



# StackState Observability Integrations: Splunk

As a Big Data solution, Splunk is widely installed for its ability to centralize massive amounts of data, coming from different tools. Data centralization enables more efficient data manipulation and analysis. However, since this data is often not correlated, IT teams must still try to parse it to understand what is going on within their IT environment.

StackState integrates with Splunk and [many other monitoring solutions](#) to correlate data from them and roll it into one, integrated topology of your entire IT environment. You get a visualization of exactly what is going on at any point in time, including all components and their relationships and interdependencies with each other.

## Management Summary

- Splunk is a powerful platform that helps IT teams centralize data from different solutions into one data lake. Teams can then search through and query their data more efficiently. Visualizing relationships and dependencies between components is possible with Splunk tools, but it's a lot of manual work.
- StackState's integration with Splunk automatically adds a contextualized topology layer to Splunk data. This gives users a clear overview of their entire environment, including all relationships between various services and changes to them over time.
- The topology can be easily defined from topological data in Splunk and will then be synchronized and kept up to date automatically. Topological data can also be extracted from other sources, such as AWS or Kubernetes.

- After visualizing the topology, StackState's [4T® Data Model](#) can be populated with metrics, events, logs and health status data from Splunk. The integration pushes and pulls from and to Splunk very efficiently – resulting in low storage and compute costs.
- When you combine Splunk data with StackState's ability to establish relationships between the full stack and underlying infrastructure, you get a more comprehensive observability solution and deeper insights into the state of your stack.



## Combine Splunk's Big Data Solution With StackState's Topology-Powered Observability

Splunk's data lake is a powerful platform that consolidates massive amounts of monitoring data from different solutions such as Solarwinds, Azure, Prometheus and Amazon CloudWatch. Aside from bringing data together in one place, Splunk also allows IT teams to sift through and query that data more efficiently.

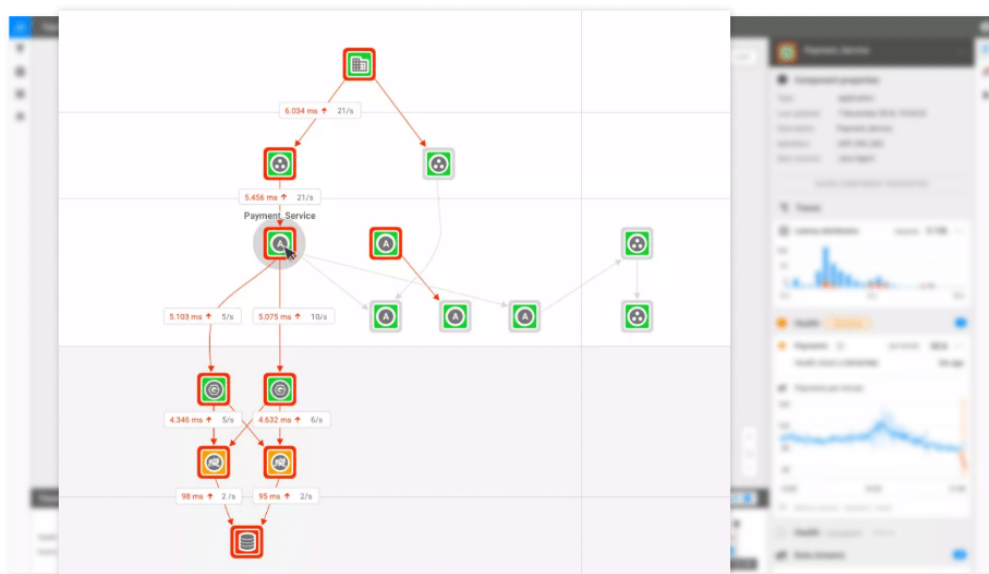
For effective root cause analysis and remediation, however, IT teams need to understand the relationships and dependencies between components in their environment. It is possible to visualize these dependencies through dashboards in Splunk tools, but creating the visualizations and keeping them up to date requires a lot of manual query work – challenging to do when you're working with ever-changing monitoring data.

This is where StackState and Splunk form a powerful combination. By integrating Splunk with [StackState's 4T® data model](#) (topology, telemetry, tracing and time), StackState automatically shows you the relationships between components in your IT stack at any point in time. This gives teams deeper, more actionable insights into Splunk data and allows for quicker, more efficient root cause analysis and remediation.

## Gaining Deeper Insights

Here is how you get deeper insights into your overall IT environment with the combination of StackState and Splunk:

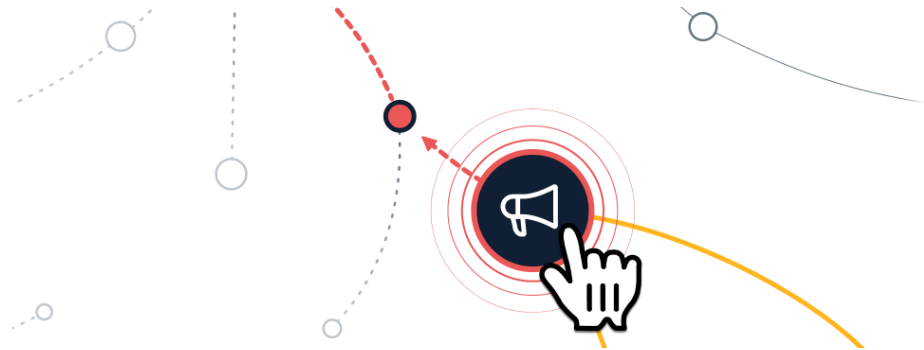
1. **Automatically visualize your entire IT environment in StackState with topological data resting in Splunk.** Automatically creating a holistic topology is made easy: Topological data resting in Splunk can be queried by Splunk's *saved searches*. Once the query has been written, the topology is automatically synchronized and kept up to date in StackState. You can also instruct StackState to use topological data from other sources, such as AWS or Kubernetes.



*StackState uses topological data from Splunk to automatically map out your topology and to show you how the components are related to each other.*

2. **Correlate monitoring data over time.** StackState uses metrics, logs, events and health status data from Splunk and correlates it over time. That way, you can look at what was occurring at any point in time - for example, a change made to a component just before a failure occurred.

3. **Prevent issues before they occur.** StackState's AI-powered Autonomous Anomaly Detector analyzes Splunk data and automatically detects anomalies before they become issues.



## How Data Is Shared Between StackState and Splunk

The Splunk integration *pulls* and *pushes* Splunk data very efficiently from and to StackState, resulting in lower storage and computing costs.

- *Push*: Topological data resting in Splunk is regularly pushed to StackState via the StackState agent.
- *Pull*: Telemetry data is pulled on-demand from Splunk. For example, when a user is exploring their IT environment within StackState, the Splunk-related data remains in Splunk, but the StackState user gets an up-to-the-minute view of what's going on in the environment.

## The Result: Visibility Into Changes and Dependencies Throughout the Full Stack

Because StackState correlates the consolidated data in Splunk, you will see every change made in your IT environment and when it happened. Should an issue occur, you will see what change caused it and when, as well as where in your environment the change was made. This includes both business processes and changes in the underlying technology stack. The integration also makes it possible to identify these changes regardless of their data source, giving you a more comprehensive view of your overall system.

## Conclusion

When Splunk is integrated with StackState's topology-powered observability, you automatically get a deep view into both your telemetry and topology. IT stack-level metrics—component monitoring, alerts, events and logs—are visualized in real-time by StackState, including the dependencies between all components.

You don't have to process or construct visualizations manually or try to analyze data manually to try to figure out what occurred; StackState does this automatically for you. This makes it easier to see exactly what went wrong, where and how the relationships between different components resulted in issues. You can also use StackState's AIOps capabilities to proactively prevent future issues.

In short, the seamless integration between Splunk and StackState helps DevOps and SRE teams to make the enormous amounts of Splunk data more actionable. The results? Faster, more effective root cause analysis, lower Mean Time To Repair and more healthy and stable IT environments.



## Use Case: StackState and Splunk Within a Financial Services Organization

Here is an example of actual StackState financial services customer, using both StackState and Splunk.

- A bank detects a problem with the conversion rate and the availability of a payment service. This is immediately displayed in Splunk.
- The root cause of the problem is a change, such as a load balancer configuration change. This configuration change is recorded automatically by StackState.
- StackState can use the KPIs displayed in Splunk (i.e., conversion rate and availability) as telemetry and plot these on high-level business components

within the topology in StackState. Using data correlation, the components that are failing are shown as red in both Splunk and StackState.

- Component monitoring reports a problem on one of the underlying IT infrastructure components, component X. This is displayed in StackState.
- Because StackState understands the relationships and dependencies between high- and low-level components, it automatically presents the load balancer configuration change as the probable root cause of the problems.
- For further investigation, you can use StackState's time travel functionality to go back in time to the moment component X changed status. You can view the event stream in StackState to establish that the change happened just before component X went red.



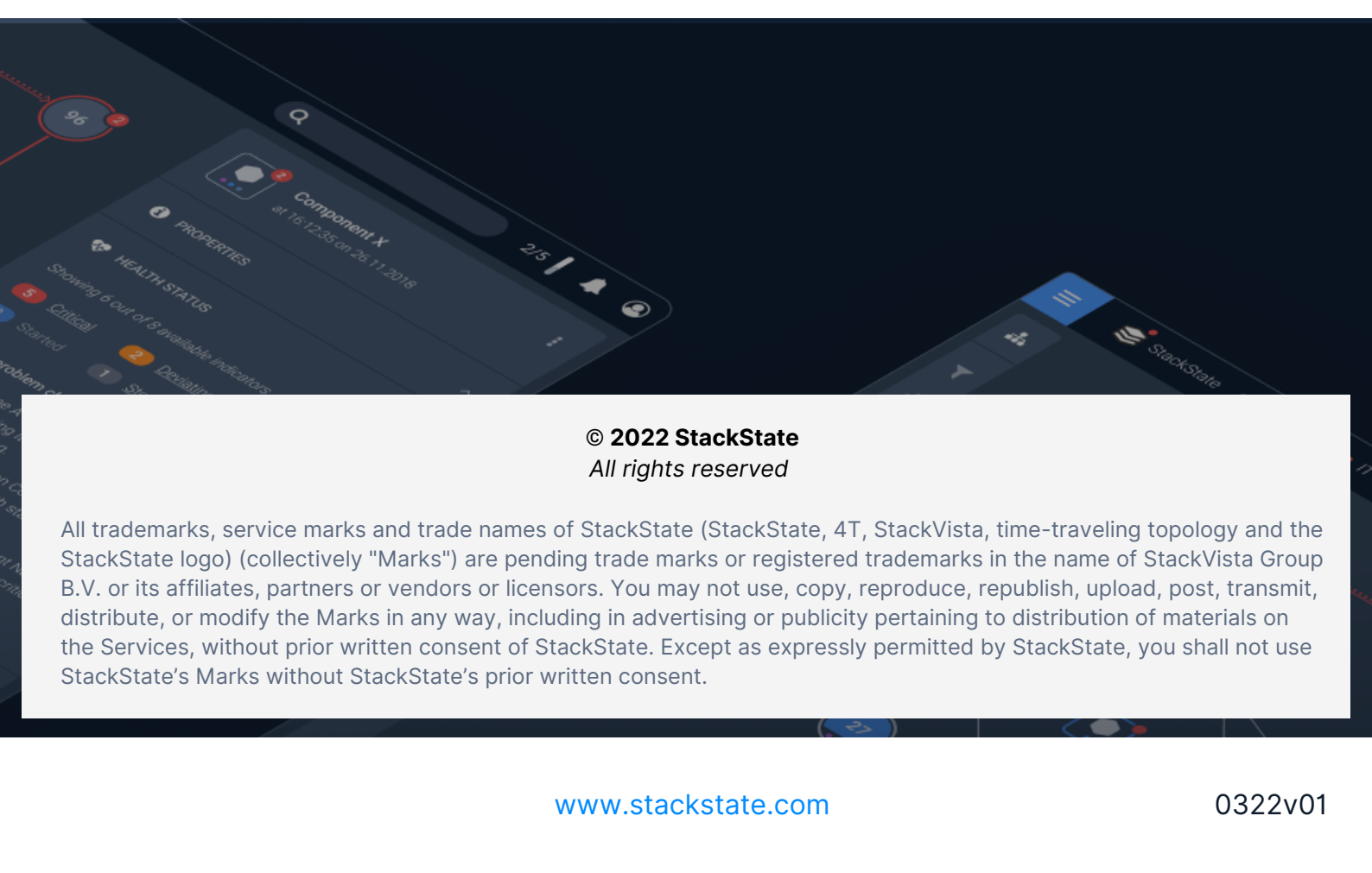
*See exactly what your topology looked like before a component changed status.*



## About StackState

StackState is the only observability company with a platform that combines topology with existing monitoring data over time. Our topology-powered approach provides the most complete picture of the state of your stack and the intelligence you need to quickly find, fix and prevent problems. StackState improves the performance and reliability of your critical business services in complex hybrid, cloud and container environments.

For more information, visit [www.stackstate.com](https://www.stackstate.com) or [book a demo](#).



© 2022 StackState  
All rights reserved

All trademarks, service marks and trade names of StackState (StackState, 4T, StackVista, time-traveling topology and the StackState logo) (collectively "Marks") are pending trade marks or registered trademarks in the name of StackVista Group B.V. or its affiliates, partners or vendors or licensors. You may not use, copy, reproduce, republish, upload, post, transmit, distribute, or modify the Marks in any way, including in advertising or publicity pertaining to distribution of materials on the Services, without prior written consent of StackState. Except as expressly permitted by StackState, you shall not use StackState's Marks without StackState's prior written consent.