1 | API-First Approach
Everything is driven through APIs. Whether it’s using NSX UI or using automation tools, APIs and SDKs provide the interface into all product functions and control.

2 | Network Infrastructure as Code
- Create complex network topologies and address complex security requirements programmatically
- Manage the source code in version control systems
- Integrate the entire process with a CI/CD pipeline, enabling network and security infrastructure to be managed as code

3 | Many Automation Tools in the Chest
- Supports VMware tools such as vRealize Automation, PowerCLI, and third-party tools such as Terraform and Ansible
- Puts flexibility into the hands of the network engineer

4 | Hierarchical APIs = API Simplicity
Designed to simplify with hierarchical structure allowing for create/edit/delete interactions with multiple objects with a single API call.

5 | OpenAPI standard-based API
- Enables and standardizes schema development and export to various programming languages
- Supports flexible interfaces and APIs

6 | Active Community | Sample code and fully automated end-to-end use-cases are available in GitHub for easy consumption: https://github.com/vmware-samples/nsx-t

7 | Wide Variety of Users
- VADs can leverage PowerCLI to interact with NSX
- DevOps/DevSecOps can use tools like vRealize Automation and Terraform to automate NSX infrastructure

8 | Wide Range of Authentication Mechanisms
- Choose from basic auth to certificate-based auth and LDAP/AD integration
- Network engineers gain flexibility and efficiency when managing their network automation

9 | Operational Efficiency
- Designed to simplify with Hierarchical structure allowing for create/edit/delete interactions with multiple objects with a single API call

10 | Risk Mitigation
- Reduce human errors and ensure consistency which translates to lower risk of equipment failure, data breaches, and regulatory compliance violations

To learn more, visit: https://developer.vmware.com/sdks