Prismatic

The Developer's Guide to Overcoming B2B SaaS Integration Challenges

From Frustrating to Friendly

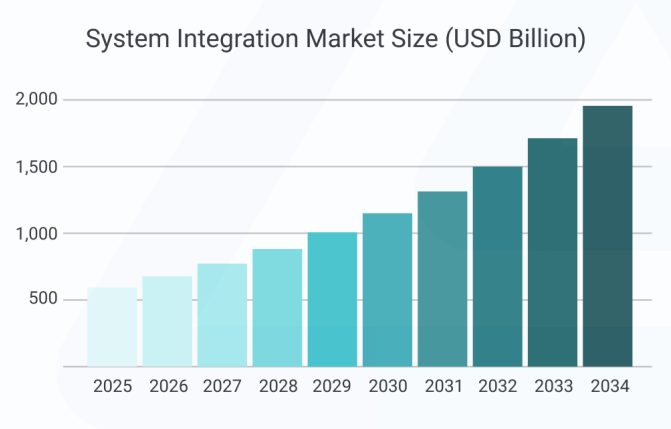
Table of Contents

Introduction

B2B SaaS integrations are essential for long-term survival in today's software market. The modern tech stack is not monolithic but integrated, with the G2 2024 Buyer Behavior Report showing that integrations are a top consideration for B2B software buyers. As companies increasingly rely on specialized SaaS solutions, expectations for application connectivity are ubiquitous. The ability to connect with your customers' other applications increases product competitivness and drives both acquisition and retention.

The stakes will only get higher. The average organization now uses hundreds of SaaS apps, creating an ever-expanding list of integration requirements that directly impact your win rates, sales cycles, and customer lifetime value. Despite this strategic importance, many B2B SaaS companies struggle with their integration approach – whether building in-house, implementing enterprise iPaaS solutions, using unified APIs, outsourcing development, or managing some complex combination of these methods.

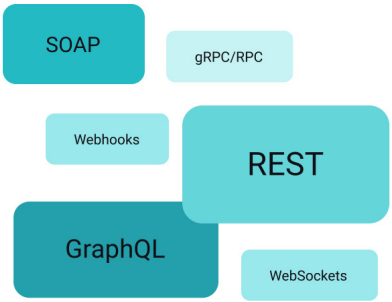
Have you ever wondered, as a developer, whether you could bypass integration development altogether? If you have, you aren't alone. Integration challenges can be overwhelming. The good news is that you can address them successfully without overloading engineering, creating additional friction for customers, or putting your product roadmap on hold.



Zoting, S. (2025, May 21). *System Integration market size to surpass USD 1,946.37 BN by 2034.* https://www.precedenceresearch.com/system-integration-market

The rest of this guide examines areas where devs who build B2B SaaS integrations face the most significant challenges. We'll explore how these issues manifest for each area and demonstrate how an embedded iPaaS approach provides better solutions than traditional in-house development.

APIs Variability



Challenge

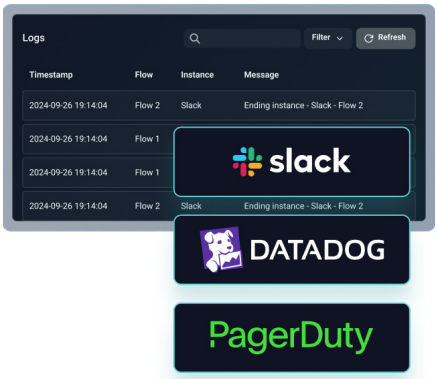
API inconsistencies create significant issues for devs. The lack of standardization across APIs leads to varied authentication methods, data formats, and rate limits. Some APIs may use OAuth 2.0, while others rely on API keys or basic authentication, forcing developers to learn and stay current with any number of auth methods.

Solution

Prismatic allows you to abstract much of the complexity of connecting with various APIs. It also includes pre-built API connectors to work with many third-party APIs, reducing the need to manually handle varied authentication methods, data formats, and rate limits. For example, our built-in connectors fully manage authentication, supporting OAuth, API keys, and other pertinent methods, allowing developers to connect popular SaaS applications without custom auth code. And, we also provide our Spectral SDK, which enables you to create custom components (including API connectors) for your product as well as niche or non-standard APIs that your customers use.

APIs

Error handling



Challenge

Error handling and resilience across APIs create significant implementation work and rework. Different APIs have varying error response formats, timeout behaviors, and retry mechanisms, making it difficult to write consistent error-handling logic. Some APIs return detailed errors with specific codes and messages, while others provide minimal feedback, and rate limiting approaches vary dramatically – from simple request counts to sliding windows with different reset periods.

Solution

Our embedded iPaaS provides robust error handling for API error responses. The platform includes comprehensive logging and can stream logs to external systems like DataDog, PagerDuty, or Slack for monitoring and detailed troubleshooting. Our built-in connectors handle rate limiting automatically, while the platform enables you to incorporate retry logic into any integration. The platform also provides alerting and monitoring features that allow you to proactively identify and resolve integration issues, in many cases, before your customers are even aware of them.

APIs

Documentation

Challenge

API documentation often lacks full explanations of endpoints, parameters, request and response formats, authentication methods, and error codes, making integration development painful. Many APIs provide only basic reference documentation without practical examples or use cases. This often forces devs to spend time reverse-engineering API behavior and testing edge cases.

Solution

Prismatic's library of connectors serves as living documentation (including connector-specific changelogs), with clear examples of how to interact with third-party APIs without requiring deep knowledge of how those APIs function. We provide extensive examples and sample code on GitHub for custom integrations, along with detailed SDK documentation for building custom components when pre-built connectors aren't suitable.

APIs

Use cases

> Prismatic is a very fleshed out product, and as we've started developing more integrations with it I'm constantly surprised at how it's able to handle niche functionality and edge cases.
>
> **Robbie T.**
> CTO

Challenge

Understanding real-world use cases and integration patterns for complex B2B integration scenarios is complicated by generic API documentation that doesn't address industry-specific or workflow-specific needs. As a result, devs struggle to translate business requirements into technical functionality, especially when integrating with niche apps that have unique data models and business logic.

Solution

Prismatic is designed for B2B software companies serving every market, providing flexible tools to build the integrations your customers need, regardless of the niche or vertical you might serve. The platform includes AI-powered integration examples and templates demonstrating sophisticated, industry-specific solutions, such as using job data to create invoices in ERP systems, importing and processing variable third-party form data, and building AI assistants that can search and interact with multiple systems. This practical approach covers building integrations and why specific patterns work best for different integration scenarios.

APIs

Versioning and deprecation

Challenge

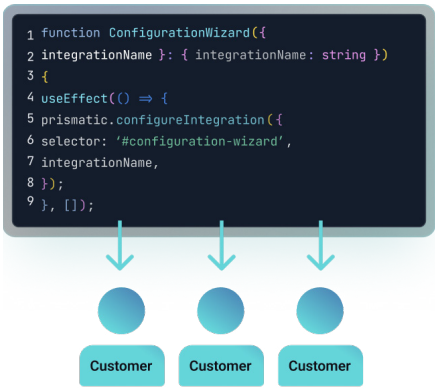
Different third-party APIs are versioned on varying schedules, and use inconsistent versioning schemes and backward compatibility support. Communicating API deprecation timelines and migration paths to customers becomes increasingly complex as B2B SaaS companies serve more enterprise clients with complex integration dependencies. These larger customers may require considerable advance notice for API changes, detailed migration guides, and extensive testing periods. However, managing these requirements across hundreds of customers and dozens of integrations is resource-intensive and error-prone.

Solution

Prismatic includes connector management and automated updates. We update the corresponding pre-built connectors when third-party APIs are versioned or deprecated, ensuring continued compatibility. We also provide version management for integrations, allowing you to test new API versions in staging before rolling them out to production. You can use the CLI and API to script these rollouts to match customer timelines and track API version usage across all customer instances. In short, Prismatic turns API change management from a resource-intensive manual process into an automated, scalable operation.

Scalability and performance

Multiple integrations



Challenge

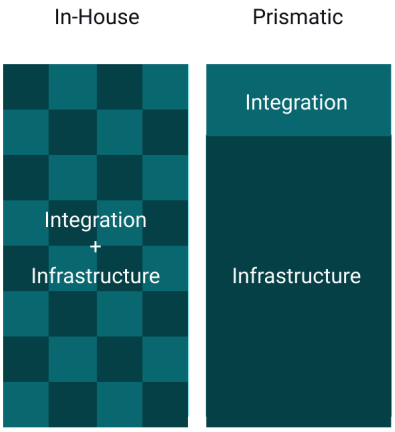
As B2B SaaS companies scale, the number and sophistication of integrations grow exponentially, creating enormous resource issues since teams can't expand linearly to match integration demands. Managing and supporting integrations once deployed becomes the most daunting aspect, as integrations rely on external systems that can unexpectedly change, potentially breaking integrations and creating support tickets for engineering and support teams. Companies must determine how to handle numerous integrations without breaking the budget.

Solution

Our embedded iPaaS addresses scaling by providing tooling that enables teams to build, deploy, and manage integrations without scaling teams linearly. The platform offers low-code and code-native building experiences, allowing non-devs to participate in integration development while giving devs the ultimate flexibility in solving complex integration scenarios. The "build-once, deploy-many" model is supported by configurations that adjust to your customer's needs. The platform's integration marketplace enables your customers to self-activate and configure their integrations, while your customer-facing teams deploy and manage integrations, substantially reducing the support burden on your engineering team.

Scalability and performance

Infrastructure overhead



Challenge

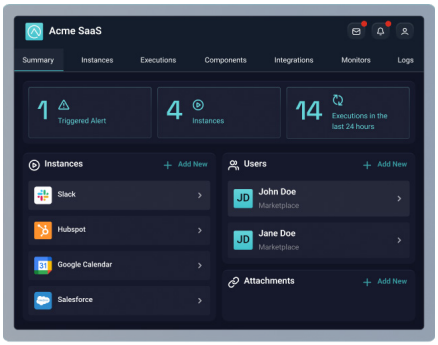
Scaling SaaS infrastructure often creates significant overhead. In-house integration approaches are particularly resource-intensive, requiring substantial investment in developing not just integration code, but also the corresponding infrastructure and tooling to manage highly variable server loads and ensure security and compliance as data moves between systems. As data volumes expand, infrastructure bottlenecks can slow performance and affect reliability, requiring complex solutions like database sharding, partitioning, and replication.

Solution

Prismatic abstracts infrastructure complexity by providing a purpose-built environment that is designed for security and scalability. The platform handles everything from auth to compute resources to webhook infrastructure to retries and more out of the box, allowing you to focus on the integration details instead of the underlying plumbing that supports the integration. The platform's cloud-native architecture scales automatically to handle bursty loads without requiring manual infrastructure provisioning or management. Prismatic also includes deployment and management tooling to completely handle those portions of the integration lifecycle within the same platform.

Scalability and performance

Monitoring and alerting



Challenge

SaaS integration monitoring becomes increasingly complex as organizations manage numerous integrations, requiring comprehensive tracking of response times, uptime, error rates, and performance. Integration-specific monitoring includes tracking dependencies on external systems that can change unexpectedly, managing integration support tickets, and maintaining visibility into integration health across all customers. Generic system monitoring tools often lack functionality for integration-specific metrics and alerts.

Solution

Prismatic provides extensive integration monitoring and alerting, including detailed logging for all integrations built on the platform and the ability to stream logs to third-party systems. The platform offers real-time visibility into integration performance, with customizable alerts to enable identification and resolution of integration issues before they affect your customers. While your team gets visibility into integration health and performance, your customers are not left behind. They can use self-service support tools, including alerts and logging, to stay current with their integrations.

Data management

Data validation and error handling

> Prismatic offers a unique approach to deploying automation across multiple clients. With Prismatic's error logs and execution logs, we can easily monitor the status of our automation.
>
> **Eveguel A.**
> Developer/Automation Specialist

Challenge

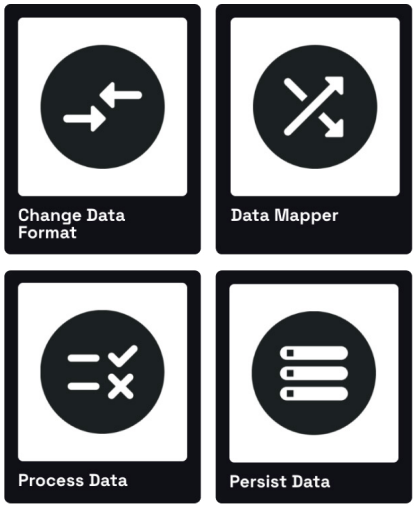
SaaS integration error handling presents significant challenges as errors are inevitable and can arise from network problems, API changes, data format mismatches, and auth failures. Without detailed error messaging, non-devs struggle to identify and resolve errors, leading to endless email exchanges between engineering and other internal teams, and creating unnecessary delays. Traditional in-house integrations often display generic error messages without handling mechanisms, making it hard for support to diagnose and resolve issues.

Solution

Prismatic provides comprehensive error management with standardized error processing and detailed logging. The platform includes sophisticated monitoring and alerting features, providing devs and non-devs with clear, actionable error information in the integration management UI. Our error handling and retry mechanisms automatically manage temporary failures, network interruptions, and API rate limiting through exponential backoff strategies. The platform also enables your customers to access self-serve support tools and monitoring dashboards, reducing the support burden on your team while providing visibility into integration health.

Data management

Data transformation



Challenge

B2B SaaS integrations bring significant data transformation challenges with disparate data formats, varying data quality, and complex data mappings between systems. Different apps use different data structures, inconsistent naming, and varied data types, leading to failed syncs, broken workflows, and data integrity issues like duplications or incomplete transfers. These challenges become particularly complex in scenarios like multi-stage data processing, where data must be converted through multiple intermediate formats before reaching its final form.

Solution

Prismatic includes powerful data transformation out of the box to handle field mapping, data type conversions, logic operations, and more. The platform offers low-code transformation functionality for non-devs and advanced coding capabilities for devs who need flexibility to implement complex business logic.

Data management

Security and compliance

Challenge

B2B SaaS integrations must often comply with regulatory rules and frameworks, including GDPR, HIPAA, and CCPA, as well as industry-specific standards, each with different rules for data encryption, access controls, breach notification, and data subject rights. Integration scenarios create additional compliance complexity as data flows between multiple systems and service providers, making it hard to maintain clear data ownership and establish proper data processing agreements. Each integration interface represents a potential security risk, requiring devs to consider data encryption, access controls, and audit trails across connected platforms.

Solution

Prismatic provides built-in enterprise-grade security and compliance, including data governance features and security controls designed for B2B SaaS environments. The platform supports robust auth handling, end-to-end encryption, and detailed audit trails. The platform also supports role-based access controls and thorough documentation of data processing activities, helping organizations demonstrate compliance with privacy frameworks like GDPR.

Development and operations

Testing and debugging

> Prismatic doesn't just stop at deployment – it's built with developers in mind throughout the entire integration lifecycle. Its robust support and maintenance tools ensure smooth operation at scale, sparing us the headache of troubleshooting and debugging in production environments.
>
> **Mandana Jamei**
> Data Engineer at SoundThinking

Challenge

Testing integrations requires teams to consider everything from multi-tenancy complexities to third-party API idiosyncrasies while handling configurable and non-configurable functions and ensuring data privacy and security for all customers. Testing concerns include cross-browser compatibility, continuous updates with shorter testing cycles, and performance bottlenecks.

Solution

Prismatic handles unit testing with the Jest testing framework (or you can plug in a different framework of your choice). The platform provides real-time debugging tools that give you deep visibility into integration execution, with comprehensive logging to enable quick identification and resolution of issues. Our code-native building experience allows your team to test integrations in controlled environments before deployment, while Prismatic's integration designer makes it easier to identify and debug integration logic within the UI.

Development and operations

Deployment and release management

Challenge

Integration deployments are far from simple, including manual steps in release processes, complexity in managing multiple deployment pipelines, and the pressure to deliver features rapidly while maintaining quality and avoiding downtime. Poor communication often results in unnecessary changes and deployment delays, while weak infrastructure can hinder scaling and efficient use of compute resources. Limited testing tools and coordination challenges in configuring test environments also create problems for CI/CD pipelines, leading to overlooked test cases.

Solution

Prismatic comes with comprehensive deployment and release management, supporting automated and manual deployment scenarios. The platform includes a deployment UI for customer success teams and provides scriptable deployment via our CLI and API. Prismatic enables automated deployment of integrations with proper version control and metadata management, supporting continuous deployment practices. You can also choose to have customers self-activate and configure integrations through the embedded marketplace, reducing your deployment overhead while maintaining a controlled process.

Development and operations

Resource constraints

> We probably save 95% of [engineering] time. We've deployed far more integrations than we would have without Prismatic.
>
> **Alttaf Hussain**
> Director of Engineering at Yoti

Challenge

Scaling integration development often runs afoul of resource constraints, including recruiting devs with the technical and client-facing skills needed for integration work. Resource challenges are often exacerbated by a disconnect between leadership and product teams, with early success leading to bottlenecks as demand outpaces ad-hoc processes, resulting in highly customized build, deploy, and manage processes and highly frustrated devs.

Solution

Prismatic includes tools for your devs and non-devs to develop integrations, significantly expanding integration capacity without requiring linear scaling. The platform's low-code integration designer allows non-devs to build and maintain integrations, while you focus on writing code to handle the detailed business logic required. And, Prismatic's embedded workflow builder shifts one-off custom development to your customers, allowing them to create the additional workflows they need with their niche apps and home-grown systems. The platform also provides extensive deployment and management dashboards for your customer-facing teams, reducing the burden on engineering resources while delivering a first-class customer UX.

Development and operations

Vendor lock-in

> As a software developer, I love how easy Prismatic makes it to create integrations that can be reused by many of my customers with minimal effort on their part.
> Prismatic also offers me the flexibility to write all or part of an integration in TypeScript for when I just need to get do something in a very specific way or when I need to solve a problem that I know I can do very efficiently in code.
>
> **Verified User via G2**

Challenge

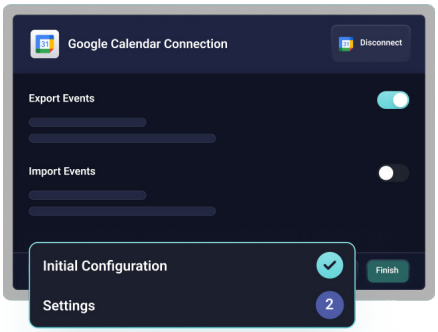
Vendor lock-in happens when companies depend on proprietary integration technologies, data formats, or platforms with technical, financial, and legal restrictions. Lock-in challenges include a lack of communication between existing management tools, incompatibility issues with on-premise software, and an inability to move to another service provider or migrate integrations in-house. Vendor lock-in creates dependency on proprietary systems where custom APIs, bespoke integrations, and ecosystem-based dependencies prevent migration from particular platforms.

Solution

Prismatic lessens vendor lock-in risks by providing flexible development options for low-code and code-native development, enabling teams to control their integration logic and data flows. The platform supports standard APIs and protocols while providing a powerful SDK for custom component development using TypeScript.

Communication and collaboration

Scope and requirements



Challenge

Unclear scope and requirements create significant B2B SaaS integration project bottlenecks. Sophisticated customers often have complex, varied requirements that demand high levels of customization, and business stakeholders struggle to define specific integration needs until deep into implementation. Customers often compound this with bespoke requirements for features, workflows, and user interfaces. Teams frequently discover more requirements mid-project, leading to scope creep and budget overruns.

Solution

Prismatic lets you manage scope and requirements through an integration lifecycle that supports consistent, repeatable processes. Our drag-and-drop designer allows you to visualize integration flows, facilitating requirement communication and reducing misunderstandings between technical and business teams. Integrations can be configured for each customer, letting you address variable requirements without forking the code or rebuilding from scratch. The integration marketplace and deployment tooling provide a standardized process for rolling out integration changes, while comprehensive version control ensures that scope changes can be managed systematically rather than reactively.

Communication and collaboration

External team dependencies

> Prismatic shines in its user-friendliness and scalability. The platform is exceptionally performant, making it a reliable choice for diverse integration needs.
>
> **Blake D.**

Challenge

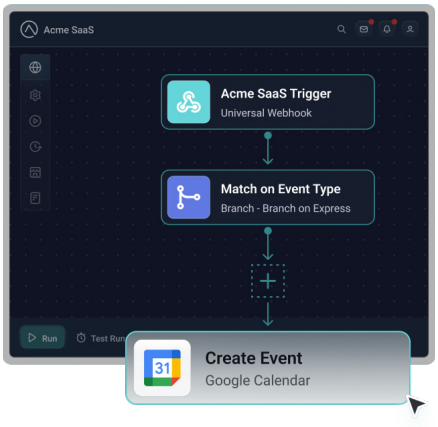
External team dependencies create significant risks because integrations rely on external systems that can unexpectedly change, potentially breaking carefully crafted integrations. When using third-party solutions, teams inevitably depend on the provider, including technical dependency for system availability and vendors needing to add functionality to their connectors. APIs and SaaS apps constantly change, and integration connectors can quickly become obsolete if providers don't keep up with those updates, leaving teams at the mercy of external vendors.

Solution

Prismatic lets your team manage the complete integration lifecycle while reducing reliance on external vendors. We proactively maintain and update pre-built connectors, monitoring API changes across those third-party apps so your team doesn't have to. The platform provides flexibility to build any integration your customers require, including the ability to develop custom connectors or write code when needed. Our code-native development options help you maintain control over your integration logic and move quickly when external systems change.

Communication and collaboration

Internal team dependencies



Challenge

Internal team dependencies create significant scaling challenges, as teams struggle to scale integrations without scaling the teams (engineering, onboarding, and support) that work with integrations. Organizations must figure out how to do more with less and execute multiple times the number of integrations while keeping teams within budget. Integration projects often require cross-functional teams with expertise in various applications, data formats, and technical protocols, creating bottlenecks when specialized knowledge is required.

Solution

Prismatic supports a model where non-devs (or customers) configure new instances of known integrations, devs get involved only when there's something new, and support monitors day-to-day operations. This allows your customer-facing teams to handle deployments while freeing you to focus on adding value. That platform also empowers non-devs with a low-code integration designer, enabling teams to create simple integrations quickly. Our code-native experience ensures ultimate flexibility for handling complex scenarios. Together, these approaches ensure that integration work is efficiently distributed across skill levels.

Communication and collaboration

Customer self-service and UX

> [Our customers] are happy. They feel in control. They feel empowered. And [Prismatic] opens up even more opportunities that our customers haven't even thought about. They came in wanting integration X and they see it there and they're like, oh, look, it's integration Y and Z. Let's also enable those.
>
> **Dustin Diaz**
> Head of Engineering at Duro Labs

Challenge

Customer self-service and UX are often the last priority as B2B SaaS companies struggle to balance powerful integration capabilities with user-friendly experiences. Building integrations with every function fully exposed to customers who thoroughly understand all the technical complexities is impossible. In-house integration approaches require customers to rely heavily on support teams for selecting, activating, configuring, deploying, and monitoring their integrations, creating bottlenecks and friction for customers. Many customers want integration UI, but providing and maintaining it is cost-prohibitive.

Solution

Prismatic enables customer self-service by empowering your customers to self-activate integrations with an in-app marketplace or have customer success teams deploy integrations for them. Our platform provides guided config wizards that walk users through setup, while the integration marketplace allows your customers to select and configure integrations independently of internal teams. We also provide your customers with access to integration logs and alerts, reducing the support load. You can set up your customers to build, deploy, and manage custom workflows while ensuring consistency with the rest of the integration UX you provide.

How we help

We provide you with the tools you prefer and all the benefits of a managed platform. By offering both low-code and code-native build paths, extensive customization through Spectral, and complete lifecycle management, Prismatic enables you to build any integration your customers need while ensuring maximum productivity for you, your dev team, and any non-devs that are part of integration efforts.

Most importantly, Prismatic provides you with a secure, scalable integration platform that includes everything from auth to automatic retries, ensuring that you and your team can spend your time building integration business logic, and not fussing about with all the behind-the-scenes stuff that shouldn't be recreated for every integration.

Schedule a demo to see how this makes us the clear choice for devs who want to bring integrations to market in the most efficient way possible.

What to look for in an integration platform

There are a lot of integration platforms out there, and sorting through them isn't getting any easier. The usual categories – like embedded iPaaS, enterprise iPaaS, and unified APIs – are starting to blur as more vendors rush to brand everything as "AI-driven."

Labels aside, what matters is what makes everything about integrations better for you as a dev.

Here are the key features you'll want to look for in an integration platform:

Infrastructure

- ☐ Scalability and performance
- ☐ Security and compliance

Build

- ☐ Low-code designer option
- ☐ Code-native development option
- ☐ Custom connector development SDK
- ☐ Embedded workflow builder (for customers to build)
- ☐ Built-in testing framework
- ☐ Thorough dev documentation
- ☐ Interoperability with existing dev tools
- ☐ End-to-end developer experience

Deploy

- ☐ Flexible deployment options
- ☐ Integration marketplace
- ☐ Improved time to value

Manage

- ☐ Monitoring and logging tools
- ☐ Versioning and lifecycle management
- ☐ Full deployment and support UI
- ☐ Customer self-service for activation, deployment, and support
- ☐ Responsive vendor support

Prismatic