

// TRAINING AS CODE

REMOTE-WORKSHOPS UND SCHULUNGEN

// WHOAMI



- > Christian Stankowic
- > Senior System Engineer und Technical Leader bei [SVA](#)
- > Linux, DevOps, Infrastructure as Code, Trainer

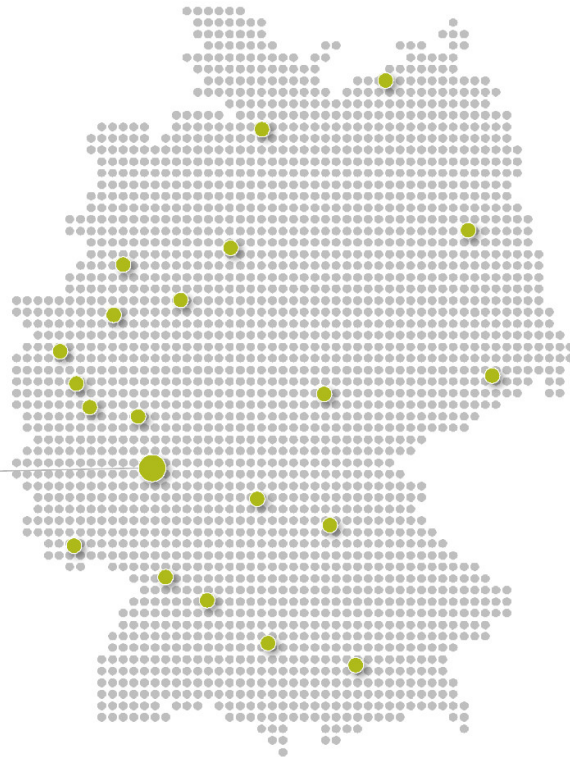
Über uns

/ Unternehmen

21

Lokationen

Wiesbaden



6 TOP

Branchen



Automotive



Retail



Public



Maschinen &
Anlagenbau



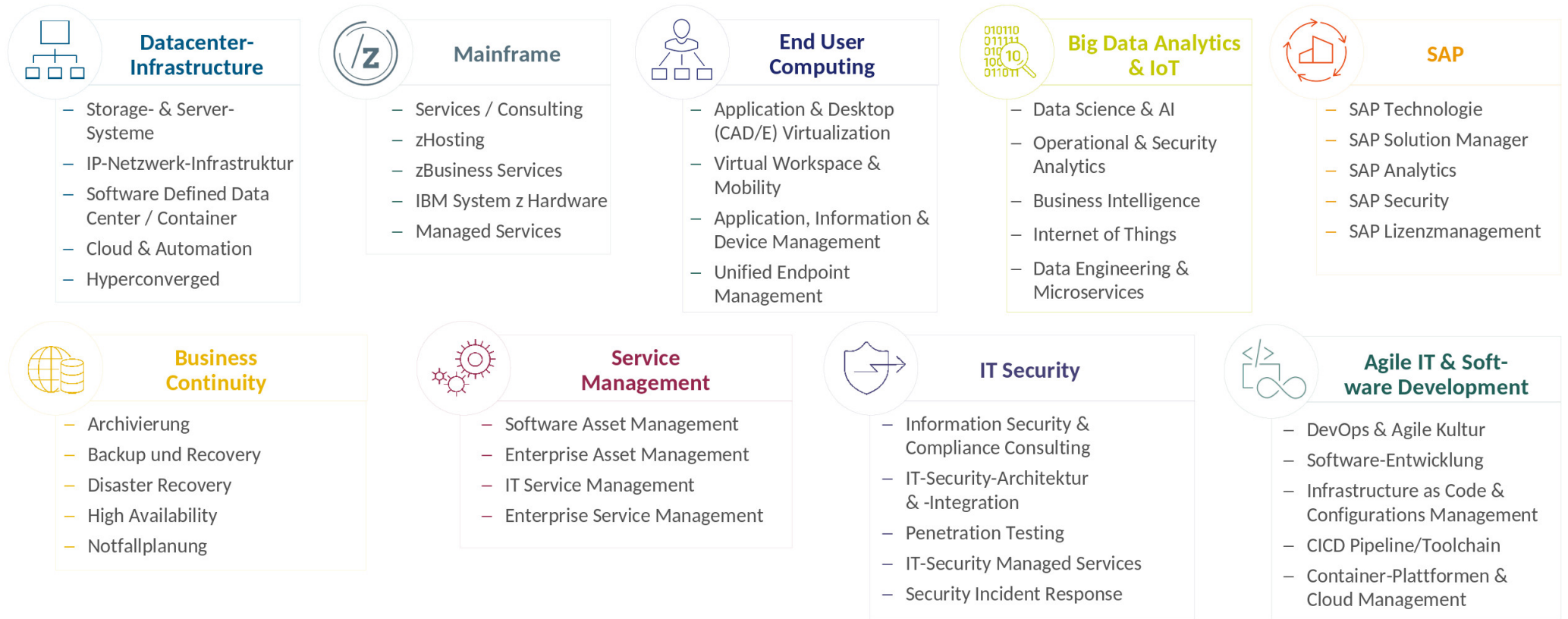
Finance &
Insurance



Telekommuni-
kation

Produkte

/ Produktportfolio auf einen Blick



// AGENDA

1. Motivation
2. Qual der Wahl: Meeting-Plattformen
3. Teil 1: Werkzeuge für Schulungsunterlagen
4. Teil 2: Automatisierte Infrastruktur
5. Teil 3: Testing für Trainer und Teilnehmer
6. Training as Code als Pipeline
7. Fazit

// MOTIVATION

MOTIVATION

> IT ist **kurzlebig**, immer **mehr** Infrastruktur in **kürzerer** Zeit

MOTIVATION

- > IT ist **kurzlebig**, immer **mehr** Infrastruktur in **kürzerer** Zeit
- > Während die Halbwertszeit von Fachwissen bei ca. 5 Jahren liegt, kommt IT-Fachwissen auf **knapp 1,5 Jahre***
- > Neues Wissen sollte schnellstmöglich adaptiert werden

MOTIVATION

- > IT ist **kurzlebig**, immer **mehr** Infrastruktur in **kürzerer** Zeit
- > Während die Halbwertszeit von Fachwissen bei ca. 5 Jahren liegt, kommt IT-Fachwissen auf **knapp 1,5 Jahre***
- > Neues Wissen sollte schnellstmöglich adaptiert werden
- > Mitarbeiter sollen über **aktuelles Wissen** verfügen aber dennoch **profitabel** sein
- > Diese Auswahl an Werkzeugen ist (zu) groß

*Klassiker ausgenommen, Quelle: [BIBB](#)

// MEETING-PLATTFORMEN

WOZU EINE MEETING-PLATTFORM?

- > Spätestens seit Corona **essentiell** um Termine wahrzunehmen
- > Auch ein guter **Anlass** um Schulungskonzept zu überdenken:

WOZU EINE MEETING-PLATTFORM?

- > Spätestens seit Corona **essentiell** um Termine wahrzunehmen
- > Auch ein guter **Anlass** um Schulungskonzept zu überdenken:
 - > Reisekosten und **Zeitersparnis**
 - > Logistische Planung (*Räume, Verpflegung*)
 - > Schulungsmethoden (*Tools, Testumgebung*)

WOZU EINE MEETING-PLATTFORM?

- > Spätestens seit Corona **essentiell** um Termine wahrzunehmen
- > Auch ein guter **Anlass** um Schulungskonzept zu überdenken:
 - > Reisekosten und **Zeitersparnis**
 - > Logistische Planung (*Räume, Verpflegung*)
 - > Schulungsmethoden (*Tools, Testumgebung*)
- > Meeting-Plattform wird benötigt, um Termine mit mehreren Teilnehmern zu planen und durchzuführen

AUSWAHL

- > Die Marktauswahl ist groß, jedes Tool hat inzwischen nahezu **deckungsgleiche** Features
- > Meeting-Bingo: jeder Termin auf einer neuen Plattform

AUSWAHL

- > Die Marktauswahl ist groß, jedes Tool hat inzwischen nahezu **deckungsgleiche** Features
- > Meeting-Bingo: jeder Termin auf einer neuen Plattform
- > Die üblichen Verdächtigen:
 - > Microsoft Teams
 - > Cisco WebEx
 - > GoTo Meeting
 - > BigBlueButton
 - > Zoom
 - > Jitsi



Meeting-Plattformen (Symbolbild)

GROBER VERGLEICH

| Name | Lizenz | On-prem | Clients | iOS, Android |
|-------|------------|---------|-----------------------------|--------------|
| Teams | Proprietär | Nein | Windows, macOS, Linux*, Web | Ja |
| WebEx | Proprietär | Ja | Windows, HTML5 | Ja |
| GoTo | Proprietär | Ja | Windows, macOS, Chrome | Ja |
| BBB | GPL | Ja | HTML5 | Nein |
| Zoom | Proprietär | Ja | Windows, macOS, Linux | Ja |
| Jitsi | Apache | Ja | HTML5 | Ja |

Features:

- > Desktop-Sharing, Präsentationsmodus, Downloads
- > Chat, Recordings

*Alpha-Version

LESSONS LEARNED FÜR LINUX (YMMV)

- > Teams funktional solange **Microsoft-Cloud** keine Probleme hat
 - > **Alpha**-Client funktioniert hervorragend, einige wenige Funktionen fehlen (*Webcam-Hintergrundbilder*)

LESSONS LEARNED FÜR LINUX (YMMV)

- > Teams funktional solange **Microsoft-Cloud** keine Probleme hat
 - > **Alpha**-Client funktioniert hervorragend, einige wenige Funktionen fehlen (*Webcam-Hintergrundbilder*)
- > Cisco WebEx via Browser, jedoch **eingeschränkte UX**
 - > z.B. Webcams können nicht fixiert werden

LESSONS LEARNED FÜR LINUX (YMMV)

- > Teams funktional solange **Microsoft-Cloud** keine Probleme hat
 - > **Alpha**-Client funktioniert hervorragend, einige wenige Funktionen fehlen (*Webcam-Hintergrundbilder*)
- > Cisco WebEx via Browser, jedoch **eingeschränkte UX**
 - > z.B. Webcams können nicht fixiert werden
- > GoTo Meeting funktioniert ausschließlich unter **Chrome** bzw. Chromium, oftmals Verbindungsabbrüche nach 1 Stunde

LESSONS LEARNED FÜR LINUX (YMMV)

- > Teams funktional solange **Microsoft-Cloud** keine Probleme hat
 - > **Alpha**-Client funktioniert hervorragend, einige wenige Funktionen fehlen (*Webcam-Hintergrundbilder*)
- > Cisco WebEx via Browser, jedoch **eingeschränkte UX**
 - > z.B. Webcams können nicht fixiert werden
- > GoTo Meeting funktioniert ausschließlich unter **Chrome** bzw. Chromium, oftmals Verbindungsabbrüche nach 1 Stunde
- > BBB und Jitsi i.d.R. hervorragend sofern der Server **angemessen dimensioniert** ist
 - > siehe auch [Jitsi DevOps Guide](#)
 - > Firefox je nach Jitsi-Version problematisch, Chromium

"ZOOM.US" REQUESTS ROOT ACCESS



SEEMS REASONABLE I THINK

imgflip.com

// TEIL 1

TOOLING

ANFORDERUNGEN

- > Einfach zu benutzen, **geringe Hürde** für neue Trainer
- > **Keine** besonderen Tools zur Verwendung/Erstellung

ANFORDERUNGEN

- > Einfach zu benutzen, **geringe Hürde** für neue Trainer
- > **Keine** besonderen Tools zur Verwendung/Erstellung
- > Themes für Corporate Identity
- > Definition im Klartext, keine proprietären Formate

ANFORDERUNGEN

- > Einfach zu benutzen, **geringe Hürde** für neue Trainer
- > **Keine** besonderen Tools zur Verwendung/Erstellung
- > Themes für Corporate Identity
- > Definition im Klartext, keine proprietären Formate
- > muss **offline** ohne Server benutzbar sein
- > Präsentatoransicht mit Timer

ANFORDERUNGEN

- > Einfach zu benutzen, **geringe Hürde** für neue Trainer
- > **Keine** besonderen Tools zur Verwendung/Erstellung
- > Themes für Corporate Identity
- > Definition im Klartext, keine proprietären Formate
- > muss **offline** ohne Server benutzbar sein
- > Präsentatoransicht mit Timer
- > Syntax-**Highlighting** für verschiedene Sprachen
- > **PDF**-Export

KLASSISCHE PRÄSENTATIONSPROGRAMME

- > Benutzen bereits **vorhandener** Vorlagen mit
 - > Microsoft PowerPoint
 - > LibreOffice Impress
 - > FreeOffice / SoftMaker Office Presentations

KLASSISCHE PRÄSENTATIONSPROGRAMME

- > Benutzen bereits **vorhandener** Vorlagen mit
 - > Microsoft PowerPoint
 - > LibreOffice Impress
 - > FreeOffice / SoftMaker Office Presentations
- > **Einfache** Umsetzung, bekannter Workflow, Präsentatoransicht
- > Änderungen in Versionskontrolle **intransparent** da Binärdatei*
- > Große Probleme mit 200+ Folien und **hochauflösenden** Fotos

KLASSISCHE PRÄSENTATIONSPROGRAMME

- > Benutzen bereits **vorhandener** Vorlagen mit
 - > Microsoft PowerPoint
 - > LibreOffice Impress
 - > FreeOffice / SoftMaker Office Presentations
- > **Einfache** Umsetzung, bekannter Workflow, Präsentatoransicht
- > Änderungen in Versionskontrolle **intransparent** da Binärdatei*
- > Große Probleme mit 200+ Folien und **hochauflösenden** Fotos
- > *Das kannst du schon so machen, aber...*

*PPTX = zip + XML

TEXTSATZSYSTEME

- > Verwendung von **TeX** bzw. **LaTeX**
- > Sehr **mächtig** und vielseitig

TEXTSATZSYSTEME

- > Verwendung von **TeX** bzw. **LaTeX**
- > Sehr **mächtig** und vielseitig
- > [Viele Präsentationsvorlagen](#) online verfügbar
- > vollständige TeX-Umgebung benötigt (4 GB+)

TEXTSATZSYSTEME

- > Verwendung von **TeX** bzw. **LaTeX**
- > Sehr **mächtig** und vielseitig
- > [Viele Präsentationsvorlagen](#) online verfügbar
- > vollständige TeX-Umgebung benötigt (*4 GB+*)
- > spezieller Viewer für Präsentatoransicht benötigt*
- > steile Lernkurve, definitiv **ungeeignet** für Einsteiger

*z.B. [pdfpc](#), [BeamerPresenter](#)

TEXTSATZSYSTEME

- > Verwendung von **TeX** bzw. **LaTeX**
- > Sehr **mächtig** und vielseitig
- > [Viele Präsentationsvorlagen](#) online verfügbar
- > vollständige TeX-Umgebung benötigt (4 GB+)
- > spezieller Viewer für Präsentatoransicht benötigt*
- > steile Lernkurve, definitiv **ungeeignet** für Einsteiger

- > *Wie mache ich mein Leben spannender?*

*z.B. [pdfpc](#), [BeamerPresenter](#)

TEXTSATZSYSTEME

```
\begin{frame}
  \frametitle{Lists}

  \begin{itemize}
    \item
      Bullet lists are marked with a red \textit{box}.
  \end{itemize}

  \begin{enumerate}
    \item
      Numbered lists are marked with a \textit{white} number
  \end{enumerate}

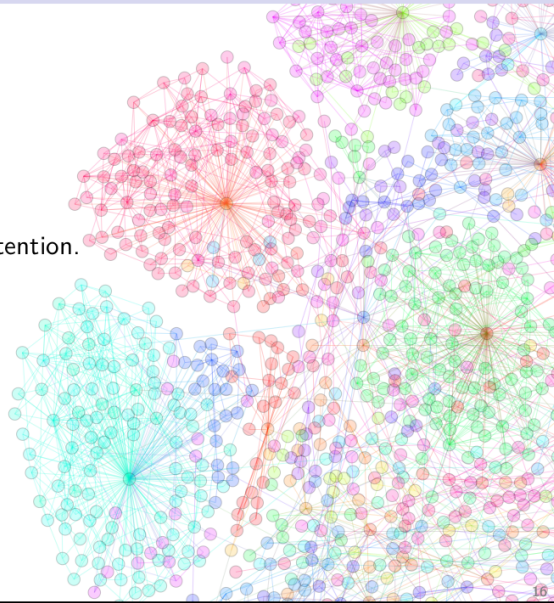
\end{frame}
```

Beispielhafter LaTeX-Code

TEXTSATZSYSTEME

QUESTIONS?

Thank you for your attention.



REFERENCES

- Newman, Mark E. J. "Fast algorithm for detecting community structure in networks". In: *Physical Review E* 69 (6 June 2004), p. 066133. DOI: 10.1103/PhysRevE.69.066133. URL: <http://link.aps.org/doi/10.1103/PhysRevE.69.066133>.
- Takáč, Luboš and Michal Záborský. "Data Analysis in Public Social Networks". In: *International Scientific Conference & International Workshop. Present Day Trends of Innovations*. Lomza, Poland, May 2012.

Talk about the coffee break

17



00:00:22

16/17

pdfpc Presenter

PREZI

- > Plattformunabhängiges Präsentationsprogramm
- > Nutzt HTML5-Technik

PREZI

- > Plattformunabhängiges Präsentationsprogramm
- > Nutzt HTML5-Technik
- > Inhalte und Effekte auf großem virtuellen Blatt Papier
- > **Abhängig** von Cloud des Anbieters

PREZI

- > Plattformunabhängiges Präsentationsprogramm
- > Nutzt HTML5-Technik
- > Inhalte und Effekte auf großem virtuellen Blatt Papier
- > **Abhängig** von Cloud des Anbieters
- > kann auch offline abgespielt werden
- > **Proprietäres** Datei-Format, Konvertierung schwierig
- > Nur bedingt für **Schulungen** nutzbar



Prezi-Beispiel: [Filmreich präsentieren auf Deutsch](#)

WEB-FRAMEWORKS

Zu den gängigsten Frameworks* zählen u.a.:

- > [reveal.js](#), [Hacker Slides](#)
- > [flowtime.js](#)
- > [deck.js](#)
- > [DZSlides](#)
- > [Puppetlabs Showoff](#)
- > [remark](#)

* neues Framework in 3, 2,...

REVEAL.JS

- > Open Source HTML-Framework für Präsentationen
- > ansprechendes Design, Animationen, Präsentator-Ansicht
- > zahlreiche Themes und [Plugins](#) verfügbar

REVEAL.JS

- > Open Source HTML-Framework für Präsentationen
- > ansprechendes Design, Animationen, Präsentator-Ansicht
- > zahlreiche Themes und [Plugins](#) verfügbar
- > [Markdown-Support](#) nicht im Offline-Modus (*Server benötigt*)
- > Trainer müssen i.d.R. HTML-Code schreiben > **Vorkenntnisse**

REVEAL.JS

- > Open Source HTML-Framework für Präsentationen
- > ansprechendes Design, Animationen, Präsentator-Ansicht
- > zahlreiche Themes und [Plugins](#) verfügbar
- > [Markdown-Support](#) nicht im Offline-Modus (*Server benötigt*)
- > Trainer müssen i.d.R. HTML-Code schreiben > **Vorkenntnisse**
- > **Hacker Slides** kombiniert reveal.js mit [Ace Editor](#)
 - > interessant, jedoch technisch veraltet



THE HTML PRESENTATION FRAMEWORK

Created by [Hakim El Hattab](#) and [contributors](#)



PUPPETLABS SHOWOFF

- > Vollumfängliches Trainingstool
- > Inhalte werden in **Markdown** definiert

PUPPETLABS SHOWOFF

- > Vollumfängliches Trainingstool
- > Inhalte werden in **Markdown** definiert
- > Zahlreiche Zusatzfunktionen
 - > Agenda, Datei-Ablage
 - > Notizen und **Handouts**
 - > **Quiz** am Kapitelende
 - > Slide-**Bugtracker**
 - > Teilnehmer-**Chat**

PUPPETLABS SHOWOFF

- > Vollumfängliches Trainingstool
- > Inhalte werden in **Markdown** definiert
- > Zahlreiche Zusatzfunktionen
 - > Agenda, Datei-Ablage
 - > Notizen und **Handouts**
 - > **Quiz** am Kapitelende
 - > Slide-**Bugtracker**
 - > Teilnehmer-**Chat**
- > Entwicklung **stockt**, zahlreiche unbehandelte Bugs

Annotations Report Issue With Slide Viewing Statistics Downloads Display Window Print Slides Settings

Timer: min Start

PACE

- Course Overview
- Classroom Environment
- Basics Review
- Using Puppet Data
- Resources
- Language Constructs
 - 94. Language Constructs
 - 95. Lesson 5: Language ...
 - 96. What is a Data Type?**
 - 97. Puppet 4 Core Data ...
 - 98. Arrays as Resource T...
 - 99. Storing Data in Hashes
 - 100. Complex Data Stru...
- Audience Questions

What is a Data Type?

The type of data a value refers to.

- A piece of data is called a value.
- Every value has a data type that affects how it is used.

| Value | Type | |
|---------|---------|----------------------------------|
| 644 | Number | No quotes |
| 0644 | Number | Octal value equal to decimal 420 |
| '0644' | String | Quoted string |
| false | Boolean | |
| 'false' | String | Quoted string |

```
[root@training]# puppet apply -e 'notice(0644)'
Notice: Scope(Class[main]): 420
```

Slide: 96/335 - 29% [Language Constructs/values_and_datatypes](#)

notes handouts

This example is particularly relevant, because it explains why the `file` type complains if you pass in a bare numeric value. It's so the mode doesn't get interpreted as octal!

REMARK

- > HTML-Präsentator auf Markdown-Basis
- > **simpler** Aufbau, einfach anpassbar

REMARK

- > HTML-Präsentator auf Markdown-Basis
- > **simpler** Aufbau, einfach anpassbar
- > Präsentatoransicht mit Timer
- > Syntax-Highlightning für verschiedene Sprachen

REMARK

- > HTML-Präsentator auf Markdown-Basis
- > **simpler** Aufbau, einfach anpassbar
- > Präsentatoransicht mit Timer
- > Syntax-Highlightning für verschiedene Sprachen
- > PDF-Export via [DeckTape](#)
- > **geringe Einarbeitungszeit** durch Markdown

REMARK

```
# Training as Code
```

```
## Remote-Workshops und Schulungen
```

```
---
```

```
# Whoami
```

```
![:img Christian Stankowic, 25%] (assets/imgs/cstankow.jpg)
```

- Christian Stankowic
- Senior System Engineer und Technical Leader bei [SVA] (<https://www.sva.ch>)
- Linux, DevOps, Infrastructure as Code, Trainer

```
---
```

Erste zwei Folien dieser Präsentation

// TEIL 2

AUTOMATISIERTE INFRASTRUKTUR

AUTOMATISIERTE INFRASTRUKTUR

- > Schulungen bestehen i.d.R. aus einer **Testumgebung**
 - > verschiedene VMs
 - > IP-Adressen, Netzwerke
 - > vorinstalliertes **Betriebssystem**
 - > **vorkonfigurierte** Applikationen

AUTOMATISIERTE INFRASTRUKTUR

- > Schulungen bestehen i.d.R. aus einer **Testumgebung**
 - > verschiedene VMs
 - > IP-Adressen, Netzwerke
 - > vorinstalliertes **Betriebssystem**
 - > **vorkonfigurierte** Applikationen
- > Teilnehmer erwarten **sofort** loslegen zu können
- > Spätere Experimente sollen jedoch auch möglich sein

AUTOMATISIERTE INFRASTRUKTUR

- > Testumgebungen kosten Geld
- > Definition von **Verfügbarkeit** oftmals nicht eindeutig

AUTOMATISIERTE INFRASTRUKTUR

- > Testumgebungen kosten Geld
- > Definition von **Verfügbarkeit** oftmals nicht eindeutig
- > **Interne** Schulungen auf eigener Infrastruktur
- > Zugriff für **externe** Kunden nicht möglich und erwünscht

AUTOMATISIERTE INFRASTRUKTUR

- > Testumgebungen kosten Geld
- > Definition von **Verfügbarkeit** oftmals nicht eindeutig
- > **Interne** Schulungen auf eigener Infrastruktur
- > Zugriff für **externe** Kunden nicht möglich und erwünscht
- > Schulungsumgebungen mittelfristig für Teilnehmer aufrecht erhalten ist **unpraktikabel** und teuer
- > Wo das Ganze hosten?

AUTOMATISIERTE INFRASTRUKTUR

- > Testumgebungen kosten Geld
- > Definition von **Verfügbarkeit** oftmals nicht eindeutig
- > **Interne** Schulungen auf eigener Infrastruktur
- > Zugriff für **externe** Kunden nicht möglich und erwünscht
- > Schulungsumgebungen mittelfristig für Teilnehmer aufrecht erhalten ist **unpraktikabel** und teuer
- > Wo das Ganze hosten?
- > **Lösung:** Vagrant, \$Cloud + Terraform

INFRASTRUCTURE AS CODE

- > Schulungsumgebung wird **automatisiert** erstellt
- > tatsächliche Plattform ist **irrelevant**
 - > lokale Testumgebung
 - > Cloud-Anbieter
 - > Teilnehmer-Notebook

INFRASTRUCTURE AS CODE

- > Schulungsumgebung wird **automatisiert** erstellt
- > tatsächliche Plattform ist **irrelevant**
 - > lokale Testumgebung
 - > Cloud-Anbieter
 - > Teilnehmer-Notebook
- > Definition des **Soll-Zustands** der benötigten Ressourcen
 - > VMs, Netzwerke, OS, Applikation,...
- > **Ansible** kümmert sich um die Implementation

HASHICORP VAGRANT

- > Vagrant ermöglicht das Erstellen von virtuellen Umgebungen
 - > Netzwerke, VMs, Anwendungen,...
- > entwickelt in **Ruby**, zahlreiche Plugins verfügbar

HASHICORP VAGRANT

- > [Vagrant](#) ermöglicht das Erstellen von virtuellen Umgebungen
 - > Netzwerke, VMs, Anwendungen,...
- > entwickelt in **Ruby**, zahlreiche [Plugins](#) verfügbar
- > Erstellt VMs aus [Vorlagen](#), führt IaC-Code aus,...
- > **Reproduzierbare** Test-/Entwicklungsumgebungen
 - > kein "*It works on my machine*" (**IWOMM**) mehr

HASHICORP VAGRANT

- > [Vagrant](#) ermöglicht das Erstellen von virtuellen Umgebungen
 - > Netzwerke, VMs, Anwendungen,...
- > entwickelt in **Ruby**, zahlreiche [Plugins](#) verfügbar
- > Erstellt VMs aus [Vorlagen](#), führt IaC-Code aus,...
- > **Reproduzierbare** Test-/Entwicklungsumgebungen
 - > kein "*It works on my machine*" (**IWOMM**) mehr
- > Einfach zu handhaben, hoher Automatisierungsgrad
- > Definition erfolgt über einzelne Datei (`Vagrantfile`)

HASHICORP VAGRANT

- > Unterstützte Hypervisor (**Provider**)
 - > VirtualBox
 - > VMware Workstation und Fusion (**80 USD**)
 - > Hyper-V
 - > XEM, KVM,...

HASHICORP VAGRANT

- > Unterstützte Hypervisor (**Provider**)
 - > VirtualBox
 - > VMware Workstation und Fusion (**80 USD**)
 - > Hyper-V
 - > XEM, KVM,...
- > Unterstützte IaC-Lösungen (**Provisioner**)
 - > Ansible
 - > Chef
 - > Puppet
 - > Salt

HASHICORP VAGRANT

```
# deploy ansible VM
Vagrant.configure("2") do |config|
  ansible.vm.hostname = "ansible.sva.de"
  # use CentOS/8 box
  ansible.vm.box = "generic/centos8"
  # define additional private network
  ansible.vm.network "private_network", ip: "192.168.0.10"
  # prepare node via Ansible
  ansible.vm.provision "ansible_local" do |ansible|
    ansible.playbook = "ansible/node.yml"
  end
end
```

Bereitstellen einer CentOS-8 VM, Konfiguration via Ansible

HASHICORP TERRAFORM

- > [Terraform](#) verwaltet komplexe (*produktive*) Infrastruktur
- > Sinnvoll für mehrere unterschiedliche Plattformen
 - > z.B. mehrere On-Premise Hypervisor + Cloud

HASHICORP TERRAFORM

- > [Terraform](#) verwaltet komplexe (*produktive*) Infrastruktur
- > Sinnvoll für mehrere unterschiedliche Plattformen
 - > z.B. mehrere On-Premise Hypervisor + Cloud
- > Infrastruktur-Definition in **HCL** (*HashiCorp Configuration Language*) oder JSON
- > Sehr große Anzahl verfügbarer Integrationen*

HASHICORP TERRAFORM

- > [Terraform](#) verwaltet komplexe (*produktive*) Infrastruktur
- > Sinnvoll für mehrere unterschiedliche Plattformen
 - > z.B. mehrere On-Premise Hypervisor + Cloud
- > Infrastruktur-Definition in **HCL** (*HashiCorp Configuration Language*) oder JSON
- > Sehr große Anzahl verfügbarer Integrationen*
- > Integration in gängiges Configuration Management
- > Prinzip: **D.R.Y** (*Don't repeat yourself*)

*ca. [100 Erweiterungen](#)

HASHICORP TERRAFORM

- > **Cloud:** AWS, Azure, GCE, vCloud, DigitalOcean, Hetzner, 1&1, Nutanix, OpenStack
- > **Infrastruktur:** Docker, k8s, Helm, Rancher, Vault, Chef, Cobbler
- > **Netzwerk:** Cisco, FortiOS, Palo Alto, F5 BIG-IP, Power/UltraDNS, DNSimple
- > **Datenbanken:** MySQL, PostgreSQL, InfluxDB, Grafana
- > **Monitoring:** Icinga2, Datadog, Runscope
- > **Community:** Actice Directory, Elasticsearch, GitHub, GitLab, Infoblox, JIRA, Proxmox, oVirt
- > ...

HASHICORP TERRAFORM

- > Konfiguration besteht aus aufeinander **aufbauenden** Ressourcen
 - > z.B. **VM** => Compute Resource, Netzwerk, Storage,...

HASHICORP TERRAFORM

- > Konfiguration besteht aus aufeinander **aufbauenden** Ressourcen
 - > z.B. **VM** => Compute Resource, Netzwerk, Storage,...
- > Blöcke können durch **Variablen** modular gehalten werden
- > Terraform hat stets die Kenntnis über Ist-/Soll-Zustand => `terraform.tfstate`

HASHICORP TERRAFORM

- > Konfiguration besteht aus aufeinander **aufbauenden** Ressourcen
 - > z.B. **VM** => Compute Resource, Netzwerk, Storage,...
- > Blöcke können durch **Variablen** modular gehalten werden
- > Terraform hat stets die Kenntnis über Ist-/Soll-Zustand => `terraform.tfstate`
- > **Workflow:**
 - > `TERRAFORM VALIDATE`
 - > `TERRAFORM INIT`
 - > `TERRAFORM PLAN`
 - > `TERRAFORM APPLY / DESTROY`

// TEIL 3: TESTING FÜR TRAINER UND TEILNEHMER

TESTING

- > Schulungen bestehen aus Aufgaben um Interaktivität zu fördern
 - > Teilnehmer sollen **selbstständig** Richtigkeit überprüfen

TESTING

- > Schulungen bestehen aus Aufgaben um Interaktivität zu fördern
 - > Teilnehmer sollen **selbstständig** Richtigkeit überprüfen
- > Cloud-Schulungsumgebungen **müssen** getestet werden
 - > Wartungsarbeiten des Anbieters, neue Features/**Fehler**

TESTING

- > Schulungen bestehen aus Aufgaben um Interaktivität zu fördern
 - > Teilnehmer sollen **selbstständig** Richtigkeit überprüfen
- > Cloud-Schulungsumgebungen **müssen** getestet werden
 - > Wartungsarbeiten des Anbieters, neue Features/**Fehler**
- > Trainer fügt neue Inhalte und **Musterlösungen** hinzu
 - > **Syntax**-Fehler sollen vermieden werden

FRAMEWORK

Denkbare Frameworks:

- > Deployment:
 - > WIP: [terratest](#) für Terraform-Deployment

FRAMEWORK

Denkbare Frameworks:

- > Deployment:
 - > WIP: [terratest](#) für Terraform-Deployment
- > Betriebssystem / Anwendung
 - > Ansible `assert`
 - > [Testinfra](#)

FRAMEWORK

Denkbare Frameworks:

- > Deployment:
 - > WIP: [terratest](#) für Terraform-Deployment
- > Betriebssystem / Anwendung
 - > Ansible `assert`
 - > [Testinfra](#)
- > Musterlösungen / Aufgaben
 - > [Goss](#)
 - > [Commander](#)

TESTINFRA

```
def test_software(host):  
    """  
    Check whether required software packages are installed  
    """  
    packages = [  
        "vim-common",  
        "bind-utils",  
        "epel-release",  
        "ansible"  
    ]  
    for pkg in packages:  
        _pkg = host.package(pkg)  
        assert _pkg.is_installed
```

Unittest überprüft installierte Software-Pakete

COMMANDER

- > In Go geschriebenes Testing-Tool
- > **Unabhängig** von verwendeter Skript-/Programmiersprache

COMMANDER

- > In Go geschriebenes Testing-Tool
- > **Unabhängig** von verwendeter Skript-/Programmiersprache
- > Tests werden in **YAML** definiert
- > Unterstützt verschiedene Ziele
 - > localhost
 - > SSH-Hosts
 - > Docker-Container

COMMANDER

- > In Go geschriebenes Testing-Tool
- > **Unabhängig** von verwendeter Skript-/Programmiersprache
- > Tests werden in **YAML** definiert
- > Unterstützt verschiedene Ziele
 - > localhost
 - > SSH-Hosts
 - > Docker-Container
- > schnell zu schreiben und zu verstehen

COMMANDER: BEISPIEL

```
tests:  
  lab01:  
    command: git status  
    stdout: "is up to date"  
    exit-code: 0
```

keen.yml - Definition eines Tests

```
$ commander test keen.yml  
Starting test file keen.yamll...  
  
✓ lab01  
  
Duration: 0.002s  
Count: 1, Failed: 0
```

Ausführen des Tests

LINTING

> Mithilfe eines **Lint**-Tools (*englisch für Fusse!*) kann eine **statische Code-Analyse** durchgeführt werden

LINTING

- > Mithilfe eines **Lint**-Tools (*englisch für Fusse!*) kann eine **statische Code-Analyse** durchgeführt werden
- > Tool überprüft Code auf **Syntax-** und **Design-Fehler**, z.B.:

LINTING

- > Mithilfe eines **Lint**-Tools (*englisch für Fusse!*) kann eine **statische Code-Analyse** durchgeführt werden
- > Tool überprüft Code auf **Syntax**- und **Design-Fehler**, z.B.:
 - > veraltete Formate und Funktionen
 - > schlechter Stil
 - > fehlerhafte Aufrufe

LINTING

- > Mithilfe eines **Lint**-Tools (*englisch für Fusse!*) kann eine **statische Code-Analyse** durchgeführt werden
- > Tool überprüft Code auf **Syntax**- und **Design-Fehler**, z.B.:
 - > veraltete Formate und Funktionen
 - > schlechter Stil
 - > fehlerhafte Aufrufe
- > Die ersten Linter entstanden 1979 um Schwächen des C-Compilers auszugleichen

LINTING

- > Mithilfe eines **Lint**-Tools (*englisch für Fusse!*) kann eine **statische Code-Analyse** durchgeführt werden
- > Tool überprüft Code auf **Syntax**- und **Design-Fehler**, z.B.:
 - > veraltete Formate und Funktionen
 - > schlechter Stil
 - > fehlerhafte Aufrufe
- > Die ersten Linter entstanden 1979 um Schwächen des C-Compilers auszugleichen
- > **Resultat**: optimierter und fehlerfreier Code

LINTING AS A SERVICE

Für Schulungsinhalte relevant:

- > [markdownlint](#) für Markdown-Code (sic!)
- > [ansible-lint](#) für Playbooks/Rollen
- > [yamllint](#) für YAML-Format
- > [flake8](#) für `testinfra`-Tests
- > [pylint](#) für Python-Code
- > [TFLint](#) für Terraform-Code
- > [rubocop](#) für Ruby-/Vagrant-Code

LINTING: SCOPE

- > Ziel soll es sein den Trainer auf **Best Practices** hinzuweisen
- > **Fehler** brechen die Pipeline, Warnungen nicht
- > Nicht jeder strebt perfekten Code an (*leider*)

LINTING: SCOPE

- > Ziel soll es sein den Trainer auf **Best Practices** hinzuweisen
- > **Fehler** brechen die Pipeline, Warnungen nicht
- > Nicht jeder strebt perfekten Code an (*leider*)
- > Mustervorlage ist fehlerfrei
- > Dokumentation erklärt typische Fehler
- > VScode-/VScodium-**Plugins** für Linting verfügbar
 - > [vscode-markdownlint](#)
 - > [vscode-yaml](#)
 - > [vscode-ruby](#)

```
README.md 02-platforms.md • 01-motivation.md 03-tooling.md 04-infrastructure.md 05-testing.md 06-pipeline.md 07-
02-platforms.md > ...
1 class: center, middle
2 MD018/no-missing-space-atx: No space after hash on atx style heading markdownlint(MD018)
3 Peek Problem (Alt+F8) Quick Fix... (Ctrl+.)
4 #Meeting-Plattformen
5
6 ---
7
8 ## Wozu eine Meeting-Plattform?
9
10 - Spätestens seit Corona **essentiell** um Termine wahrzunehmen
11 - Auch ein guter **Anlass** um Schulungskonzept zu überdenken:
12 --
13
14 | - Reisekosten und **Zeitersparnis**
15 | - Logistische Planung (*Räume, Verpflegung*)
16 | - Schulungsmethoden (*Tools, Testumgebung*)
17 --
18
```


// TRAINING AS CODE

ALS PIPELINE

PIPELINE

- > Pipeline kombiniert drei Teilschritte
 - > Erstellen von Schulungsunterlagen
 - > Testing (*Slides, Labs*)
 - > Infrastruktur bereitstellen (*und testen*)
- > MVP* erfolgt, weitere Schritte in Arbeit

*Minimum Viable Product

PDF-EXPORT

- > Mit [decktape](#) existiert ein Werkzeug zum Drucken von HTML-Präsentationen
- > Unterstützt zahlreiche Frameworks, u.a. remark
- > Geschrieben in JavaScript, Installation via `npm`

PDF-EXPORT

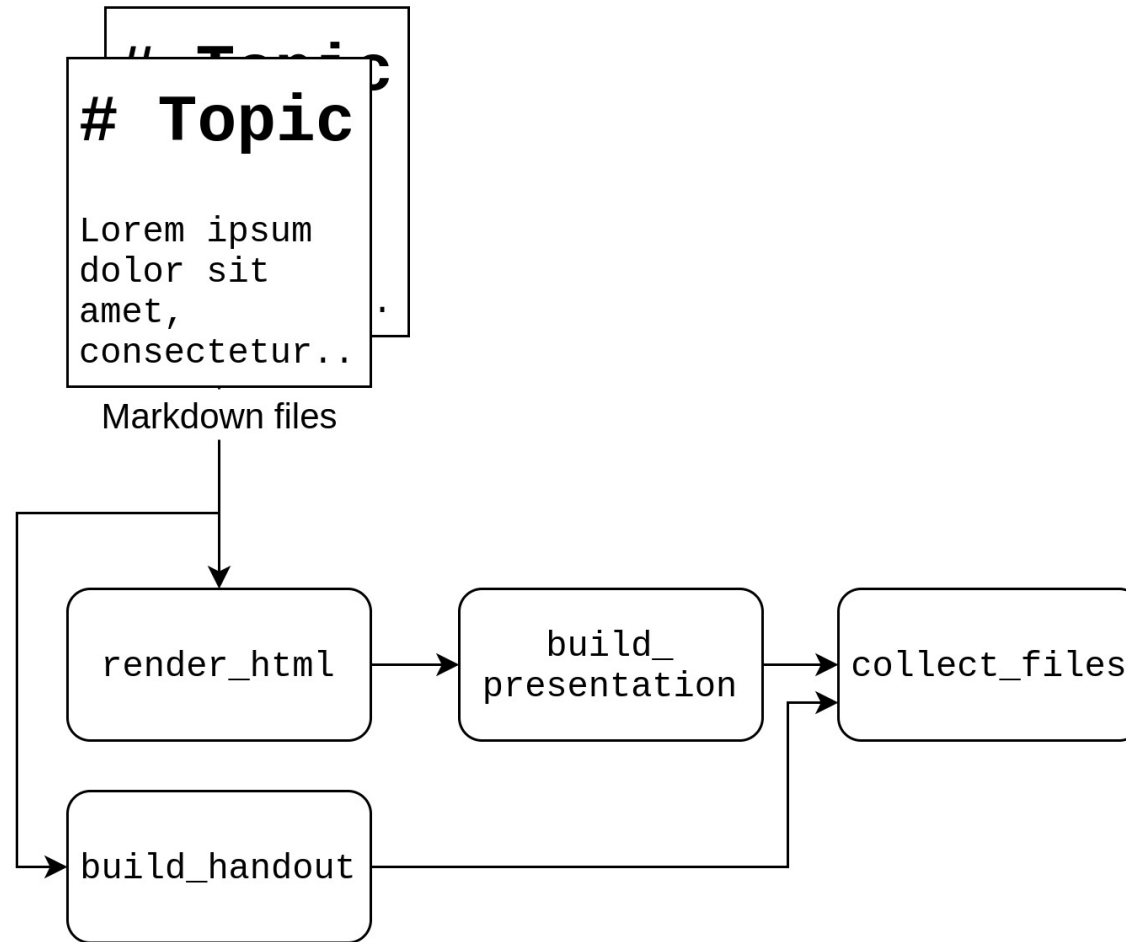
- > Mit [decktape](#) existiert ein Werkzeug zum Drucken von HTML-Präsentationen
- > Unterstützt zahlreiche Frameworks, u.a. remark
- > Geschrieben in JavaScript, Installation via `npm`
- > Glücklicherweise gibt es ein [vorgefertigtes Docker-Image](#)
- > einfache Integration in CI-/CD-Pipeline möglich

PDF-EXPORT

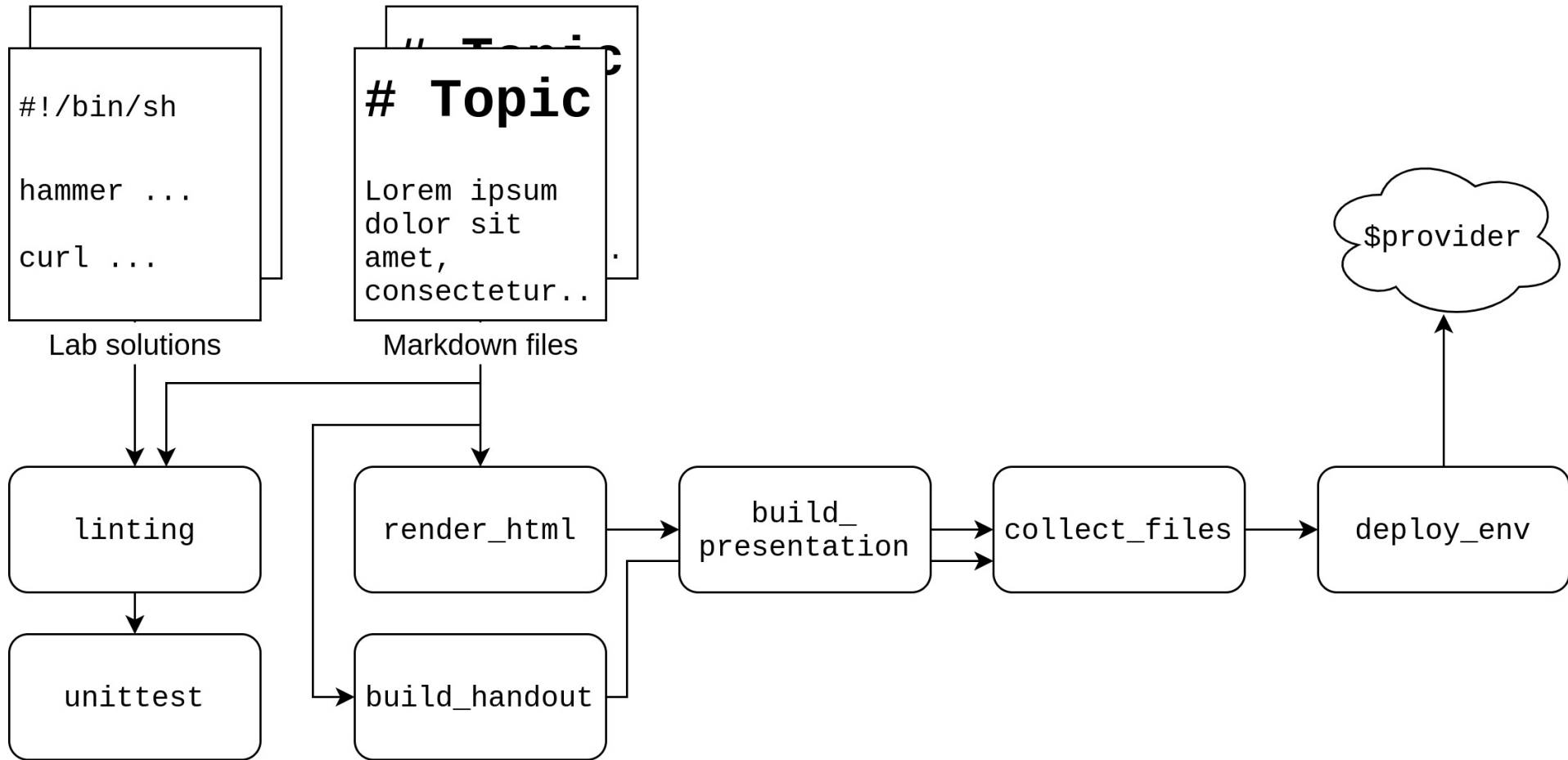
```
build presentation:
  stage: pdf
  image:
    name: astefanutti/decktape
    entrypoint: [""]
  script:
    - node /decktape/decktape.js --chrome-path chromium-browser --c
  artifacts:
    paths:
      * - presentation.pdf
        - presentation.html
```

`.gitlab-ci.yml` - Teil der CI-/CD-Konfiguration für PDF-Export

PIPELINE: IST-ZUSTAND



PIPELINE: SOLL-ZUSTAND



// FAZIT

FAZIT

- > [remark](#) eignet sich **hervorragend** als Präsentationswerkzeug
- > In Kombination mit einer Meeting-Plattform und Testumgebung auch für Schulungen geeignet
- > MVP wird bereits genutzt, Feedback **positiv**

FAZIT

- > [remark](#) eignet sich **hervorragend** als Präsentationswerkzeug
- > In Kombination mit einer Meeting-Plattform und Testumgebung auch für Schulungen geeignet
- > MVP wird bereits genutzt, Feedback **positiv**
- > Weitere Ideen:
 - > **Bugtracker**-Integration in Folien
 - > Weitere **Layouts** implementieren
 - > Code-Check der Folien bzw. Code dynamisch in Folien einfügen

CONTRIBUTE!

Ein Schulungsbeispiel findet sich auf GitHub:

<https://github.com/svalabs/training-as-code>

Feedback ist jederzeit willkommen!

LINKLISTE

- > [Jitsi DevOps Guide](#)
- > [Puppetlabs Showoff](#)
- > [remark](#)
- > [HashiCorp Vagrant](#)
- > [HashiCorp Terraform](#)
- > [Gruntwork.io Terratest](#)
- > [Testinfra](#)
- > [commander](#)

FRAGEN?

DANKE FÜR DIE AUFMERKSAMKEIT

