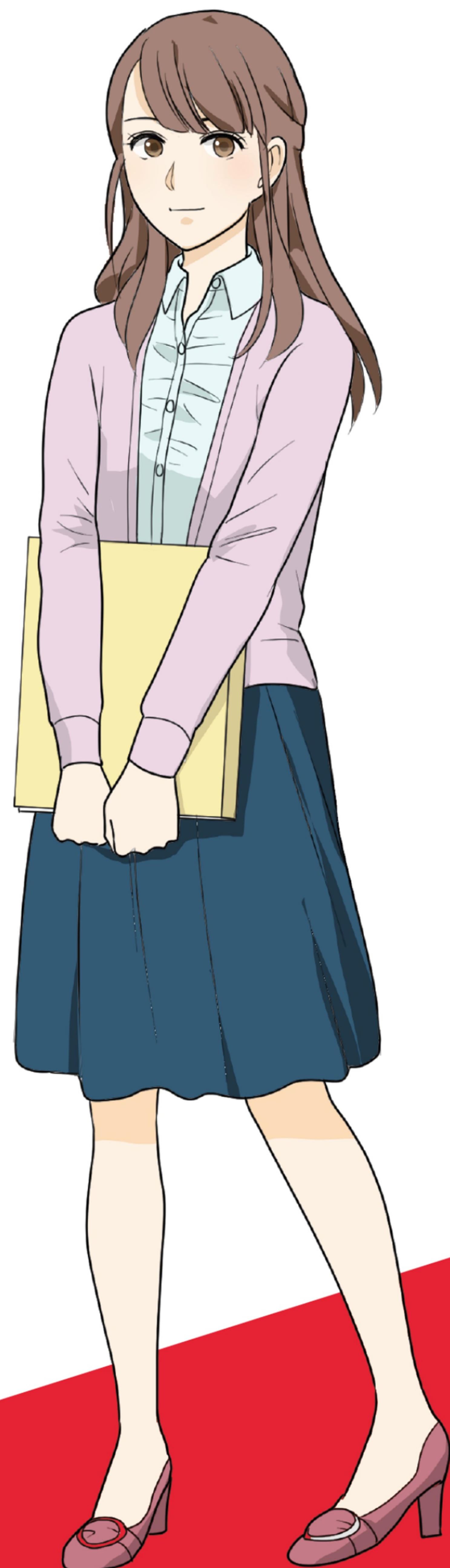


エンジニアの働き方改革を進める  
**Ansible Automation**

マンガでわかる



**RED HAT**<sup>®</sup>  
**ANSIBLE**<sup>®</sup>  
Automation



# 登場人物紹介



## 山田 徹 35歳

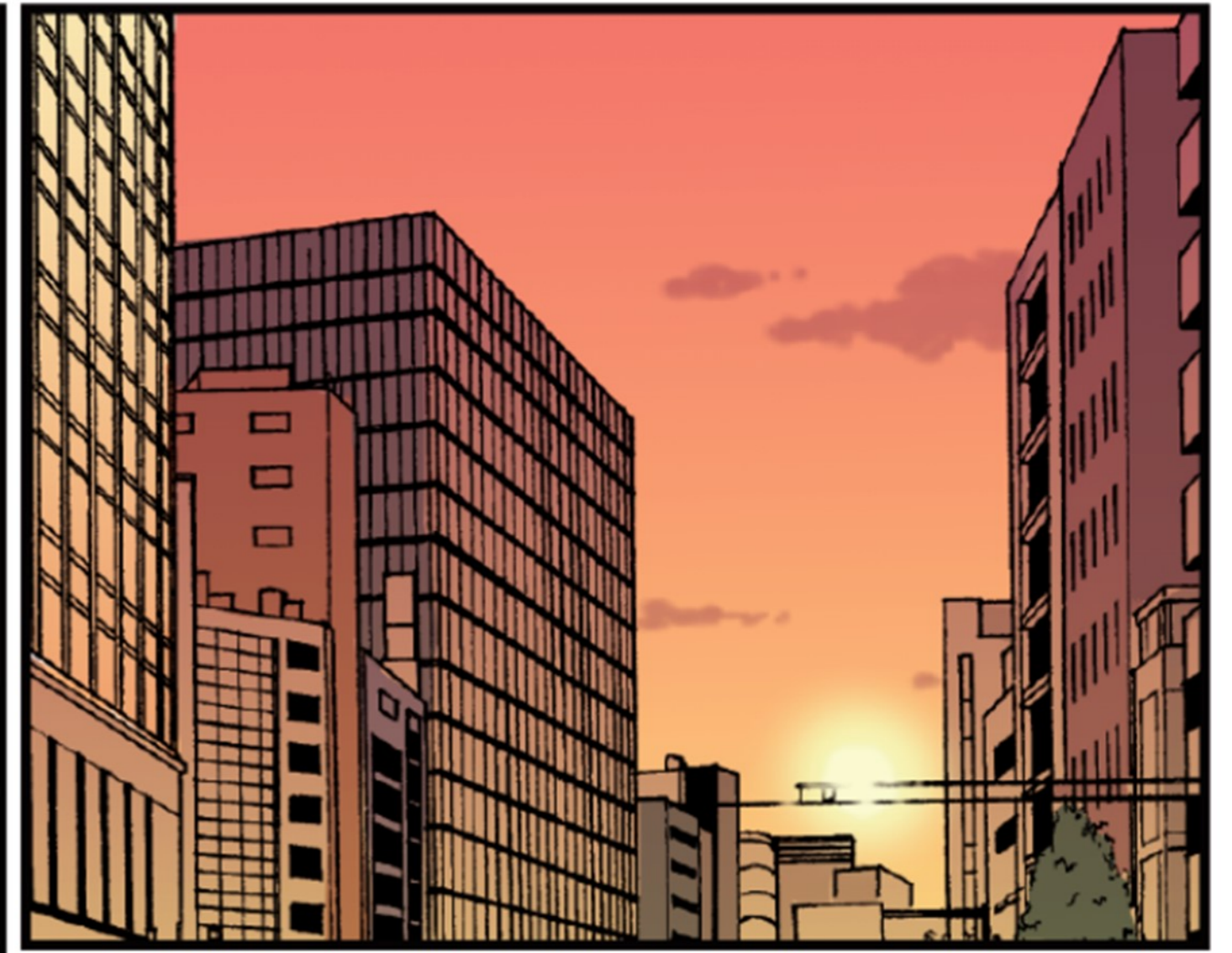
進藤の先輩。  
「己の技術こそが至高！」と信じてやまない中年ギラギラエンジニア。



## 松本 豊 25歳

進藤の同僚であり  
良き相談相手。  
進藤とともに  
日々技術を磨く。





情報  
システム部

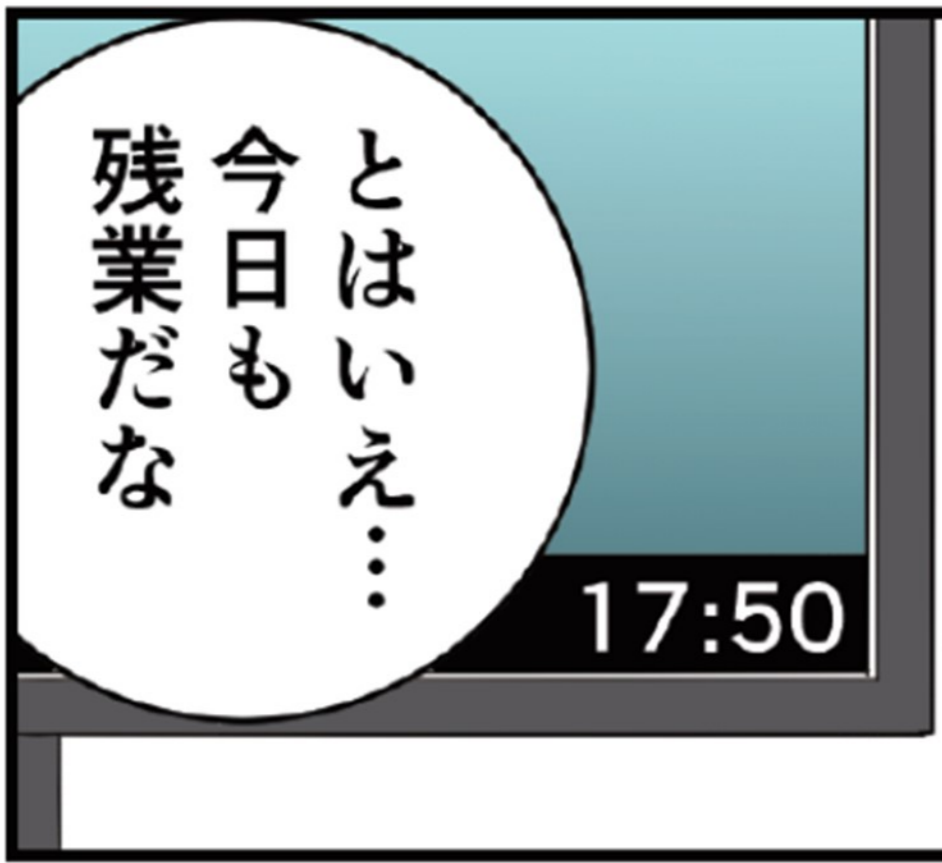


夜間  
メンテナンスの  
準備  
間に合うかな

進藤直樹(25歳)

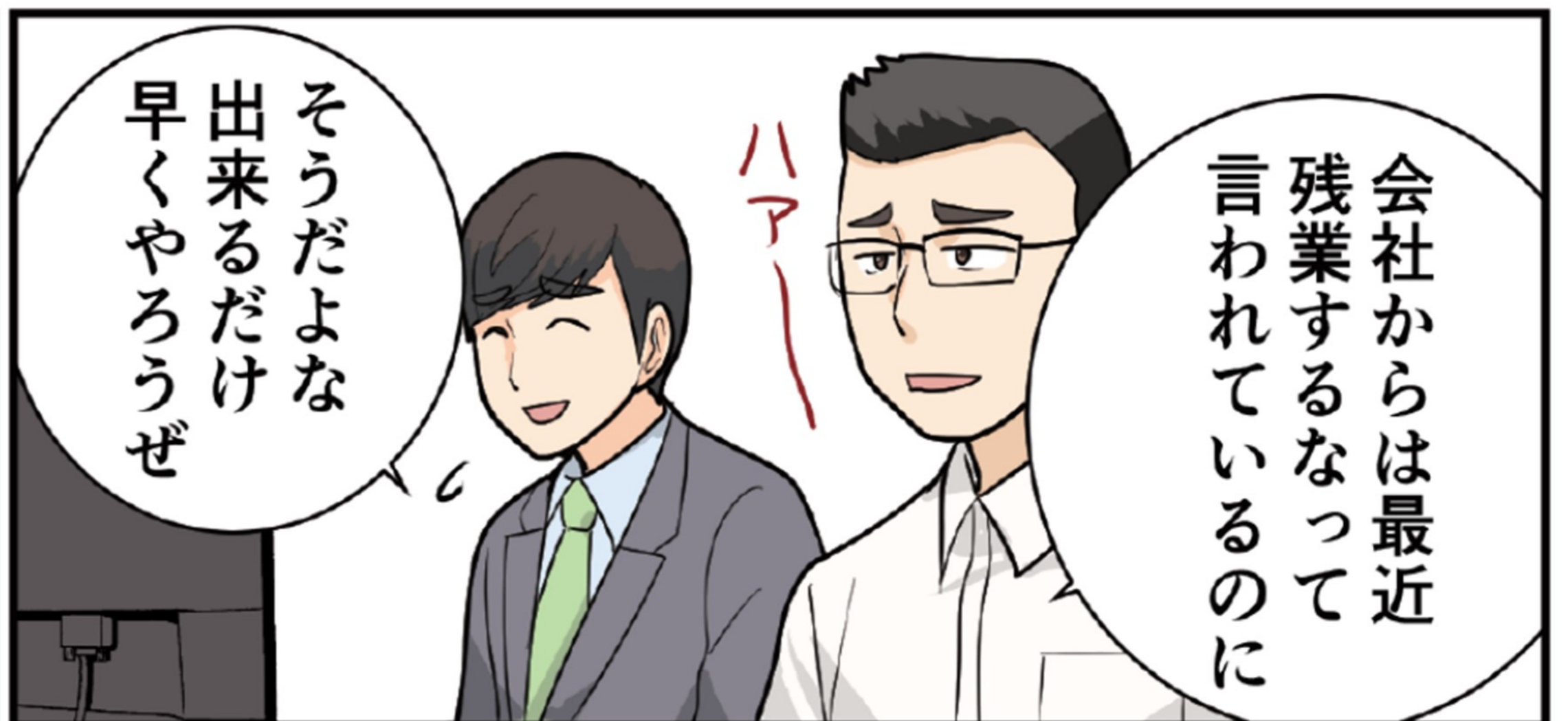


先週凡ミスで  
皆に  
迷惑かけたから  
今回はシッカリ  
準備するぞ



とはいえ…  
今日も  
残業だな

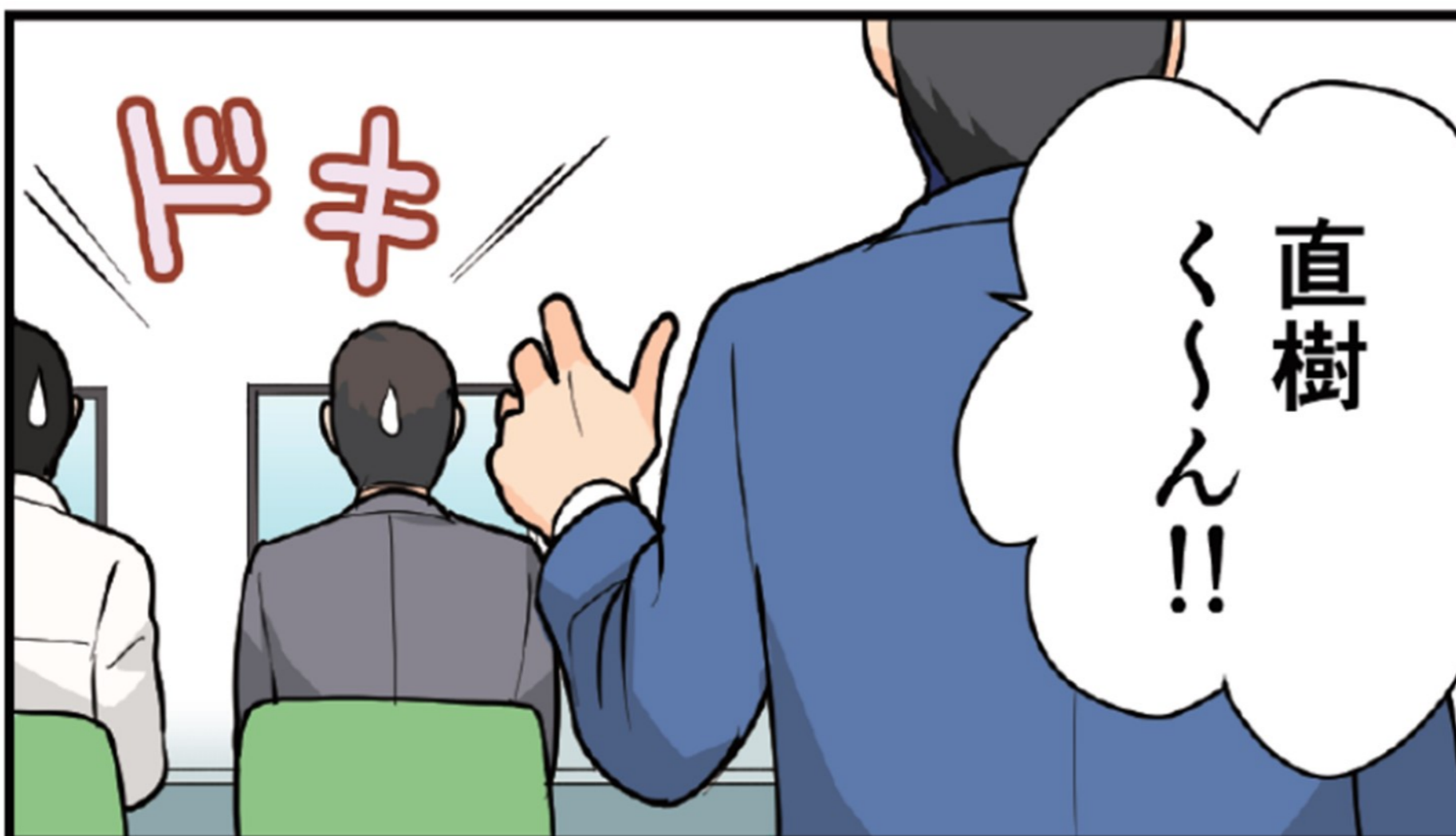
17:50



会社からは最近  
残業するなって  
言われているのに

そうだよな  
出来るだけ  
早くやろうぜ

ハア



直樹  
く〜ん!!

ドキ



直樹!  
今日も残業か  
頑張るね〜

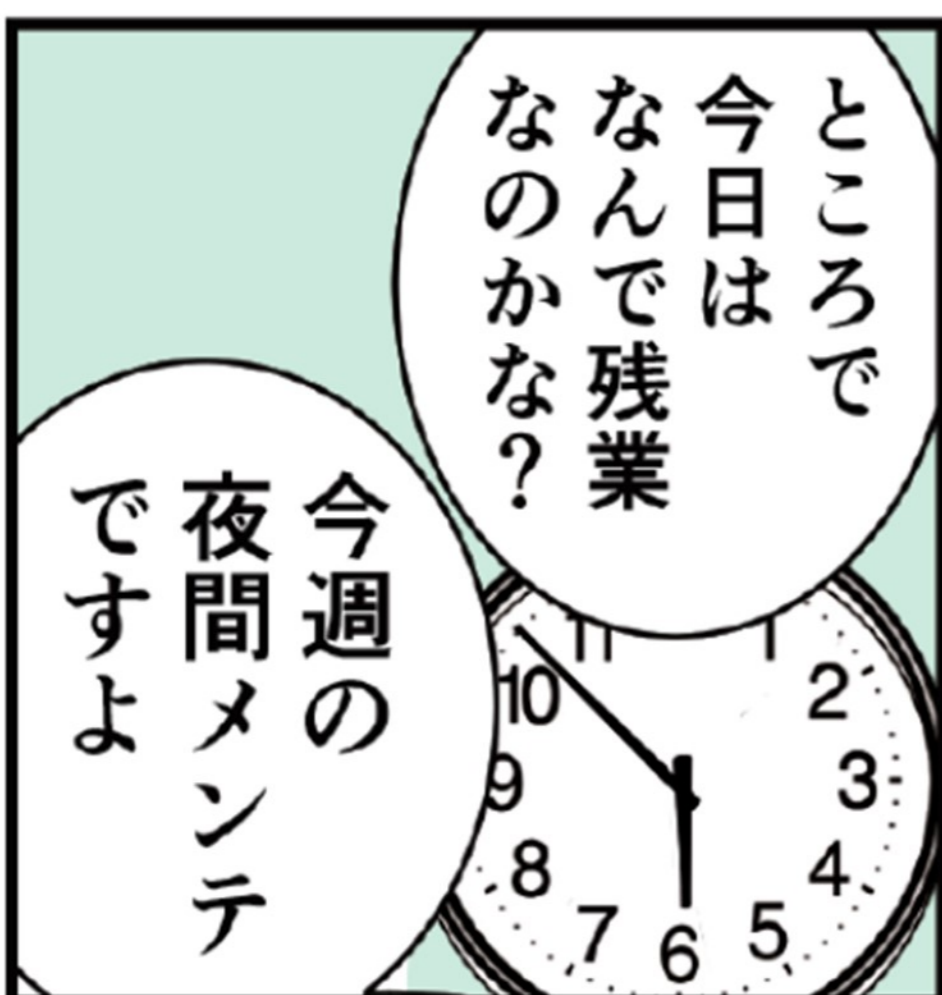
山田徹(35歳)



先輩はいつも  
お元気  
ですね…

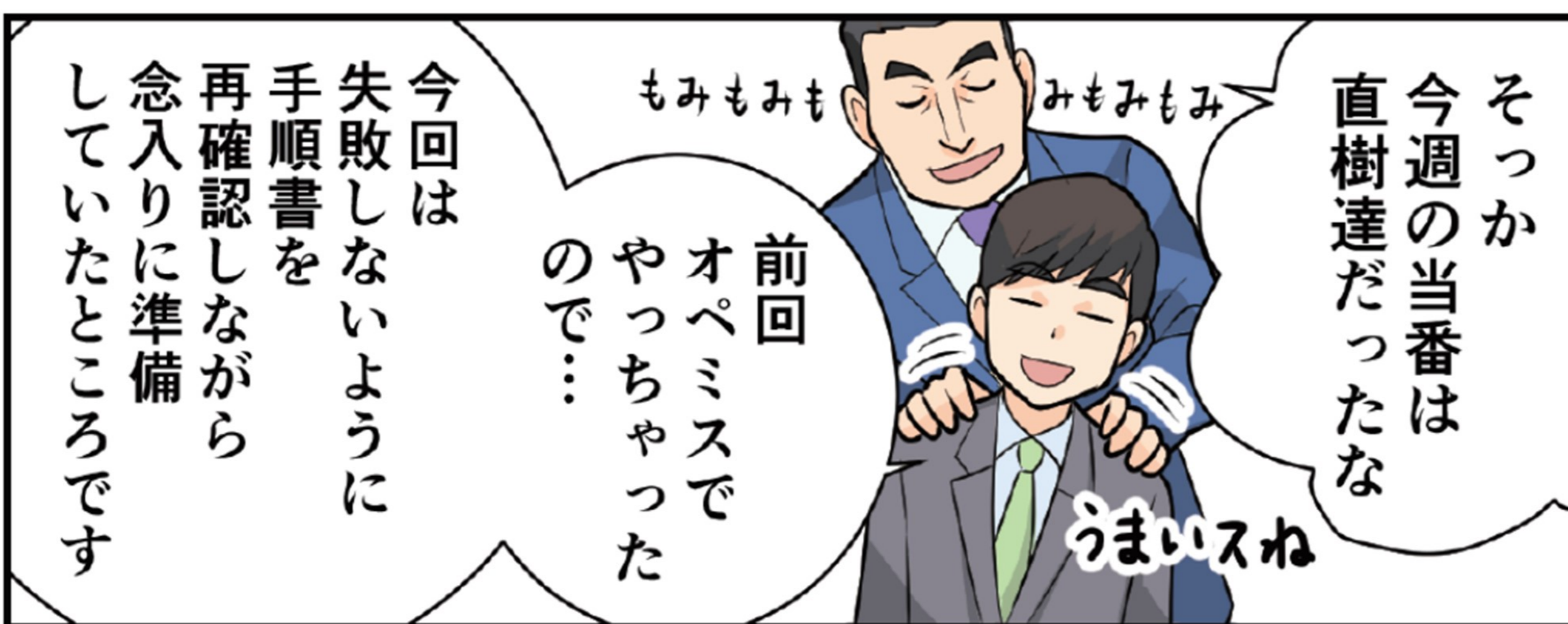
お前も  
元気よ!  
なあ!

ガハハ



ところで  
今日は  
なんで残業  
なのかな?

今週の  
夜間メンテ  
ですよ



そっか  
今週の当番は  
直樹達だったな

前回  
オペミスで  
やっちゃった  
ので…

今回は  
失敗しないように  
手順書を  
再確認しながら  
念入りに準備  
していたところです

うまいスね



まっ  
3年目の  
君達には大変  
だろうけど

ドヤァ

俺のように  
何百回と  
夜間  
メンテナンスを  
経験すれば簡単な  
作業だけだな!

ネットワークの  
設定 苦手で

でも  
ユーザー設定も  
やらなきゃ  
ダメだし

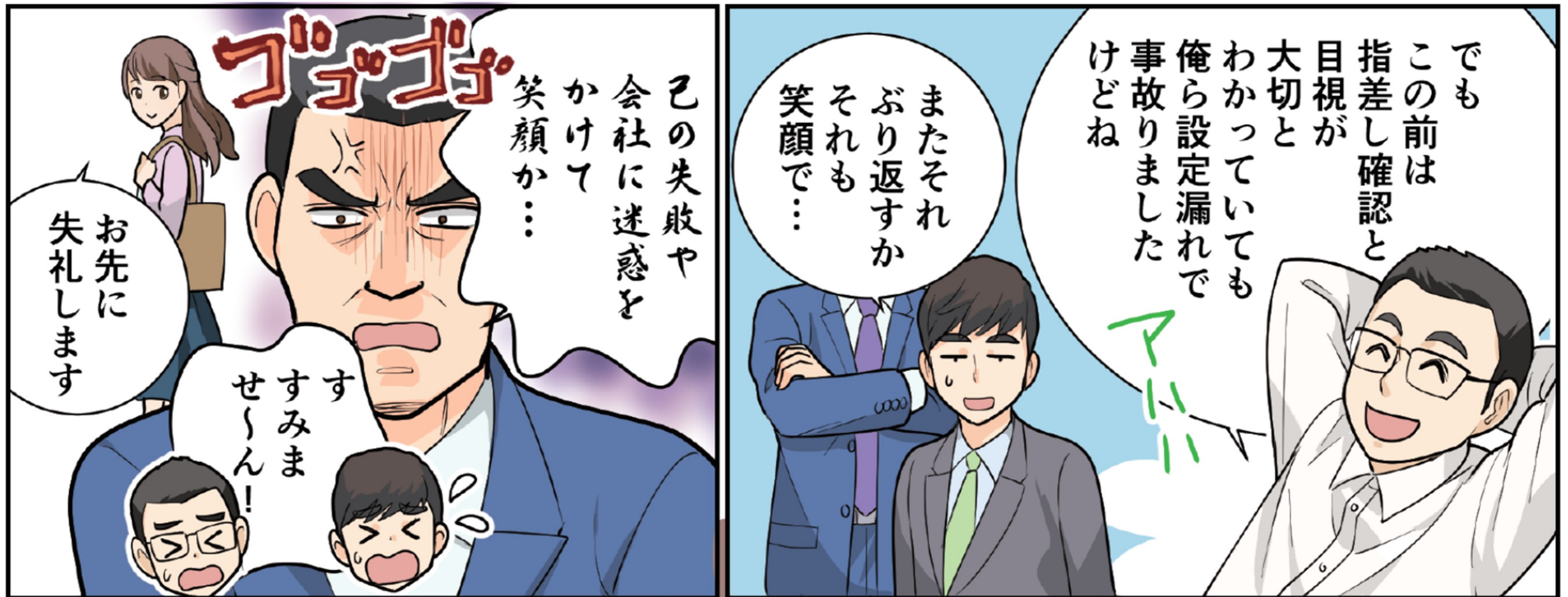
エヘ...

メンテが  
簡単にできる  
仕組みとか  
あればいいのに

確認項目も  
多いから  
大変ですよ

ハア...

俺の  
エナジー  
取りました  
ね



ゴッゴッ

己の失敗や  
会社に迷惑を  
かけて  
笑顔か...

でも  
この前は  
指差し確認と  
目視が  
大切と  
わかっていても  
俺ら設定漏れで  
事故りました  
けどね

またそれ  
ぶり返すか  
それも  
笑顔で...

お先に  
失礼します

すみま  
せくん!

ア...



そっか  
コレ!

あはい

まっ  
座りなよ

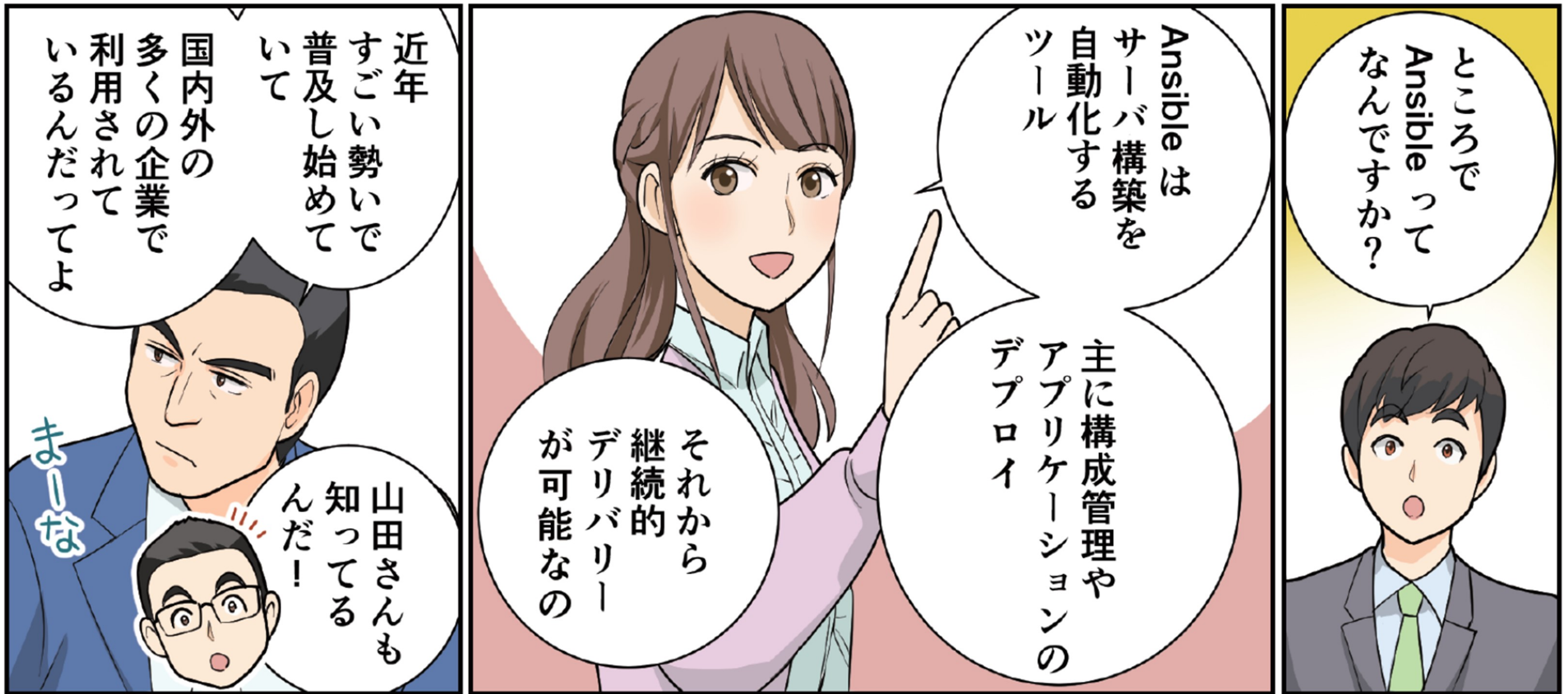
俺のエナジー...

私が  
いなくても  
大丈夫  
なんです

氷室 杏(28歳)

私のチームの  
メンテナンス作業は  
Ansideで  
全て自動化されて  
いるので

あれ 杏ちゃんの  
チームって  
今夜大規模な  
メンテじゃ  
なかったっけ?



## 自動化を表現する「Playbook」

・YAML形式で記述されたパラメータと手順

```
- name: install and start apache
```

```
hosts: web
```

```
become: yes
```

```
vars:
```

```
  http_port: 80
```

```
tasks:
```

```
- name: httpd package is present
```

```
  yum:
```

```
    name: httpd
```

```
    state: latest
```

```
- name: latest index.html file is present
```

```
  copy:
```

```
    src: files/index.html
```

```
    dest: /var/www/html/
```

```
- name: httpd is started
```

```
  service:
```

```
    name: httpd
```

```
    state: started
```

設定内容を記述  
(ここではApacheを  
インストールし、起動する)

対象はWebサーバ

ルート権限で実行する

httpの使うポートは80番

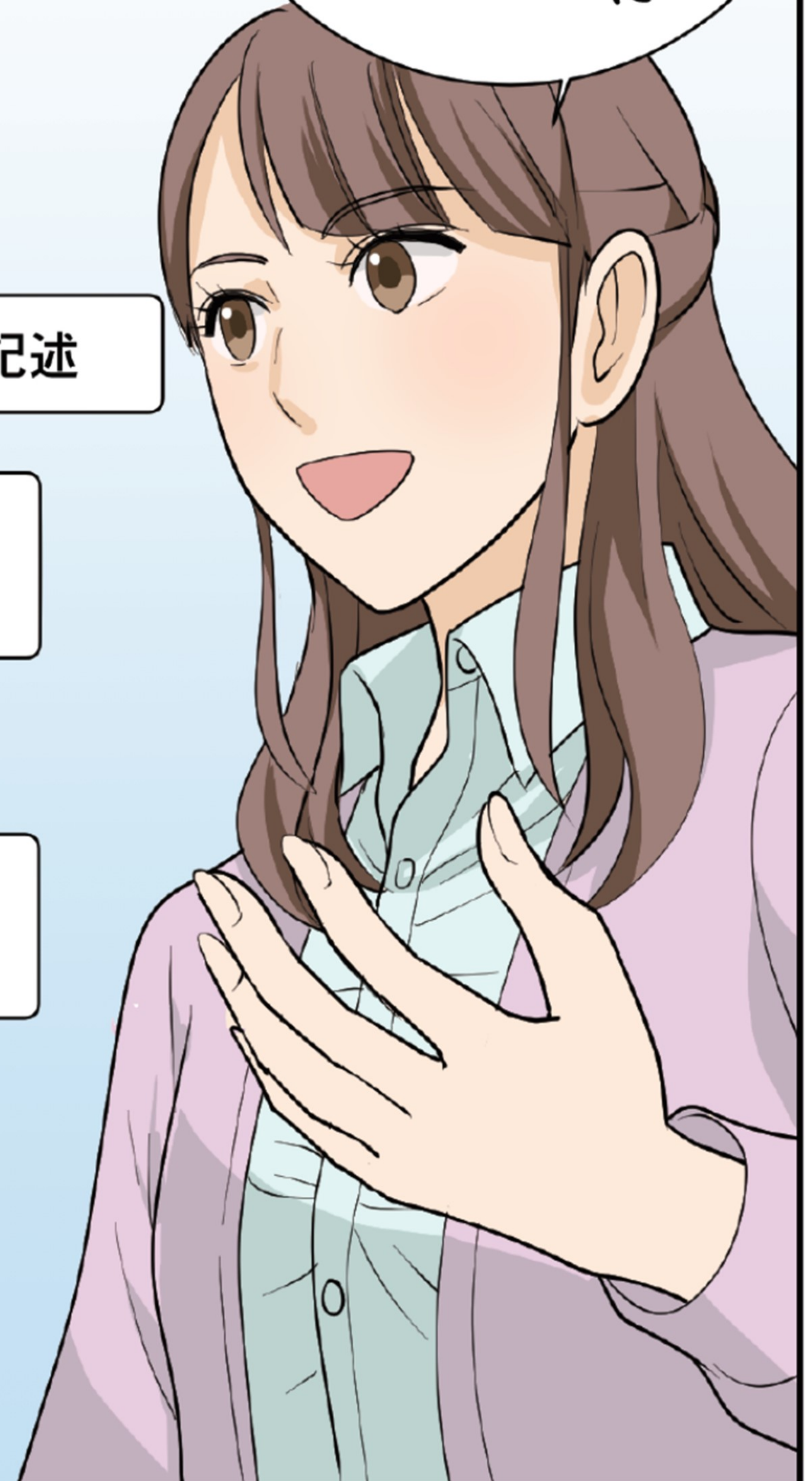
実行内容を記述

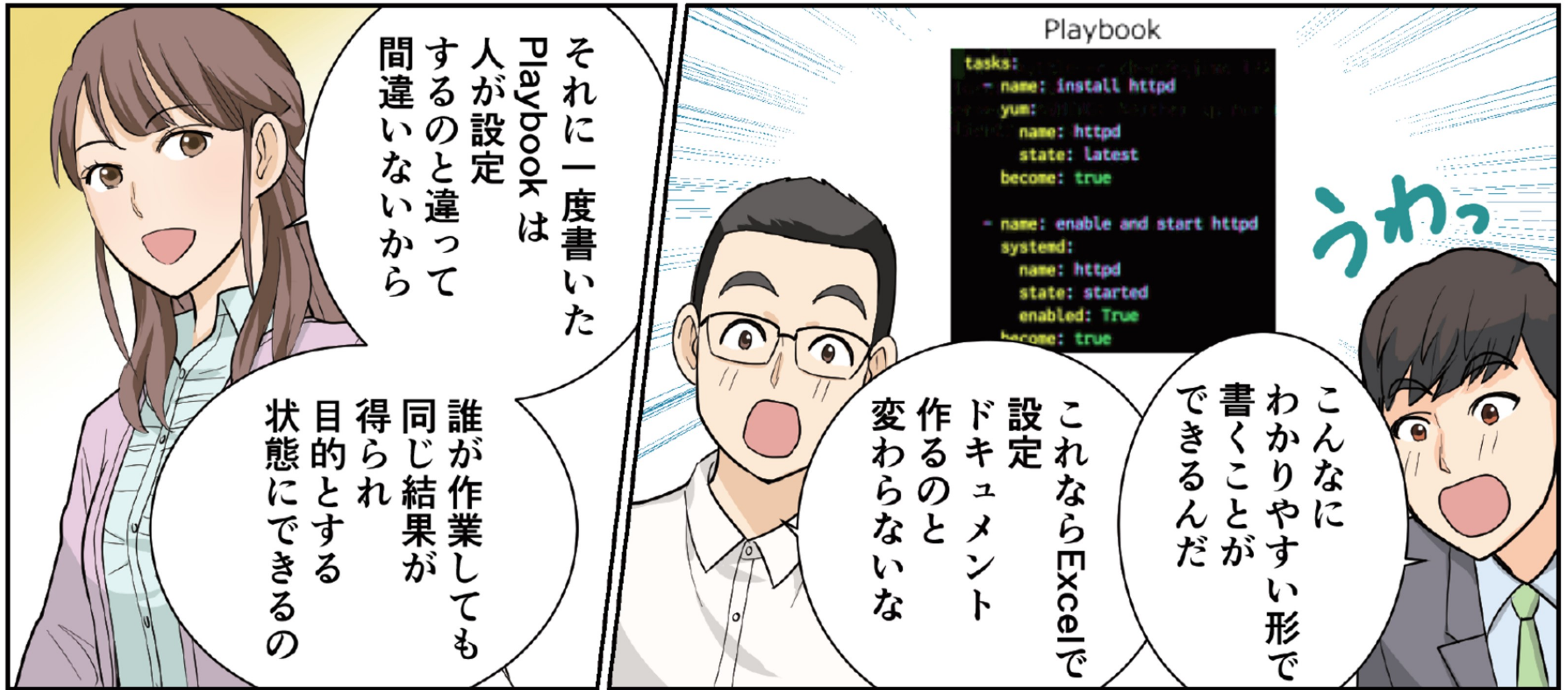
Linuxコマンドのyumで  
httpdの最新版をインストール

htmlファイルを  
任意の場所にコピー

httpdを起動する

Playbookって  
呼ばれるファイルに  
設定したい状態を  
記載するだけで  
あとは自動的に  
設定が  
完了するのよ





それに一度書いた  
Playbookは  
人が設定  
するのと違って  
間違いないから

誰が作業しても  
同じ結果が  
得られ  
目的とする  
状態にできるの

これならExcelで  
設定  
ドキュメント  
作るのと  
変わらないな

こんなに  
わかりやすい形で  
書くことが  
できるんだ

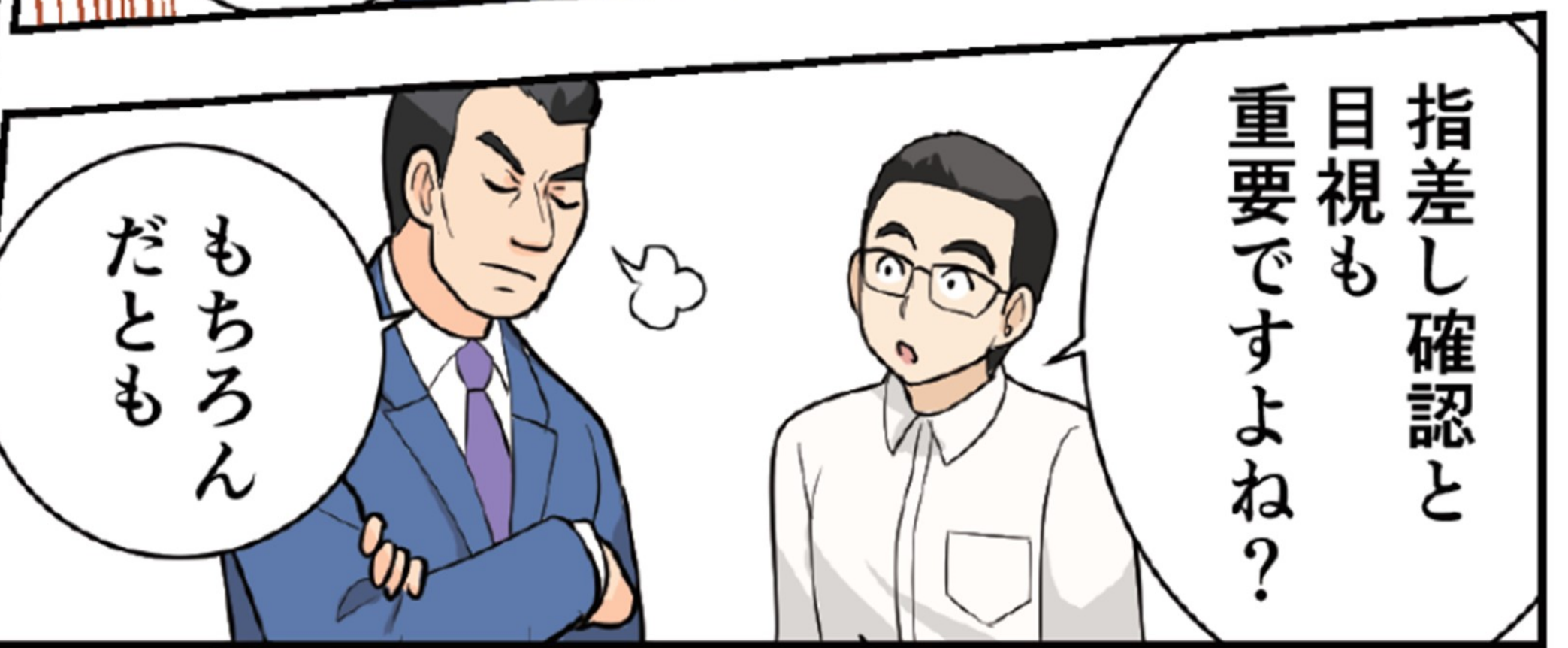
うわっ



ちよつと  
まった!

エッ...



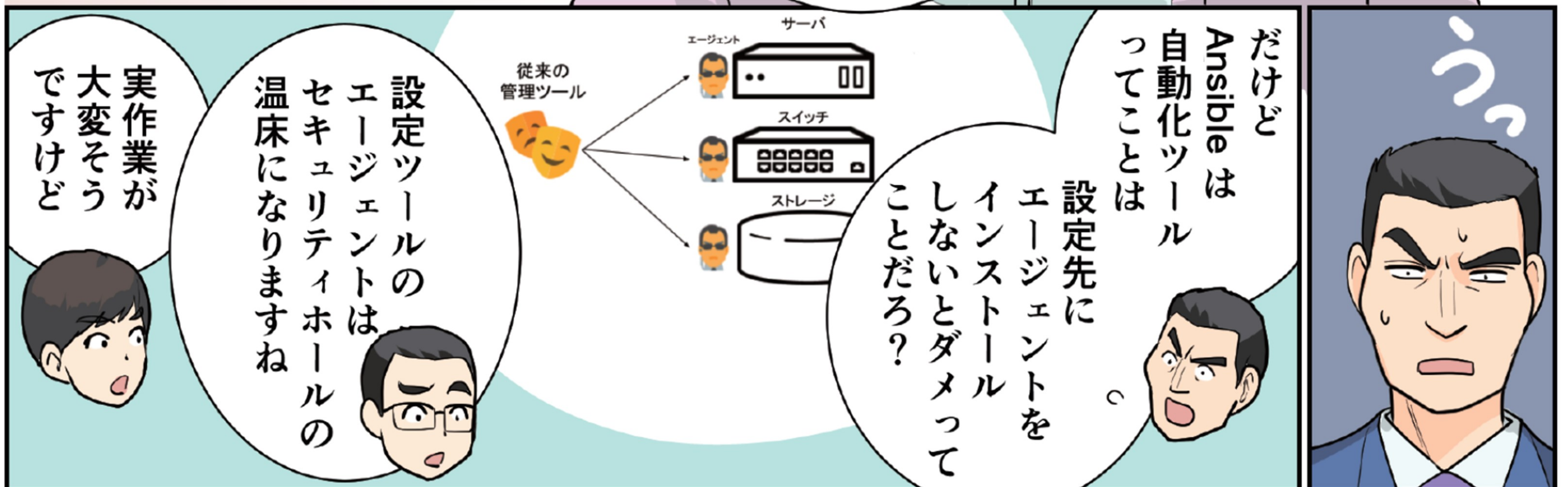


### 1. スクリプトの巨大化と秘伝化

優秀なエンジニアが作った自動化スクリプトは、あらゆる作業をスクリプトに実装してしまい巨大化する傾向に。優秀なエンジニアが部署異動などをした場合に、自動化スクリプトの保守ができず、ブラックボックスになる可能性がある。

### 2. 組織間での共有がしにくい

各チームや個人が異なるスクリプト言語を使ってしまい、部署間でのベストプラクティス共有ができなくなる。運用管理の手法がサイロ化され、部署間で同様の作業を別の手法でしてしまう可能性がある。



大丈夫！

Ansibleはエージェントレスと呼ばれるタイプの設定ツールなので

設定側にエージェントをインストールする必要は無し

SSHやWinRM, RESTなど

サーバ  
スイッチ  
ストレージ

RED HAT ANSIBLE Automation

SSHもしくはWinRMでサーバからつなぐことができるものなら基本Ansibleで設定が可能

1,750以上のモジュールを提供しあらゆる機器を自動化することが可能

クラウド	仮想化・コンテナ	Windows	ネットワーク	チャット	監視ツール
AWS Azure CenturyLink Digital Ocean Google OpenStack Rackspace +more	Atomic CloudStack OpenStack OpenShift RHEV VMware +more	ACLs Files Packages IIS Regedits Shares Services Configs Users Domains +more	Arista A10 Cumulus Bigswitch Cisco Cumulus Dell F5 Juniper Palo Alto OpenSwitch +more	Email HipChat IRC Jabber Rocketchat Sendgrid Slack Twilio +more	Airbrake BigPanda Datadog LogicMonitor Monit Nagios New Relic PagerDuty Sensu StackDriver Zabbix +more

[https://docs.ansible.com/ansible/latest/modules/modules\\_by\\_category.html](https://docs.ansible.com/ansible/latest/modules/modules_by_category.html)

また多くのクラウドサービスがAnsible対応しているの

オンプレ仮想化パブリッククラウドやプライベートクラウドに関係なくAnsibleは利用できるの

だから今まで自動化ツールの対象ににくかったネットワーク機器やベンダー特有のソフトウェアなんかもAnsibleでは設定が可能になったのも嬉しいポイントよ

おおー

このPlaybook

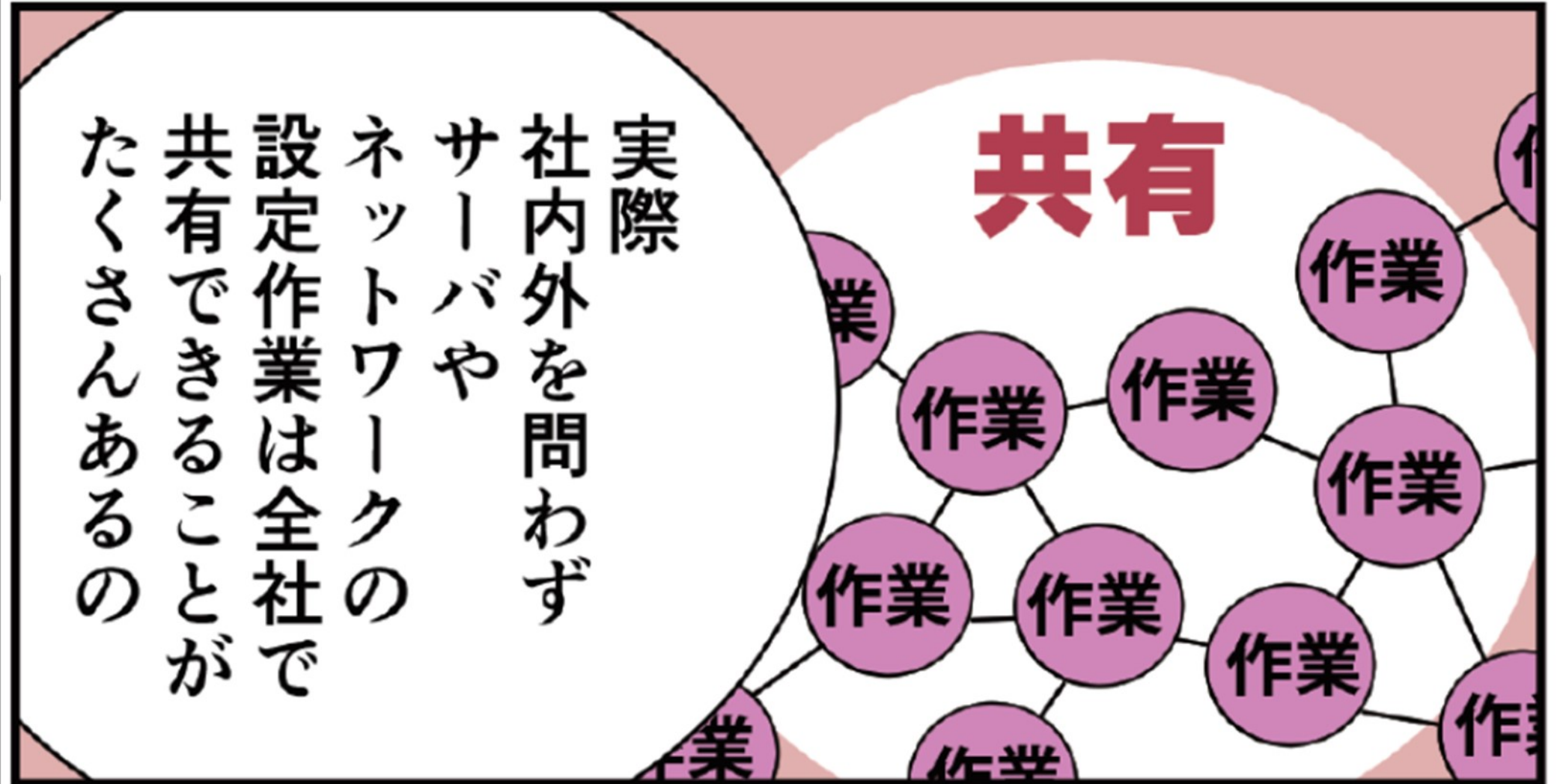
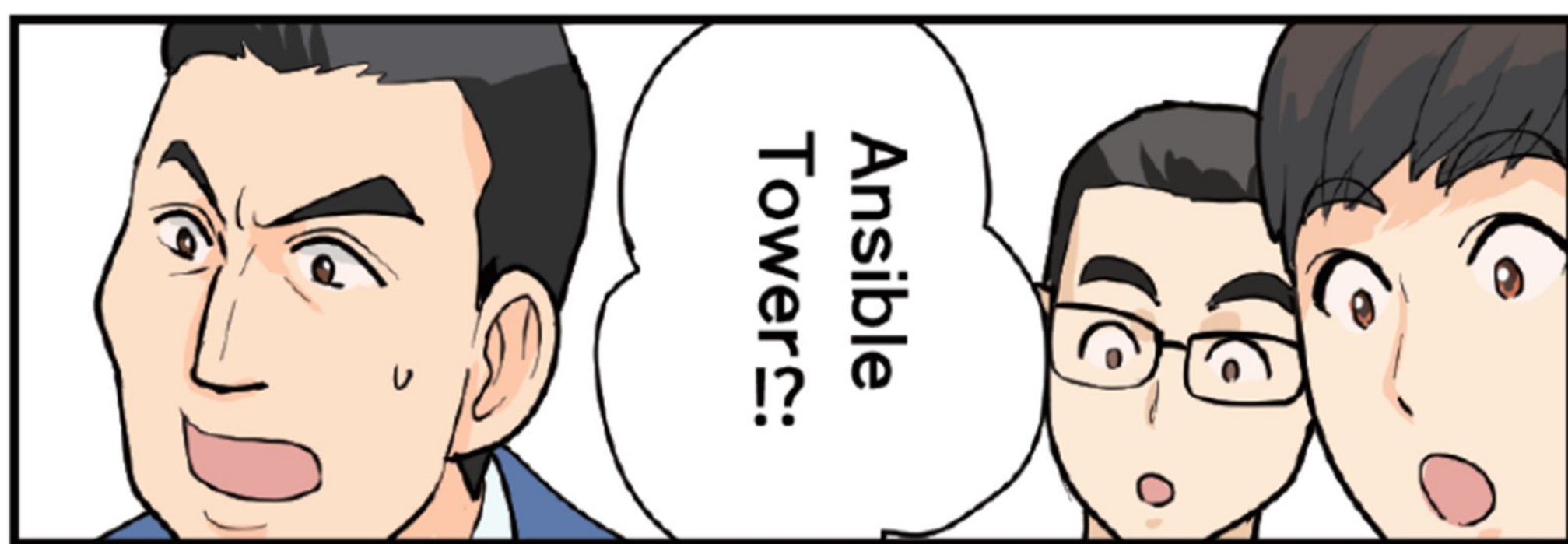
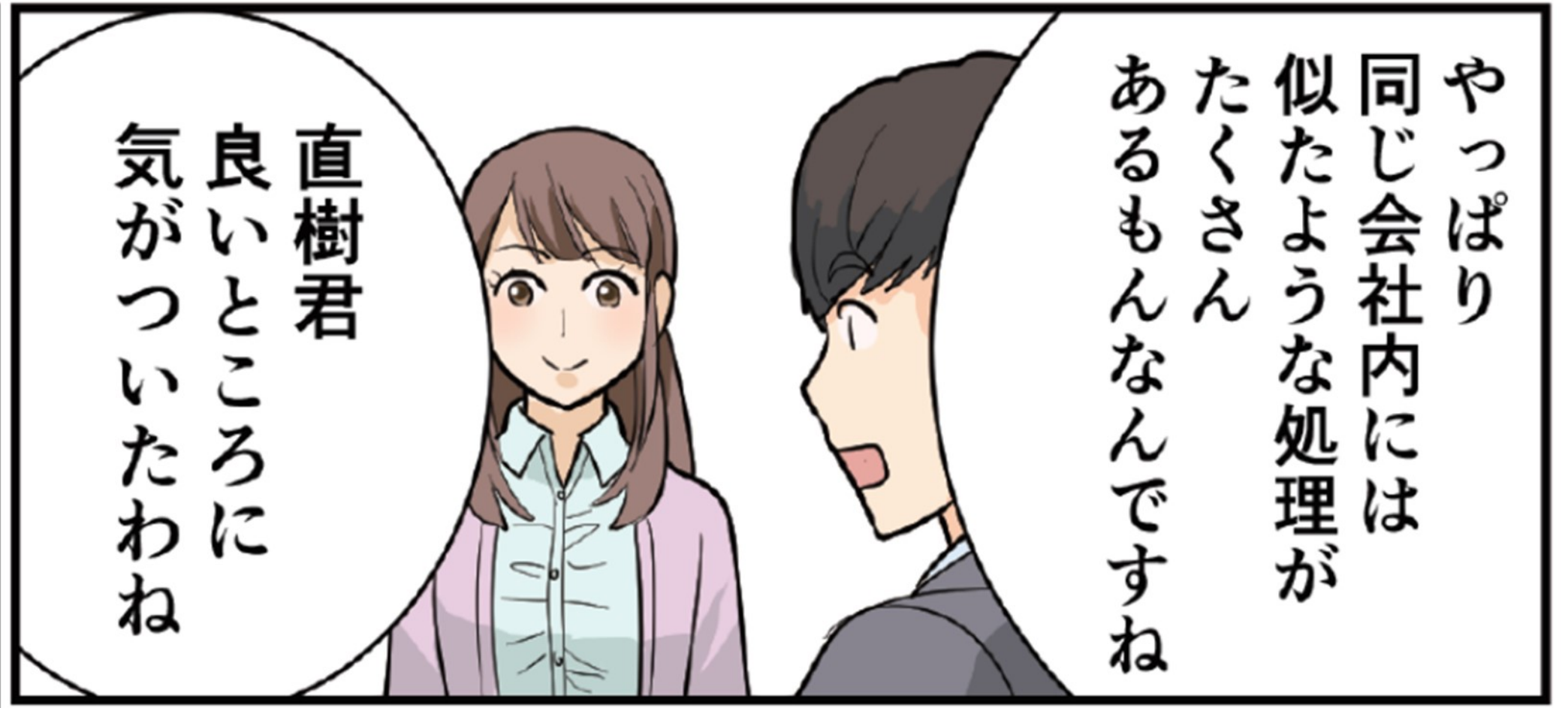
ぼくが今からやろうとしているメンテナンス処理の内容にそっくりだ

ほんとか？そんな偶然あるのかよ

あれ？

どうしたの？





RED HAT ANSIBLE Tower

アクセス制御  
監査ログ  
スケジュール  
権限管理  
カタログ管理  
ワンクリック実行  
API & CLI  
Job Workflow

Ansible Tower は 機密情報の セキュリティを 担保しつつ

組織内の円滑な ナレッジ共有を 実現できる マネージメント ツール

Playbook の 共有はもちろん

Web ツールからの Playbook の 実行や状態の把握 監査機能等もあるの



国内Ansibleの導入効果例(一部机上での試算を含みます)

Before	After
<b>アプリケーションのリリース作業</b> <b>通信業</b> 定期的に行われるアプリケーションのリリース作業を自動化して90%の工数を削減 200時間 → 20時間	<b>プライベートクラウドのリソース払い出し作業</b> <b>情報サービス業</b> リソースの払い出しを自動化し、大幅なスピードUPを実現 12日(リードタイム含む) → 10分
<b>仮想化基盤の全通用作業</b> <b>社会インフラ</b> 日常的な運用作業を自動化することで全体工数を半減。削減した工数を新たな仕事へシフトすることを検討 100% → 50%	<b>大規模クラウドサービスの通用作業</b> <b>情報サービス業</b> クラウド基盤のメンテナンス作業を自動化して大幅な時間短縮を実現し、オペミスも削減 60時間 → 10分

多くの企業でサーバやネットワーク機器が自動化された運用実績がこれ

たしかにAnsibleはOSS

でもねレッドハットがサポートを提供してるので安心して利用できるのよ

Redhat

わああああ

むしろプロプラツールよりも安定してると言えるわね

ドン!

1年後に数万ノードの自動化まで進んだ事例もあるそうよ

一年で数万ノード!!

海外の事例ではAnsibleで数百ノードを自動化した後

RED HAT ANSIBLE Tower

ボタッ

ただチーム全体で利用するときつと時間がかかるんだろ?

Ansibleは多くのエンジニアが理解しやすく効果が出るのも早いので

チーム全体どころか全社で利用というケースも珍しくないツールなのよ

HP

ハア ハア

やったー!!

さああればミスや残業もなくなるぞ!

そうだね

……

頑張っただけ帰るよ!

さあ早く終わらせて帰ろう!

俺も帰るよ!

ちなみに社長にはAnsibleの利点は来週から全社展開が決まってるのよ

直樹くんや山田さんのチームにもAnsibleの導入は決定済みよ

# 「Ansibleは現代のPerl」

## 人の仕事をラクにしたIT技術

ITはその誕生から人の仕事を楽にするためにあらゆる進化をしてきた。Ansibleもその進化の過程で生まれたのだが、1990年代にエンジニアから好まれていたPerlと立ち位置が類似していると考えている。

1990年代、運用管理者は日常的な作業のために様々なテキスト操作やsocketを使う簡単なプログラムを書いていた。この時代、これらの処理を簡便に記述できるPerlはエンジニアに広く愛された。2018年の現在では運用管理者はより多くのシステムやサービスを管理するようになり、SSHの普及や各種APIの整備など環境も変わっている。Ansibleはこの背景を踏まえた仕組みになっており、現在の運用管理者にとって必須の道具になりつつある。

## AnsibleとPerlの類似点

まず、PerlとAnsibleは大きく異なる言語だが、どちらも運用管理者の日常的な業務をよく想定した言語機能やモジュールを提供しているという点で類似している。

Ansible	Perl
<ul style="list-style-type: none"><li>・リモートホストへ接続しコマンド実行</li><li>・各種のAPI呼び出し</li><li>・多数のシステムに対して同一処理を実行することに特化</li></ul>	<ul style="list-style-type: none"><li>・基本的なプログラミング</li><li>・Cに対応する低レベル処理（ソケット接続など）を提供</li><li>・高度なテキスト処理</li></ul>

図：PerlとAnsibleは運用管理者の日常的な業務を想定した言語機能やモジュールを提供

適切な言語機能が用意されているため典型的な作業は簡単なスクリプトを書くだけで期待通りの結果が得られる。そして、上級者がモジュールに記述することも、初心者の「ベタ書き」も許容してくれる。こういう点が様々なバックグラウンドを持つエンジニアに支持されたPerlと似通っている。

## Ansibleの強み

AnsibleはSSHで接続し、Pythonがサーバ機器側での処理を行う構造になっている。あらゆる機器・ソフトウェアがSSHで接続可能になり、主要なOSでPythonが動作するという現在の状況は、Ansibleを積極的に選択する一つの大きな理由にもなっている。合わせて、新しく登場した各種ソフトウェアやサービスが当たり前のようにAPIを提供するようになり、これらを容易に操作できるAnsibleが、より一層活用される背景にもなっている。

様々なエンジニアが興味を持ち、お互いの実践ノウハウを語り合う場（コミュニティにおける勉強会など）があるというのもAnsibleが多くのエンジニアに利用されている理由でもある。このような勉強会に積極的に参加し、よりよい使い方を模索し、共有することもできる。30年前に培われたPerlの技術は今なお通用する部分も多く残っている。Ansibleも、この先30年を支える技術として、皆様の技術の柱としてみるのはいかがだろうか。



## レッドハット株式会社

TEL : 03-5798-8500 (代表)  
〒150-0013 東京都渋谷区恵比寿4-1-18 恵比寿ネオナート  
<http://www.jp.redhat.com>

**ANSIBLEについて詳しくはこちらのQRから簡単アクセス! ▶**

<https://www.redhat.com/ja/explore/ansible>

