WHITE PAPER



Accelerate Reproduction of Customer Network Issues

With automation of TAC labs

KEYSIGHT

Why TAC Labs Need Automation

The problem

When a customer reports a network issue to a Network Equipment Manufacturer (NEM), the Technical Assistance Center (TAC) team first has to reproduce it. That means recreating the customer's network in the lab, finding the right equipment, cabling the right topology, configuring the right devices, and isolating exactly where things break down. Every case starts the same way: reproduce first, then resolve.

Networks keep getting more complex, and most TAC labs still run on manual setup: engineers hunting down equipment, hand-cabling testbeds, and configuring devices one at a time. That work can take days, sometimes weeks, before an engineer even starts diagnosing the actual problem. Meanwhile, customers expect answers in hours, occasionally minutes.

Adding more lab technicians or duplicate equipment would close that gap, but TAC budgets do not have that kind of room. The lab has to do more with what it already owns. The real cost is not the equipment sitting idle — it is the customer case sitting open while nobody can move it forward.

The solution

Lab automation closes the gap by connecting equipment through Layer 1 automation switches and automating the connectivity, configuration, and network changes that used to take an engineer hours of hands-on work. Instead of re-cabling a testbed for every case, the switch fabric reroutes signal paths without the need for physical connectivity changes.

A common portal sits on top of that fabric, giving every team — regardless of time zone — remote, 24/7 access to the lab and the ability to reserve equipment across groups instead of hoarding it. Engineers stop waiting on physical access to a rack and start working the moment a case lands.

> **Lab automation is not about automating the engineer's work. It is about automating the lab around the engineer, so the right environment is connected, available, and ready to use.**

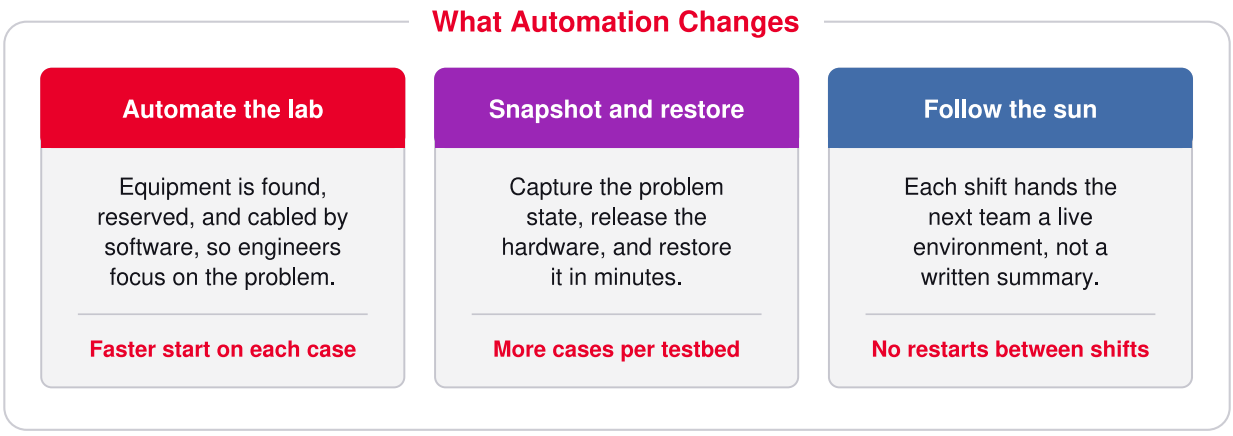
The benefits

With connectivity and configuration automated, a team can stand up, reconfigure, and tear down a full test environment with less manual effort. That speed changes how often engineers can afford to test: they run more validation passes, catch more edge cases, and close cases faster because the lab setup is no longer a bottleneck.

Shared, reservable resources mean teams stop maintaining separate equipment pools "just in case." The same rack of gear now supports financial-services compliance testing, service-provider validation, and TAC troubleshooting — whichever case needs it most that hour. Labs scale to meet demand without a matching increase in capital spend.

How Automation Changes Daily Work

Automation changes where TAC engineers spend their time. Software handles the lab mechanics so engineers can concentrate on the investigation. Snapshot and restore frees hardware between sessions. Follow-the-sun teams inherit a working environment instead of a written summary.



Automate the lab, not the engineer's judgment

TAC engineers need to investigate, experiment, and decide what to try next — no automation should take that away. What it should remove is the friction around that judgment: finding equipment by capability and availability, reserving it without informal Slack-and-spreadsheet coordination, and automating the switching and cabling that assembles a testbed. From there, engineers work interactively — changing configurations, running helper scripts, applying a patch, trying a different topology — because customer issues are unique and often incompletely documented.

When they hit a known problem state, they capture it — a snapshot that preserves the environment as both evidence and a reusable starting point. When the investigation pauses, the hardware is released back to the shared pool instead of sitting reserved. When work resumes, the environment is restored in minutes, carrying the context needed to understand how the issue reproduced, and the whole reproducible setup is handed off to the next engineer or team — not just a written description of it.

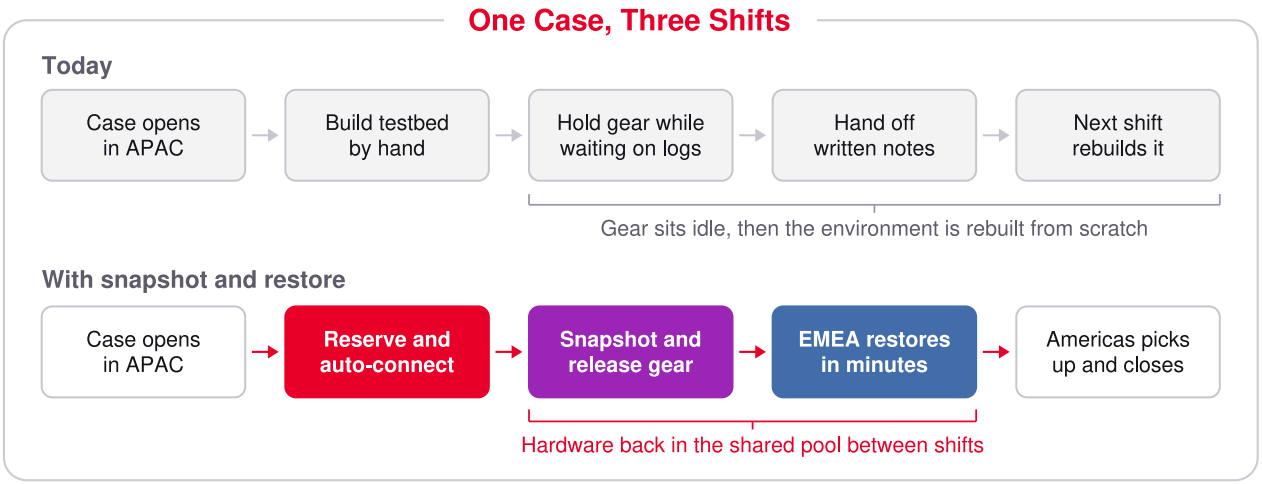
Snapshot and restore changes the economics of shared equipment

The most valuable pattern in TAC automation is the ability to capture a complex environment mid-investigation, release the underlying hardware, and restore it later exactly as it was. That separates the state of the customer's problem from continuous ownership of physical gear.

Without it, engineers hang onto expensive testbeds because rebuilding them later is too costly — understandable, but it creates a hidden capacity tax: equipment sits unavailable even while the engineer is just waiting on customer logs or engineering guidance. With snapshot and restore, the team gets that capacity back. More cases share the same infrastructure, and engineers spend less time reconstructing environments they've already built once.

Built for global, follow-the-sun operations

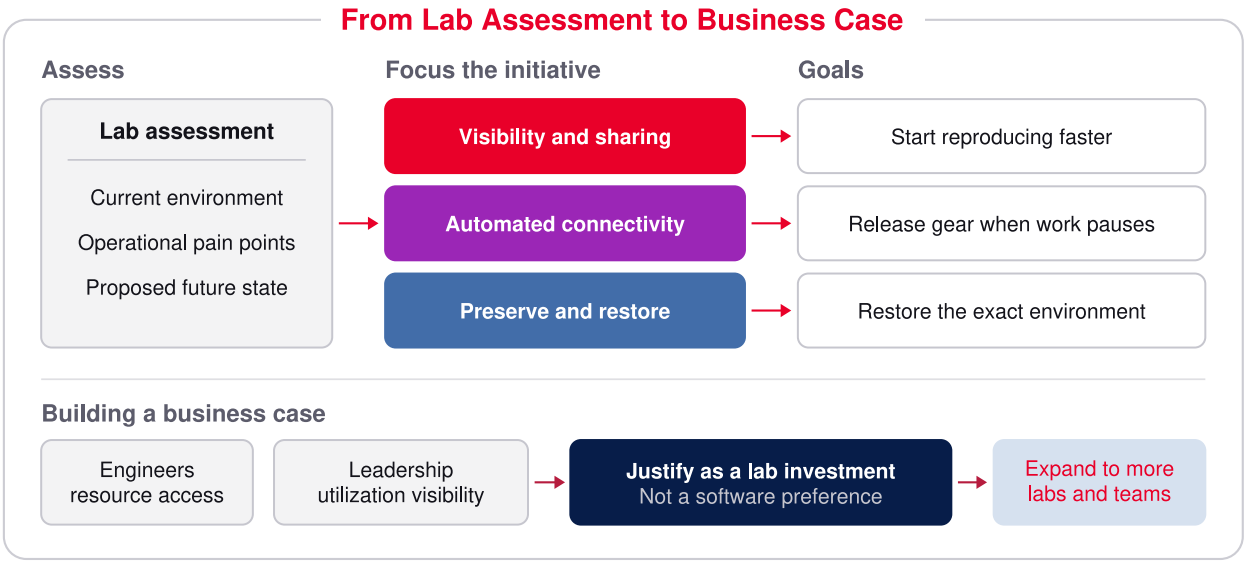
The engineer who opens a case usually is not the one who closes it. Teams span time zones and often share a larger lab across multiple technology groups. Automation makes the handoff clean — an engineer preserves the case state at the end of a shift, releases resources if appropriate, and passes the environment to the next authorized team. That team restores the same environment and keeps going instead of starting over.



How a Global NEM Built the Case for TAC Automation

The customer's TAC organization needed to find, reserve, and connect the right equipment fast enough to reproduce customer issues — some of which spanned testbeds owned by different technology teams. Asset visibility was limited, and manual processes made it hard to use expensive lab gear efficiently.

The work started with a lab assessment that documented the current environment, the operational pain points, and a proposed future state — connecting engineering friction directly to resource utilization and resolution speed, which is what built the business case.



The resulting initiative focused on three things: better resource visibility and sharing, automated connectivity, and the ability to preserve and restore test environments. The goal was simple — help engineers start reproducing faster, release shared equipment when active work paused, and bring back the exact environment when a case resumed or a different team picked it up.

The customer is now extending the model across additional labs and technology teams so resources can be shared more broadly instead of every group maintaining its own isolated equipment pool.

What made the case compelling was not just the engineering benefit. The problem was visible at every level — engineers wanted faster access; leadership wanted better visibility into utilization — and the purchase was justified as a lab investment, not a software preference.

Build a Case Around Measurable Outcomes

TAC automation is easier to fund once the conversation moves from features to outcomes you can measure. The right metrics depend on the organization, but a practical baseline covers time, utilization, and case throughput.

Measure	Baseline it	Why it matters
Time to reproduce	Hours from intake to a usable environment	Shows whether lab friction is delaying the case
Setup / teardown time	Hands-on hours locating and rebuilding environments	Quantifies avoidable engineering effort
Equipment utilization	Active use vs. reserved/idle time	Shows whether the lab investment is fully used
Cases per environment	Customer cases supported by shared resources	Demonstrates capacity gained from sharing
Time to handoff	Time for another team to continue a case	Shows the value of state preservation
Time to resolution	Case duration from reproduction to fix	Connects lab improvements to customer outcomes

Build the case around time saved, higher utilization, and increased case throughput — but measure the baseline first. Then track the improvement after automation is in place.

A Practical Path to TAC Lab Automation

1. **Assess the current lab.** Map equipment, connections, owners, workflows, and where engineers lose time today.
2. **Prioritize the highest-value workflow.** Pick a recurring TAC workflow where faster reproduction or better sharing will matter most.
3. **Establish the resource model.** Build one consistent view of equipment, availability, reservations, connectivity, and access.
4. **Automate the lab mechanics.** Automate switching, connectivity, setup, teardown, and environment capture/restore.
5. **Pilot with real cases.** Use representative TAC issues, not a synthetic demo, and measure time and utilization as you go.
6. **Expand by team and location.** Extend the shared-resource model to more technology groups, labs, and follow-the-sun teams.
7. **Manage by metrics.** Track utilization, reproduction time, handoff time, and resolution time as ongoing KPIs.

What Good Looks Like

A mature TAC lab isn't just a pile of equipment with software layered on top. It's a shared operational system that moves engineers from customer case to reproducible environment with almost no friction. Resources are visible, reservations are deliberate, connectivity is repeatable, environments can be captured and restored on demand, and any authorized team can pick up a case without rebuilding the lab first. Engineers spend their time on the customer's problem, not on managing the infrastructure around it.

The payoff is faster resolution and better use of the equipment you already own. Start with the workflow, not the tool. Find where engineers wait, where resources remain tied up, and where environments get rebuilt repeatedly. Automate those points, measure the result, and expand from one priority lab to the wider organization.

About Keysight Velocity

Velocity Core: The platform behind the automation

Keysight Velocity Core is a lab automation platform that turns distributed lab equipment into a web-accessible shared resource. It gives every team a self-service way to reserve, connect, and orchestrate lab resources through a drag-and-drop topology editor, integrated directly into CI/CD.

Just-in-time reservation lets engineers claim resources and release them the moment they no longer need them, even mid-test. Deep visibility into inventory lets leadership forecast usage, and workflows can be built and shared across teams through Git, Jenkins, or Jira.

New: Velocity 4900 series automation switches

Velocity Core's connectivity layer now supports the 4900 series automation switches, with true Layer 1 protocol-agnostic forwarding for functional and performance testing.

Model	Configuration
Velocity 4906	6RU 288 SFP28 ports 100 Mb/s; 1, 2.5, 5, 10, 25 Gb/s
Velocity 4902	2RU 72 QSFP28 ports; up to 288 with breakouts 100 Mb/s; 1, 2.5, 5, 10, 25, 40, 100 Gb/s

The 4902 fits teams that want to pool lab equipment, create testbeds programmatically, and reduce the time required to set up and test. The 4906 switch offers higher port density and scalability to thousands of ports, making it well suited for large test labs that need to scale connectivity while preserving test fidelity.

As a mission-critical design enablement partner, Keysight connects design, emulation, and test so customers can solve the world's most complex engineering challenges. Accelerate innovation at www.keysight.com.

KEYSIGHT

This information is subject to change without notice. © Keysight Technologies, 2026, Published in USA, October 29, 2026, 7126-1494.EN