AVD Extended Workshop

Intro into Ansible, Ansible AVD, Git and VSCode for new and existing AVD users

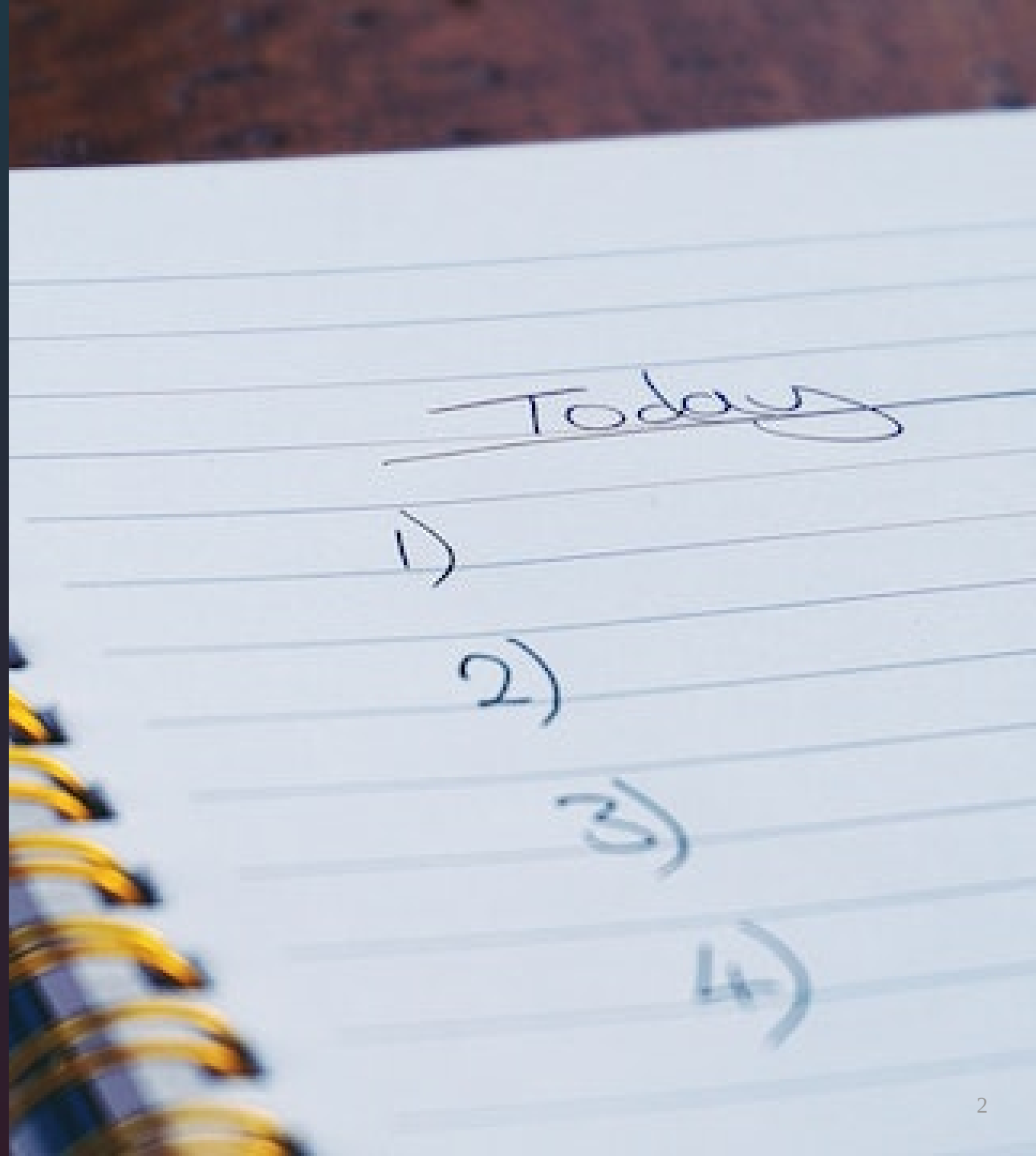
What is this Workshop about?

Topics:

- Git basics
- VSCode basics
- Ansible basics
- Containers basics and using container to build Ansible AVD environment
- Building simple L3LS network with Ansible AVD

Structure:

- This workshop is split into 3 sections. Each section takes around 2 hours to complete. That can be done as a full day workshop or split into 3 separate sessions.
- Make a break when you see a slide with a coffee cup ☕
- Ask questions at any time!



What is NOT covered in this Workshop?

- This workshop is not a deep dive into each and every topic. It is covering some advanced concepts, but you may need additional documentation and training to understand every topic in details.
  For additional information please refer to the following resources:
  - Ansible AVD Documentation
  - VSCode Documentation
  - Git Documentation - Pro Git book is a good start
  - Container Trainings by @jpetazzo:
    - Github repository
    - Training materials
- We are not going to use Arista CloudVision Portal (CVP) in this workshop. It provides a lot of advantages, but is not essential to understand the concepts covered in this workshop.
- If you will not find something you expect in this workshop, there can be 2 reasons:
  - It is not covered in this workshop
  - It is waiting for your contribution to this repository! 🤝

Requirements

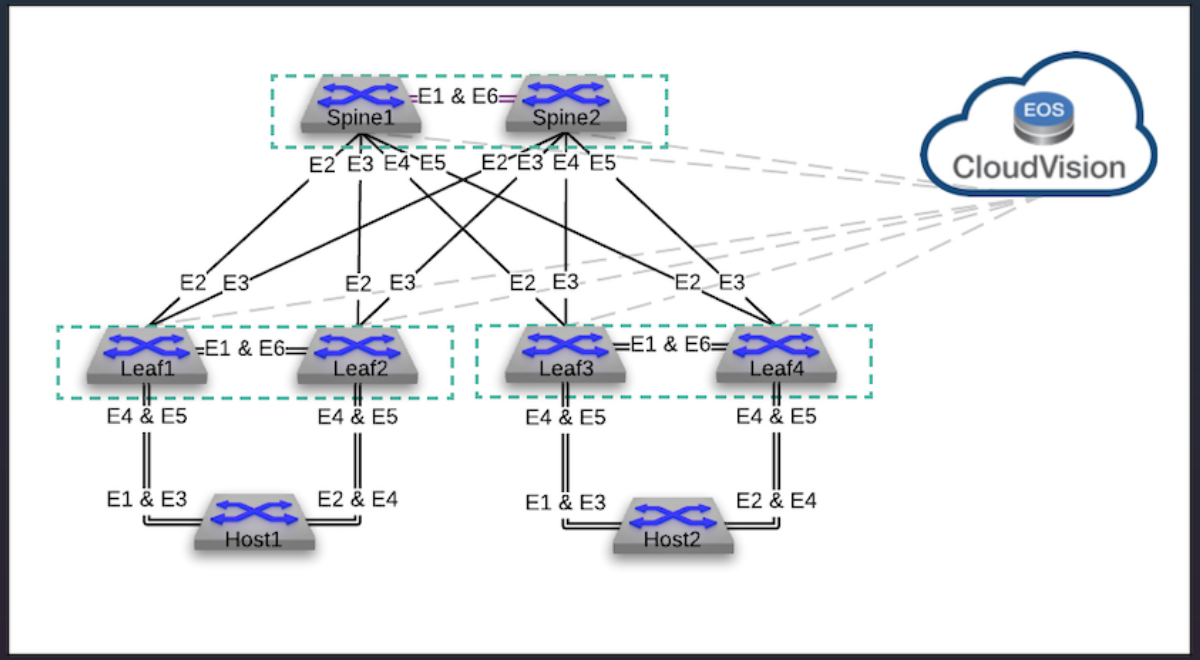
- You **MUST** have a Github account ❗
  Register here.

How to use this Workshop?

- To try all practical examples you need to have access to the lab environment. There are 3 possible options:
  - Use Github Codespaces. This is the preferred option, but double check that you understand all the costs and free tier limits.
  - Use Arista Test Drive - Single DC topology. Please ask your Arista SE to create an ATD lab for you.
  - Build your own lab environment: Ubuntu LTS + Docker + ContainerLab. This option is not described in detail, but the devcontainer used to build Codespaces environment will work on any machine with Docker installed. Please contact your SE if you need help.

Lab Topology

- This workshop is using Arista Test Drive Single DC topology.
- To match minimize resources and fit default Codespaces 4-core machine, the topology was reduced by removing leaf3, leaf4, host1 and host2.
- Feel free to adjust Ansible inventory and group variables if you are using ATD lab and would prefer to activate them all. But it's not essential for this workshop.
- CVP is not used as it's not required for this workshop.



How to Setup ATD Environment

- skip practice this section if you are using Codespaces
- still read the slides as they explain AVD installation process

References

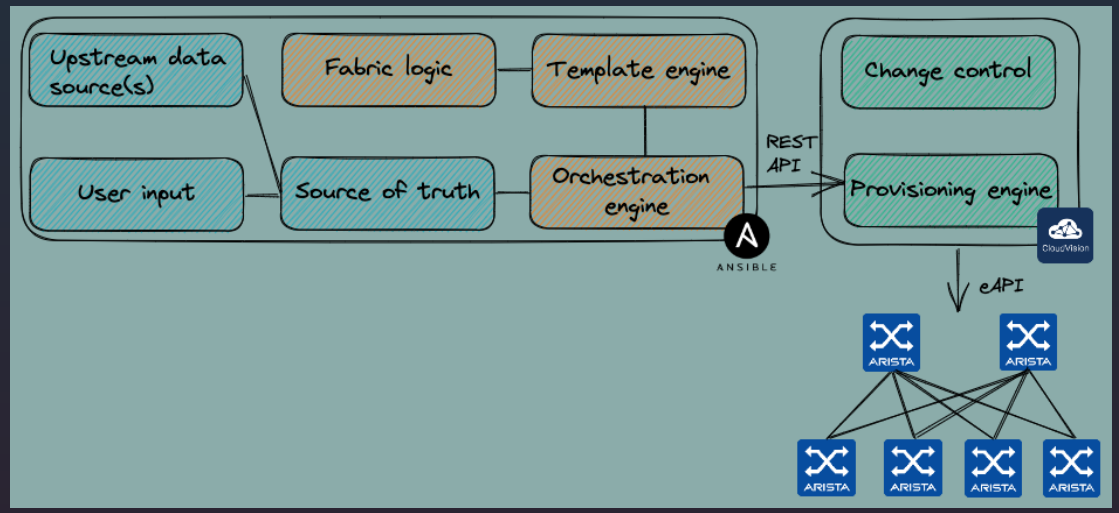
- If you are not using ATD, the functionality of this repository will rely on many amazing open source projects:
  - ContainerLab
  - VSCode
  - DevContainers
  - Marp
  - Excalidraw VSCode
- This repository is also relying on following free/commercial Github features:
  - Github Actions
  - Github Pages
  - Github Codespaces
- All photos are taken from Pexels and Unsplash. Excellent free stock photos resources. It's not possible to reference every author individually, but their work is highly appreciated.

How to setup Ansible AVD in Arista Test Drive environment?

- We could use a script to setup required Ansible collections and tools in Arista Test Drive environment, but it's a good opportunity to discuss what are the requirements but installing them manually.
- For details please check AVD documentation `Installation > Collection Installation` section.

Typical Ansible AVD Automation Workflow

- Collect user input from various data sources and aggregate in a single source of truth. For ex. git repository.
- Generate low level variables from abstracted input data using sophisticated fabric logic
- Parse Jinja2 templates to produce plain text configs
- Push plain text configs via CloudVision Portal as change-control "proxy" or directly to devices via eAPI.



What is Git?

- **In Short**:

  > Git is a distributed version control system that tracks changes to a set of files and enables collaborative work.

- **Fun Fact**:

  > Git was created by Linus Torvalds in 2005 to develop Linux kernel.

What is GitHub?

- GitHub is a Git repository hosting platform.
- Allows to coordinate multiple local copies of the same repository and more.