



# The IT Guide to Synthetic Monitoring

Good end-user experience is part of many IT teams' mandate right now. It's no longer enough to keep networks and applications up and running—now IT has to make sure users aren't dealing with continually slow or badly performing apps. That's true even (or especially) for apps that IT doesn't control, like SaaS or cloud-hosted apps such as Office 365 or Salesforce.

Synthetic monitoring is an important part of an overall monitoring strategy or tool, as it's active, rather than passive, monitoring. That's essential in the complex infrastructure that many IT teams are managing, where multiple networks, providers and applications can all consume IT time and resources. Synthetic monitoring is a big help for troubleshooting in these cases, since it shows IT teams what users are seeing in a continuous way.

Synthetic monitoring is a modern way to see the performance of today's SaaS and web applications and networks. It uses scripting to emulate the paths that end users take as they experience an application. Synthetic monitoring can take various forms, though, and varies widely in sophistication and complexity. Consider some best practices for deploying these tools.



## Why We Need Best Practices for Synthetics

Synthetic monitoring provides IT staff with performance context for business-critical SaaS and web apps. Except for company-wide deployments, IT staff rarely has cause to interact with SaaS apps until something goes wrong. Synthetic monitoring helps you in IT to understand the baseline performance over time and match the deviation reported by users.

Many synthetic providers offer basic availability checks as the first line of alerting for these apps, but in our modern

world, apps rarely go down. Rather, app performance will often slow due to myriad factors including local networks, wide area networks and general congestion. While some SaaS providers will underprovision apps, this occurs less and less often in the auto-scaling world of AWS, Azure and Google Cloud. Availability checks also fail because IT teams target the homepage of a company instead of the login page of the app. Additionally, the login page is a poor substitute for a script that actually logs into the app to be monitored and loads data like an actual user would.

## Best Practices to Make Synthetic Monitoring Work for You

There are a few ways to find the right synthetics setup for your business. IT teams should consider these details when configuring a synthetic monitoring solution.

**Do a Ticket Analysis.** Take a look at the past month or quarter of tickets and analyze which apps were the most troublesome. Whether apps have unreproducible issues, common performance problems or are high-profile, you'll want to have a list ready for synthetic monitoring.

**Find the Expensive Apps.** When you are paying top dollar for an app, a bad user experience can spell disaster for productivity or confidence in the app. Identifying these apps up front and proactively monitoring to them can help mitigate these issues.

**Consider the App's Footprint.** Performance issues with apps like G Suite or Office 365 likely affect your entire employee base. You have to monitor them appropriately. Availability checks on the Office 365 login screen aren't going to cut it. Each of these massive apps is actually an ecosystem of apps that perform independently and are most often hosted on different clusters of machines within the same location.

**Set Trackable Goals.** Setting up a synthetic monitoring solution can seem like a set-it-and-forget-it endeavor. While that's mostly true, you'll need to do some legwork upfront to determine what you need out of synthetic monitoring. Often synthetic scripts are used to show the normal performance of an app over time, so that IT teams can alert on deviations and take action instead of waiting for user complaints. Taking action, however, requires more information. Network context is extremely important for monitoring SaaS and web apps, because while apps traverse between source and user, they travel the LAN and WAN with routing that changes dynamically. Correlating the route, app and end user at the time of incident is crucial to understanding the root cause of an issue.

The goal of synthetic monitoring should be to closely simulate user interactions, so that alert thresholds are triggered when users could conceivably be impacted. This means ignoring the marketing-generated homepage and gateway login screens of applications. Creating more complex scripts is the secret to getting accurate and useful results out of synthetic monitoring. The end result should be the IT Ops team's ability to solve issues faster. Tracking ticket resolution times for common complaints is a great way to assess your synthetic monitoring success.

All these potential features are important, but you'll have to decide what your company and users need most. Take your particular networks and workloads into account when shopping around.



## What to Look for in a Synthetic Monitoring Provider

Synthetic monitoring helps IT to understand baseline performance over time, then see deviations from it when users complain. Synthetic monitoring can come as a standalone product or as part of a broader monitoring solution (like AppNeta's) but there are some essential features to look for in a provider or solution.

### Must-Have Features for Synthetic Monitoring

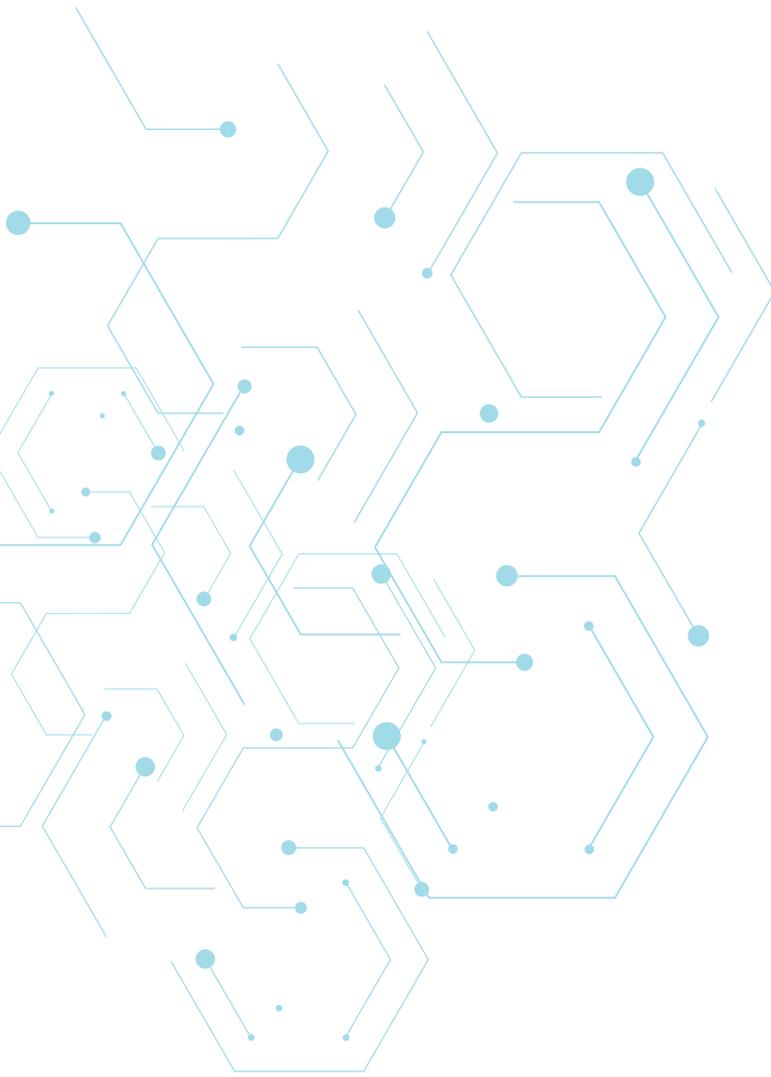
Every synthetic monitoring provider should offer these features to enable complex SaaS and web app monitoring.

**Milestones for Complex Scripts.** The evolution of SaaS and web apps means that instead of multiple pages of data in one app, many are now just a single page that dynamically loads data as the user requests it. This largely mimics a desktop experience without the burden of a thick client. The issue is that synthetic scripts are dependent on page loads to display the timing information. Milestones are a solution AppNeta has come up with that hides in the comments of Selenium (for portability purposes) and allows users to determine when actions are complete. This allows timing information to be shown for any sequence of actions without page loads, which increases the value of synthetics in the single-page app world.

**Assert/Verify Functions.** Users will note when something on a page doesn't load correctly. Synthetic scripting historically hasn't tracked this consistently, nor have users taken advantage of this to the extent that they should. Using the [Assert function within Selenium](#), synthetic scripts will cancel operation if an element is not present. For critical components of a dashboard, this can lead to early detection of app failures. The Verify feature similarly detects whether or not an element has loaded, but instead of canceling the rest of the script it will log the error and continue. This can be helpful to note failures that may be less critical, but are important for troubleshooting purposes.

**Screenshots on Error.** Any deployment of synthetics should have the ability to capture a screenshot when an error occurs. This is essential when troubleshooting, as it allows IT Ops teams to see exactly what the user sees during the failure. This can not only help contextualize an error to the visual representation of that error, but aids in matching user complaints with synthetic testing results.

**Points of Presence.** While originally reserved for SaaS providers monitoring from different global locations, this has become more important in our increasingly distributed corporate world. Not every employee is under one roof, so the significance of monitoring from an actual user to their location (or one nearby) can not only provide more accurate data but allow IT to troubleshoot remote issues faster.



**Global presence points.** Most synthetic providers today claim a global presence, but that model serves SaaS providers better than their consumers. Unless you are a multinational corporation with 100 offices, you are rarely using a global map to monitor performance. A few global points of presence, however, can give you an idea of how users in locations near yours are experiencing your SaaS or web apps.

**Custom points of presence.** If you are a multinational corporation, or a company with offices in remote locations where you care about [ensuring a good end-user experience](#) (read: the CEO's home office), then you'll want to set up custom devices or virtual agents to monitor performance from specific end-user locations back to your SaaS apps or data center. The benefit here is that when that CEO calls to complain, you're not relying on a monitor that is at best 30 miles away and likely on a different carrier for the last mile. You'll have actual data from their location over the same ISPs and you'll see it trended over time so you can pinpoint when issues began.

Keep all these features in mind when you're evaluating synthetic monitoring providers. In our widely distributed world of cloud and SaaS applications, synthetics can save you a ton of time on troubleshooting.



## ABOUT APPNETA

AppNeta is the only network performance monitoring solution that delivers deep, actionable, end-to-end network performance data from the end-user perspective. With AppNeta's SaaS-based solution, IT and Network Ops teams at large, distributed enterprises can quickly pinpoint issues that affect network and business-critical cloud application performance, regardless of where they occur. AppNeta is trusted by some of the biggest Fortune 1000 companies, including 3 out of the 5 largest corporations in the world, as well as 4 out of the 5 largest cloud providers. For more information, [visit www.appneta.com](http://www.appneta.com).

1.800.508.5233 | [SALES@APPNETA.COM](mailto:SALES@APPNETA.COM) | [APPNETA.COM](http://APPNETA.COM)