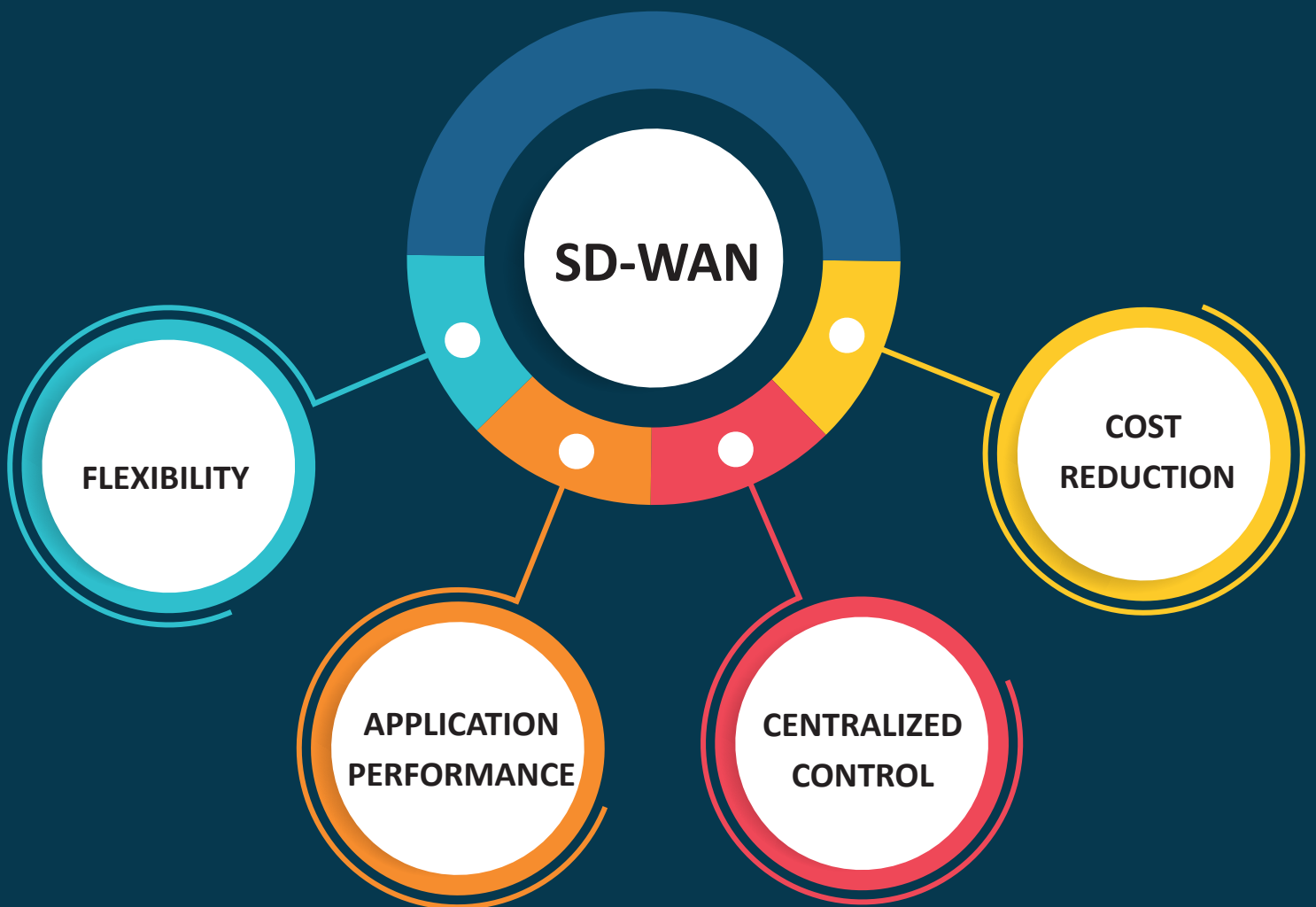


Automation of SD-WAN Solution



CLIENT OVERVIEW

Our client, an American IT giant, is a pioneer in WAN optimization. The company provides integrated SD-WAN, WAN optimization, performance management and cloud-ready branch solutions to drive the cloud-centric enterprise. It is at the center of hybrid networking, cloud, SD-WAN, SaaS, mobile, big data, infrastructure visibility and focuses on building software-defined architecture for digital business.

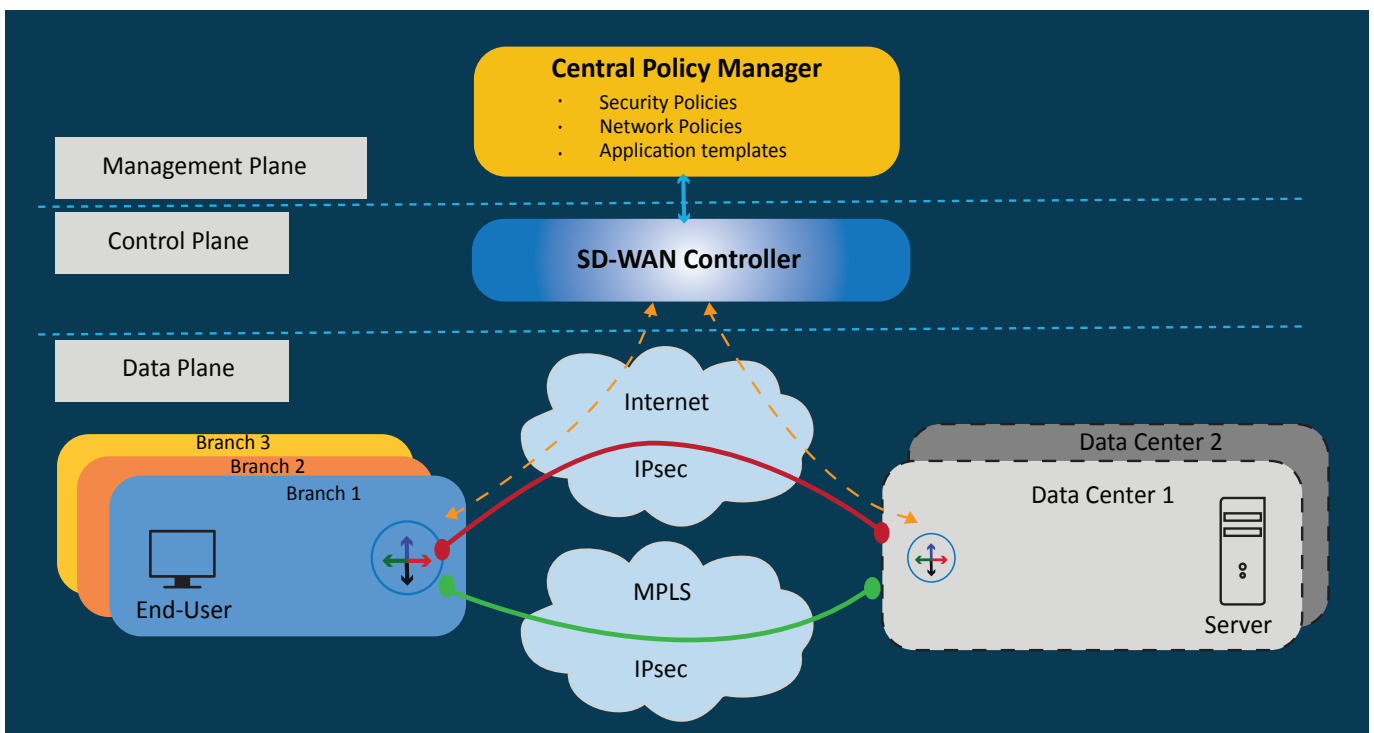
BUSINESS NEED

Our client is moving from traditional WAN optimization to SD-WAN, the emerging technology that can be deployed in the cloud infrastructure. ThinkPalm's technically adept QA team was required to perform the automation of the appliances and applications in the client's SD-WAN solution.

SERVICES OFFERED

ThinkPalm supported in authorizing and automating the customer's SD-WAN solution through the following testing services:

- Manual testing including Feature testing, System testing, Integration testing, Solution testing and UI testing of submodules such as Catfish (Branch device), Gateway (Branch device), Panther (Data center device) and Steelconnect GUI (SD-WAN Controller)
- Test automation of the submodules
- CLI and REST API Automation using Python and Pytest frameworks
- UI Automation using Selenium and Robot frameworks



Module-Specific Testing Services:

Catfish and Gateway

- Automation of different features/scenarios of Catfish and Gateway devices
- Running the automated test scripts on each release using Bamboo/Jenkins continuous integration tool
- Triaging and resolving execution issues

Panther

- Automation test scripting
- Incorporation of code changes in failed automation test scripts
- Execution/triaging failures at the time of releases
- Creation of test scripts based on the priority of tasks

Steelconnect Manager

- While the testing conducted in various submodules was appliance specific, the final validation of Steelconnect Manager was an integration testing of client appliances. The tests performed ensured that all appliances worked together when deployed on the field.
- Complex, customer-centric topologies that had multiple client platforms performing various functionalities on the network, were built for this module.
- The automation scripts written were made scalable and reusable for testing other modules as well.

CHALLENGES FACED

- Openstack, Jenkins execution, Python scripting and PQ framework were new to the team. Internal and external trainings were conducted to familiarize them with these platforms and technologies.
- Bamboo/Jenkins continuous integration tool was used for running testcases on physical/Openstack virtual setups. This required detailed understanding of the CI/CD pipeline.
- Since it is important to understand the behavior of devices before moving into automation, manual testing was performed as the first step to understand the features better.
- Complex, customer-centric topologies that could be applied to most of the customer scenarios were created for integration testing.
- Rigid, scalable and reliable scripts, that are topology independent, were created for a stable release, devoid of release blockers.

BENEFITS

- Helped the customer deliver solutions within time to market
- Supported the auto regression and maintenance of test scripts for new releases
- Reduced the automation backlog release after release by dividing the test case automation based on features, employing Openstack topology for automating testcases, creating testbeds using ESXI setups in the absence of physical testbeds, and reviewing the code meticulously after every release.