

# Don't fear the **devops**

Taming servers infrastructure with



Michał Ostruszka

I enjoy **building stuff**\* at **SoftwareMill**

\* Scala, JavaScript, TypeScript, Node.js, some Clojure



# Devops

*development and operations*

It is a **culture, movement or practice** that emphasizes the collaboration and communication of both software developers and other IT professionals while **automating the process of software delivery and infrastructure changes**. It aims at establishing a culture and environment where **building, testing, and releasing software, can happen rapidly, frequently, and more reliably**.

<https://en.wikipedia.org/wiki/DevOps>



## AUTOMATION FOR EVERYONE

App deployment, configuration management and orchestration - all from one system.

Powerful automation that you can learn quickly.

# Automation with Ansible

- Infrastructure as code
- Reliable and repeatable provisioning
- Quick envs setup (prod, staging, perftest)
- Serverless, agentless (just SSH)
- Really easy start (just an inventory file)
- Fully declarative, readable YAML code

Talk is **cheap**  
show me the **code** \_

\* no devops were harmed while preparing the demo

# Ansible building blocks

```
---  
- hosts: app  
  tasks:  
    - name: ensure wget is installed  
      apt: name=nginx state=present  
- hosts: monitoring  
  tasks:  
    - name: ensure logging.conf is in place  
      copy: src=logging.conf dest=/opt/logging
```

Playbook

Play

Task

# Ansible built-in modules

## Module Index

- [All Modules](#)
- [Cloud Modules](#)
- [Clustering Modules](#)
- [Commands Modules](#)
- [Crypto Modules](#)
- [Database Modules](#)
- [Files Modules](#)
- [Identity Modules](#)
- [Infrastructure Modules](#)
- [Inventory Modules](#)
- [Messaging Modules](#)
- [Monitoring Modules](#)
- [Network Modules](#)
- [Notification Modules](#)
- [Packaging Modules](#)
- [Remote Management Modules](#)
- [Source Control Modules](#)
- [Storage Modules](#)
- [System Modules](#)
- [Univention Modules](#)
- [Utilities Modules](#)
- [Web Infrastructure Modules](#)
- [Windows Modules](#)

800+ built-in modules



# Not all is well - pain points

- Ansible-galaxy, not the best registry I've seen in my career
- Community roles and their quality
- Advanced debugging and testing roles, heavy dry-run
- You may need some Python eventually
- Community extensions may be buggy (e.g. docker)
- Variables - “cognitive overload”

# Give it a try

- Predictability via automation
- Infrastructure as code, you can't beat it
- Availability - every single \*nix machine with SSH
- Tons of built-in modules and community roles
- Easy to start with
- Readable even after long time

Seriously. **5 minutes.**

\* 5 minutes represent “think” not “react”

<https://signalvnoise.com/posts/3124-give-it-five-minutes>

# FIN

PLAY RECAP

\*\*\*\*\*

**BuildStuff:**            **ok=1**    **changed=1**        **unreachable=0**        **failed=0**