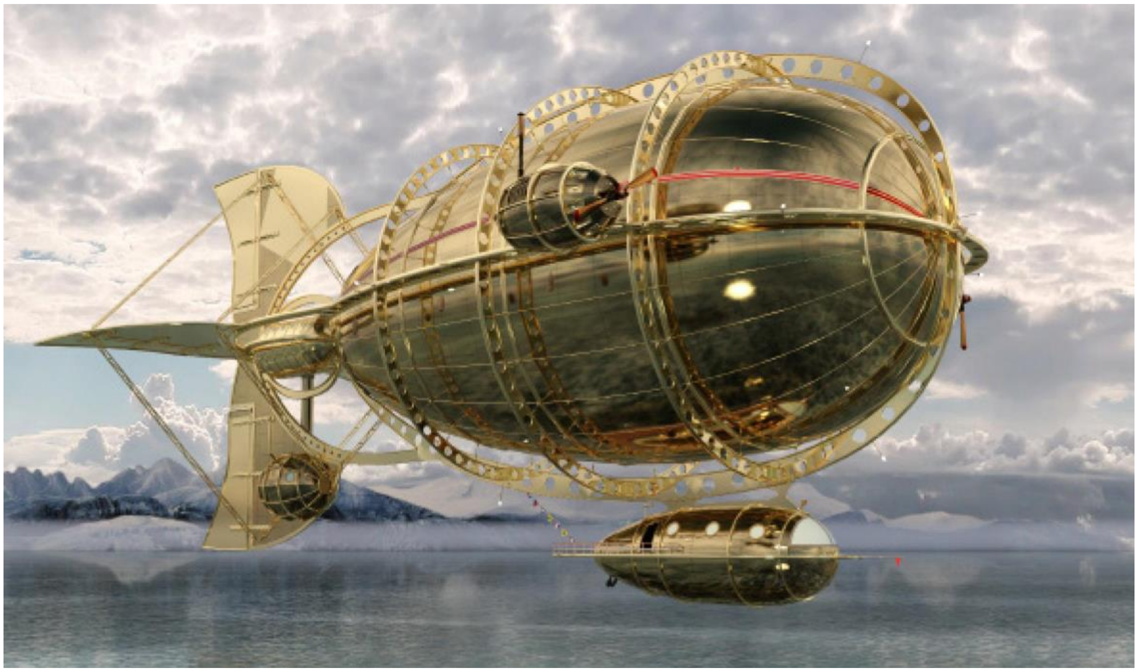

Don’t Add Another Tool To Your Marketing Tech Stack Before You Can Answer These Questions

Abstract

As companies settle into the New Year, marketing teams will deploy up to 20 new technologies in 2022. However, most organizations aren't set up to effectively manage their existing technology investments. Before your team deploys the next tool - adding complexity and likely confusion and waste - make sure your technology team can execute at a baseline of quality.   The white paper will explore seven points on the topic like Build vs Run state operations, ROI, SaaS Security, Regulations and Quality Management. The white paper also provides a checklist of questions any leader can use to evaluate their team's readiness for adding complexity to a technology stack.

Run State concepts include understanding the ongoing headcount required to administer the new tool set, including data management, user management, integration and configurations. Very frequently, going through a run state exercise unveils that marketing and technology teams are already over capacity in managing their existing technology stack. Without adding efficiency or headcount, the only way to add new tools is to willingly sacrifice quality or consistency.

Quality management concepts include understanding things like service level agreements with stakeholders, the launch plan and functional, regression testing, and rendering testing.

Total stack view should help you think about this tool in terms of which technology vendor supports the integrations and what SLA's they provide, monitoring of the end-to-end tech stack, and the ability to run a regression test - when they add a new technology, how will they know mission critical operations in other tools won't be affected?

ROI for new tools is very simple to calculate - most sales and marketing systems can demonstrate the return based on simply winning one or two new deals a year. The Opportunity Cost of that investment is an incredibly different discussion. Is the cost of devoting resources to this new system that all of our attribution fails for three weeks? Is the cost that the Marketing Operations team is so under-resourced as we roll out the project that we miss 15% of leads from our Request a Demo campaign and never provide follow-up?

Table of Contents

Build vs Run State Operations

While building a certain technology into your stack takes a set amount of effort, this process is short-lived and may be viewed by your organization as a cost worth the value of a new platform. This part of the adoption is important to consider, but it is often the only time allowance given to the introduction of a new tool. This fallacy leaves the recurring time cost that goes into maintaining any platform unaccounted for, which means it either is not properly monitored or the people responsible for it are not prepared for the impact it will have on their overall work capacity.

Run State operations refer to the entire lifespan of a technology; the upkeep, maintenance, and monitoring of it requires team contributions that are permanently added to their plate, and this is often not sustainable for the long term. Consider the current workload of your employees—ideally, an evaluation of this will find that each member has a manageable workload that is neither too light nor too heavy, a sign of a well-constructed team. By adding a new tool, there are now two scenarios: either consistency and quality take a hit as the group takes on an extra project they don't have time for, or you make new additions to the team to compensate for this added platform. If you're unable to invest in bringing new professionals into your organization to manage the new tool, but are certain that it is needed to impact your team's productivity, then it's time to take a look at the current priorities existing team members have. Reevaluating what each person is responsible for may identify an opportunity to realign someone's day-to-day role to fit in the Run State operations for the platform. However, it's crucial that this decision is taken seriously and current priorities are realistically weighted, not simply brushed off to make room for this new tool.

If you are prepared to hire team members dedicated to its Run State operations, then it's important to have a way to appropriately determine exactly how many people are needed, and what qualifications they should have. Speak with your current employees to get an idea of how their time is spent and what attributes they feel are important to the work they do, and then assess the time requirements that go into maintaining this new tool. From there, the determination of how many new professionals you need to hire becomes mathematical.

In order to ensure you're fully prepared for the long term requirements of adding a new platform, take the time to weigh the benefits and costs of this new piece of your stack to determine if it's a must-have tool. Avoid forcing something new into your stack for the sake of having it, especially if it will impact current technologies or lead to dysfunction in your team. Prioritize unlocking the full potential of the tools you already have to make the most of them and minimize wasted resources for something that doesn't add anything new to your lineup.

ROI

Determining the ROI of a new tool is standard practice for your sales and marketing system, but the impact of your investment in this new platform doesn't end with how many deals it will assist you in securing. Other factors that are influenced by the cost are less straightforward to assess, but are crucial to get an accurate picture of how it will affect your operations. Depending on how much room your marketing team has to take on a new responsibility, you may need to be willing to perform less well in other areas of your process. There might be circumstances that justify this sacrifice, but many organizations are caught by surprise when, despite their calculations, the end goal seems further out of reach.

As you evaluate your team's capacity, also take the time to consider how you rank each role that they currently perform. The worst course of action is to lower the value of a more impactful task by placing it under this new tool in your hierarchy, making it less effective. If your organization decides to go the route of sacrificing performance in other areas in order to accommodate this new platform in your stack, don't realize too late that your original order of priorities was actually helping you more with reaching your goals than the new arrangement.

When evaluating the ROI that this tool will bring, there's likely going to be subtractions in the ROIs of other technologies that should also be considered in your overall performance expectations. This assessment also allows you to think about how well the tools already integrated into your stack are performing against their own expected ROIs. If you brought a certain platform on earlier with expectations that it could help in multiple areas of your organizational process, ask questions now to determine if that is holding true. It's possible that you'll find more impactful ways to improve your performance simply working with the platforms already at your disposal, and that adding a new tool into the mix will hinder the potential of what you have.

If you go forward with introducing a new technology, work with your team to create a plan that will maximize its ROI as much as possible without detracting from the ROIs of other areas. Be realistic about which current roles are the most important to protect, and ensure that the integration and maintenance of new platforms doesn't interfere with those prioritized areas. Meanwhile, determine which tasks can take a back seat in order to shift focus to this new item, and help your employees transition away from them to move into their revamped responsibilities.

And finally, continue to monitor after a new tool has been implemented. Track that particular tool's ROI, and also keep an eye on the overall success of your organization to be aware of any changes, positive or negative, that may be appearing with its integration.

SaaS Security

For the good of your organization and your customers, SaaS security is always something to take into consideration. Adding a new software tool comes with the inevitable risk of a security issue if the aforementioned tool has access to customer data, so this part of integrating a new technology into your company is particularly important.

With citizen developers becoming a more prominent part of the mainstream software community, there's a tendency for the exact makeup of a tool to be less than clear. If one platform requires the use of a different third party software, it's possible that any security issues stemming from the third party platform could go unreported by and unaccounted for by the larger platform that you're using. Since the problem maker technically wasn't created by the larger platform, they may feel that it isn't their responsibility to monitor and communicate with their users.

While this mentality is unfortunately a common one, it isn't something your organization should strive to emulate. Implementing tools at the expense of privacy is at best unethical and at worst a crisis waiting to happen for your company, so due diligence should be heavily prioritized with the adoption of any new technology. Determine exactly what data the tool will have access to from your customers, and which elements of your system it will have access to. It's likely that you've invested a great deal of time setting expectations for your customers as to what they can expect from you when it comes to security and privacy, and it's important to continue to meet that regardless of what the tool is.

Remember that your privacy protection is only as strong as its weakest link—and the more links you have, the greater the risk that something will go wrong and jeopardize your security. Thoroughly vet all proposed new stack integrations and make no compromises if something isn't performing to your standards when it comes to safety. If something goes wrong, it's your reputation on the line and your customers that will be affected, so act wisely.

Also consider that more does not always equal better, and in cases like this one, can often make things more complicated and difficult to keep secure. Make the most out of each tool you have and take full advantage of its capabilities to avoid having to add more platforms to your stack. By doing so, not only are you minimizing the chance that something will go wrong, but you're also making it easier on the employees that oversee privacy protection.

Regulations like GDPR, Privacy (what's the potential impact?)

While there may be a heavy overlap between SaaS security and consideration for GDPR/Privacy, it's important to evaluate your new vendors for exactly how they adhere to the privacy regulations for the regions in which your organization serves customers.

As most of us know by now, privacy is a concern of your customer – which means it's a concern for your business as well. And, depending on where your customer is located, the regulations your business face can be very different—as can the consequences.

> Did you know?
> There are different privacy laws by country and, in the USA, State-by-State? Whether you service customers in the United States, Europe, Canada and other parts of the globe, it is critical to dig into how these laws may affect your business.

It's important to consider how your vendors are validating that GDPR compliance has been satisfied. Even though an organization is built to help you adhere to privacy standards, you can expect there will be errors.

For example, AvePoint, the largest independent software vendor of SaaS solutions to migrate, manage and protect data in Microsoft 365, discovered their privacy solution was inaccurately marking records in their database as non-marketable – despite their contact records completing the necessary steps to DOUBLE-OPT-IN to be contacted by the company.

> Some Possible Fines for Violating privacy laws:
> CCPA
> $2,500 for each violation or $7,500 for intentional violation after notice and a 30-day opportunity to rectify has been provided
> GDPR
> Severe Violation: The greater of 4% of offending company prior-year annual revenue or €20 million
> Minor Violation: The maximum fine is 2% of prior-year annual turnover or €10 million

Ultimately, be realistic about your business needs. It's reasonable to put the best protections in place for your customers and your business and have a line of sight on continuing to improve over time.

Total Stack View

The logistics of adding a new platform to your tech stack are an important part of the integration process, and handling them well can make the difference between a successful adoption and doing damage control six months from now. Some of the considerations to make might seem obvious, but others may not come to mind at first when you're planning the tool's rollout.

Determining who exactly owns this integration with the other technologies in your stack will ensure that the right person is getting all necessary information and everyone stays on one page. Instead of fragmenting the role more than is needed, decide from the beginning whose jurisdiction the integration falls under so that the correct person can be there from start to finish. If this new platform also integrates with systems outside of your team, determine the asks you'll be making to its peer team. Failing to do so could make it overwhelming from both teams, and may delay progress if the peer team isn't able to meet your team's needs in addition to their own day-to-day operations.

Are you relying on a third party integration? If it is a native integration, did the tool provider build that integration? Do they own the integration and maintain it? Frequently, the cause of errors is the integration, and both sides of the integration will point fingers. If the new tool vendor created the integration, make sure that integration support and related SLA's are in the contract.

In the same area, think about if there's a rollout strategy that still makes sense even if the peer integration were to become decommissioned at some point in the future. And in the event that a peer team makes changes to their system after the integration is made, discuss how that could impact your own system's performance. Determine how this potential issue would be identified, and map out the monthly time commitment it will take to support that monitoring.

Understanding the baseline performance of your technology stack gives you a metric to evaluate any changes after this new integration. Knowing how many leads flow through your system daily and their current latency will allow you to better predict and plan for how those numbers could change after a new platform has been deployed.

Knowing your system's limits is also important, especially when it comes to understanding how your different platforms communicate with each other. Discuss your API volume and limits and how adding a new tool will affect daily and long term peaks after it's launched. Any planned projects that will impact API volumes should also be mentioned so that your team can think about how your new platform will impact this outlook.

Quality Management

Quality management is always a primary concern, and adding a new tool to your tech stack opens up lots of opportunities for something to go wrong. Preemptive planning of how this platform fits into your existing system gives you space to evaluate the procedure you will follow to minimize the impact on older tools.

What is the plan for deployment? In any software deployment, requirements for success and the related test plan are developed first, often called test driven development. For instance, is the data augmentation tool receiving the webhook? Is it replying when the webhook fires? If the system should not be called, is it being called? The most simple test plan combines user scenarios with the data map. For example, a newspaper would have subscribers and non-subscribers, and create two independent test streams based on that assumption.

Incremental to validating the new tool, it is also critical to understand that all the existing systems remain up and working. Do you have a 2 minute delay in your routing systems? If this new tool creates a 1 minute delay, will critical processes like augmentation, privacy and lead routing fire too soon? A comprehensive regression test will show this.

In any regression test, start with the most time sensitive and highest revenue generating activities first, and then get into privacy or back office systems (most can be rectified if impacted for a short period). Once regression test scenarios are written they can be used again and again. If the regression is written in automation, testing can take minutes instead of weeks.

Compatibility: As you consider a roll out, consider how this will be used by your customers and prospects. Far more than most vendors care to admit, technology only works properly on certain mobile devices or certain browsers. Run a quick analysis of your users to validate which platforms they use to engage with your product, marketing and content and make sure that the technology limitations match your customer's behaviors. For example, imagine if a product only worked on Chrome was being presented to a Microsoft reseller, with customers who require their users to default to Edge.

Rendering Testing: Similarly, if a product has an integrated user experience that will be transparent to your customers, for instance a website widget, make sure that rendering testing across your customers preferred devices and browsers is completed post deployment.

As you think about all the requirements for this tool to work in your whole stack view, what is the obligation of the vendor to support this? Consider including uptime and support service level agreements with financial or contract extending obligations if the vendor does not meet their SLAs. More than just the vendor SLAs, also consider internal SLAs. For instance, is this new tool dependent on a different team? If yes, what is that team's commitment to keeping systems up and running. If the tool has dependency on a highly unreliable system with support resources outside your control, it should be considered out of the gate. Do you want to invest in a 100K a year product that likely will only work 6 months out of the year?

Recovery and Remediation

Instituting a remediation plan to address fails is another area your organization should prioritize before implementing any new tool. Understand the hierarchy of your tech stack and which tools directly rely on the new technology you're considering; if it goes down, what else will be impacted? Predicting the potential scope of an outage helps you estimate the required capacity from your teams to deal with normal errors and breakages and get your stack working again in an acceptable time frame.

Determine how your organization will track whether a new (or existing) technology is functioning as expected. Aligning on the definition of what it means for a certain tool to be considered "functioning as expected" is an important step in enabling your teams to properly run and maintain your tech stack. When something goes wrong, your recovery and remediation processes will dictate how quickly you can have your platforms back online, and preempts the same errors from being repeated.

Looking into the future, map out how a new tool will affect existing tools and processes down the line so you can set expectations for your organization. Consider how you will communicate any needed changes to your teams so you can disseminate information effectively and keep your people and your technology operating smoothly.

It's also important to determine who is needed to help troubleshoot and resolve errors as they occur. Deciding who is responsible for fixing issues with a new technology will make incidents smoother to deal with, and also gives you foresight into how equipped your organization is to handle a new piece of technology. If you find that your team is maxxed out when it comes to the workload they're already handling, it may be time to bring on more professionals to give the needed attention to your new platform. If your organization is unable or unwilling to make hires to guarantee your tech stack is able to be maintained, then it's likely better to pass on any new technologies until those resources are available.

Recovery and remediation are crucial aspects of successfully managing any tech stack, since failures are part of every organization's marketing operations journey. Preparing now and asking yourself these questions means that when the time comes that something breaks, your team will have the ability to take action and fix the problem in a way that prevents it from occurring again. Without that aspect of a solution plan, your team loses valuable time repeating this process for the same issues.

Checklist

What is the total current capacity of the Marketing Operations Team?

- □ New Campaigns & Projects
- □ Run state on existing tools and projects

What does the training and roll out look like?

- □ How many person hours will roll out take for my team?
  - Vendor selection/Feature comparison
  - Design
  - Security Reviews and Procurement
  - Setup and Integration
  - Testing
  - Training
  - Run state
- □ If a third party/peer team does the implementation, how do we validate that the integration was run successfully?
- □ What other systems does this integrate with?
  - What are the requirements for those systems?
  - What plan level is required?
  - How much time is needed to implement/test in those systems?
  - Who owns the integration?
- □ If the tool integrates with systems owned outside of our team:
  - Is this the one "ask" we want of that peer team?
  - Is there a rollout strategy where this tool makes sense if the peer integration is deprioritized?
  - If the peer teams make changes to their system after integration, could it impact our system's performance? How will we know if that happens? How will they know it has happened? What is the monthly time commitment to support that?

Impact on our technology:

- □ What is the baseline performance of our technology stack?
  - How many leads flow through the system each day?
  - What is the current latency of leads flowing through the system?
    - What will the impact on latency be post deployment?
    - Does the latency vary throughout the day (for instance, batch systems running at a certain time? Peak user times?)
  - What is our API volume and limits in our core systems? What are our 6-month peaks?
    - What is the impact on this post deployment?
    - Are there any big projects planned that will likely have an impact on API volumes?

Recovery and Remediation

- □ If this tool breaks, what goes down with it (dependencies)?
- □ How do we know?
- □ What technologies and processes will be impacted downstream? How will we notify them?
- □ Who is required to help fix/troubleshoot?

What is the objective measure of ROI:

- □ For the cost:
  - ○ Build
  - ○ Run
- □ Time investment
  - ○ Build
  - ○ Run
- □ Opportunity cost
- □ Frequency of measurement