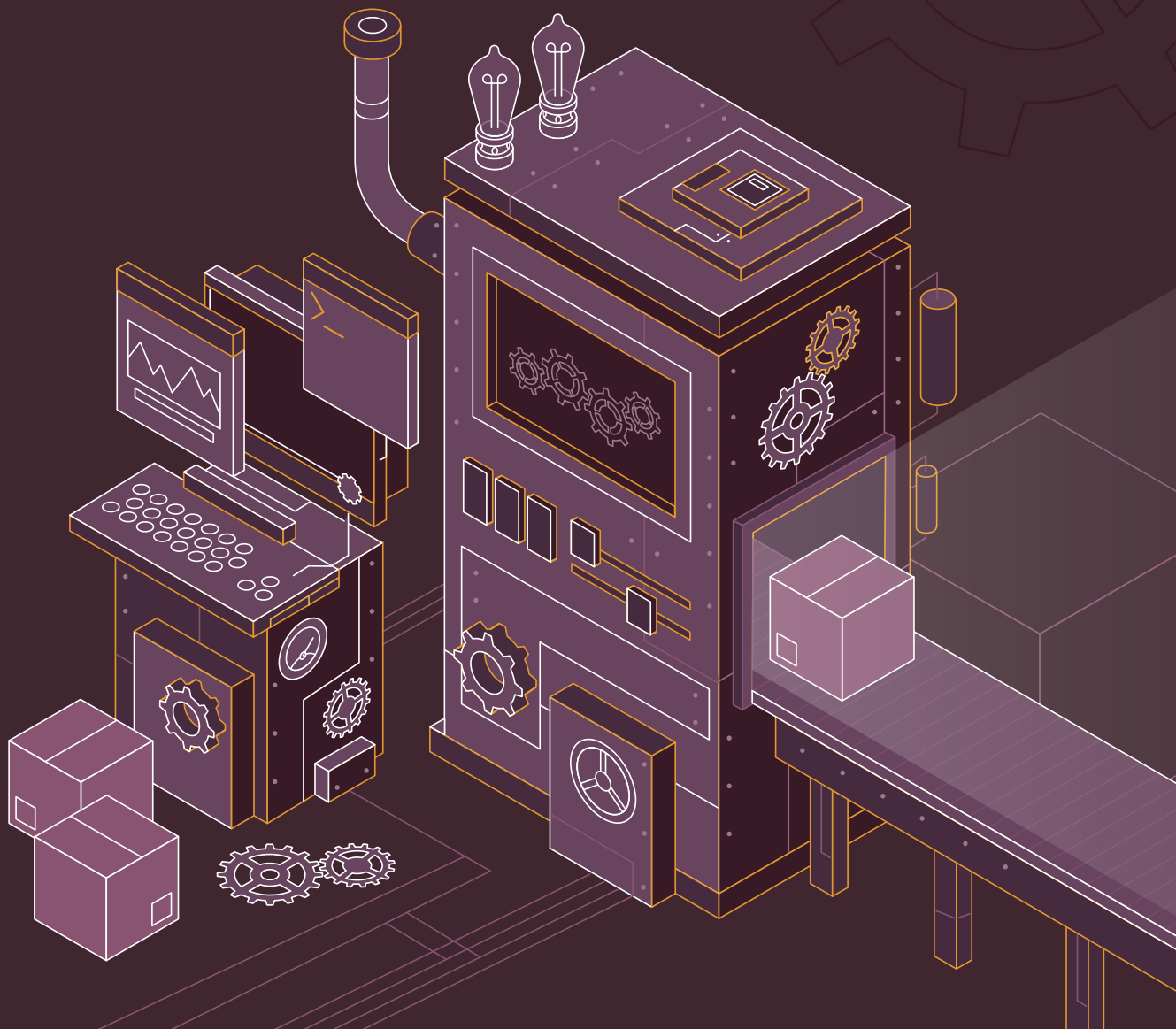
XLAB Steampunk

The importance of high-quality Ansible Collections for your product



ABOUT THIS BOOK

Objectives

This book introduces the importance of high-quality Ansible Content Collections, explains the importance of documentation and testing, and illustrates the value with a case study.

Here's what you'll learn:

- Why Ansible Collections add value to your existing solution
- Why it's important to invest into a high-quality Collection and not develop it as a side project
- What "high-quality" actually means and how to ensure it
- How Sensu managed to attract new customers with their Ansible integration

Intended Audience

This book is written for product teams who already have the basic understanding of Ansible. We assume that your solution can be system integrated and you'd like to understand the meaning and value of Ansible Collections.

Beyond this book

If you find this book useful and want to learn more, take a look at https://steampunk.si/blog/, where we publish new content about Ansible and Content Collections. If you can't find what you're looking for, simply shoot us an email at steampunk@xlab.si.

Contents

CHAPTER 1

How to benefit from Ansible Collections

Today, we can't imagine managing complex IT environments without efficient automation, and luckily, we don't have to owing to ingenious tools such as Ansible.

More than four million customer systems worldwide are automated by Red Hat Ansible Automation Platform, so there's a big chance your clients are using it. And an even bigger chance they want to use it with your product. Help them by integrating it with Ansible.

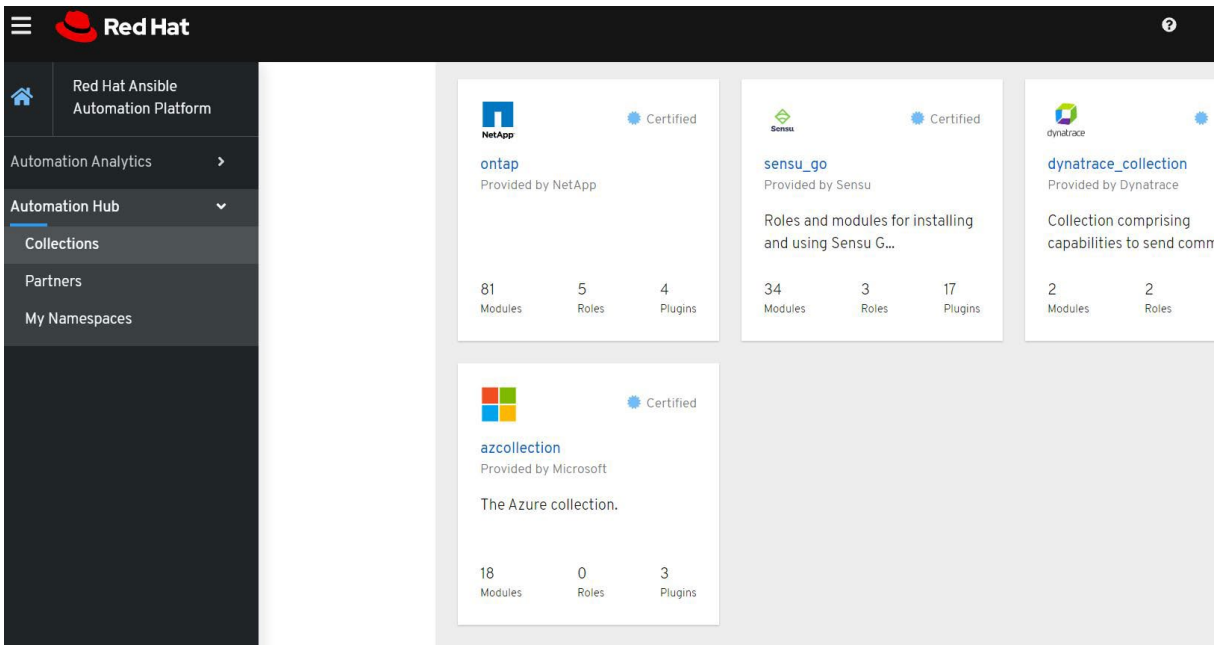
How?

By designing, developing, maintaining and supporting Ansible Collections.

What are Ansible Collections and why you shouldn't miss out on them

To make enterprise automation even more efficient, Red Hat launched Red Hat Ansible Automation Platform and combined powerful Ansible solutions with new capabilities, such as Certified Content Collections and Automation Hub. **Collections are a new method of distributing, managing, and consuming Ansible content.**

You can find certified, fully supported Ansible content, such as modules, roles, plugins, playbooks all in one place, called the Automation Hub.



Certified Collections in Automation Hub

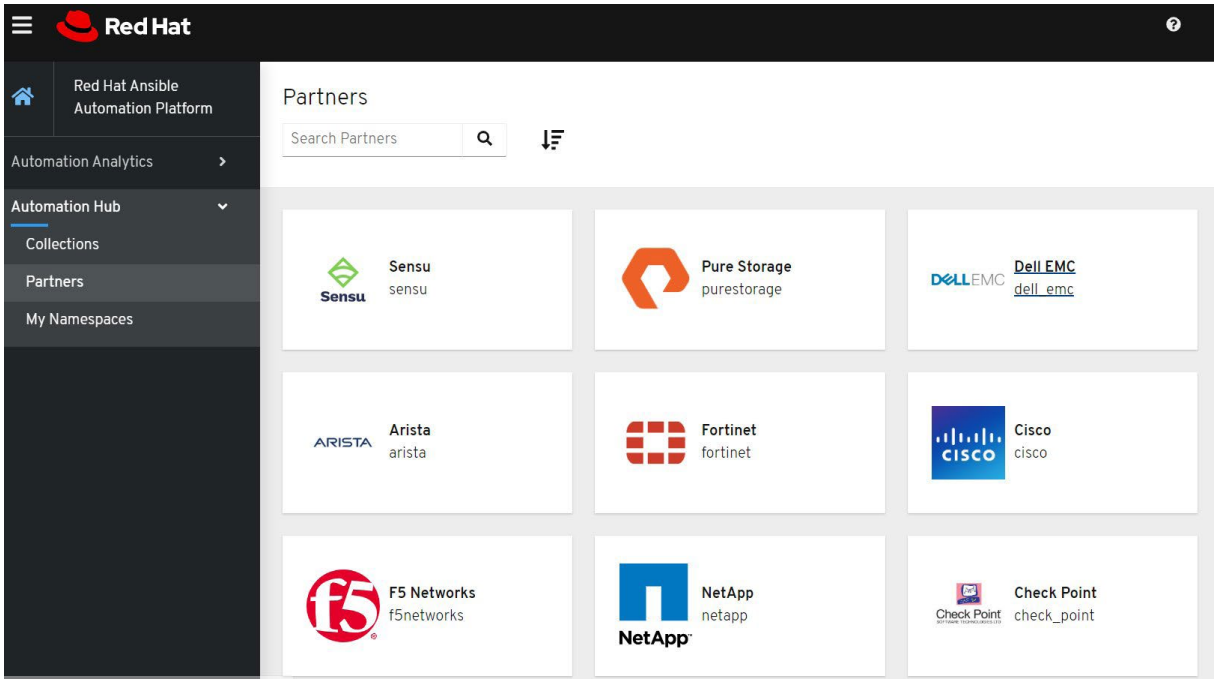
And the platform did not only bring a whole new way of organizing Ansible content but a whole lot of benefits, too. For you and your users.

Ansible Content Collections simplify and accelerate delivering and consuming content. With the introduction of collections, Ansible content no longer needs to be a part of the Ansible core, so it's no longer tied to the Ansible release cycle. That means you can keep up the pace with the demands: new functionalities of existing service or a completely new service can be rolled out along with the ability to automate it. Modules can now be improved and updated independently of Ansible release cycle.

And because collections come prepacked with Ansible modules, roles, plugins, and also the corresponding documentation, your users can start automating right away.

How to benefit from Collections

Another big benefit collections bring is that all content in Automation Hub is certified - that means fully supported and maintained by Red Hat and partners. Your users always have someone to turn to for help if issues occur (and that someone doesn't have to be you).



Ansible Automation Platform Certified Partners

Certified Ansible Collections doubtlessly bring comprehensive benefits. They enable building focused, defined packages of content, enable users to consume fully formed solutions, and they make a product stand out.

CHAPTER 2

How to attract new users with high-quality Ansible Collections

The IT industry has been increasing in complexity for quite some time now. Organizations have to find a way to manage an additional maintenance burden. Most of them are trying to solve the problem via automation. But this cultural shift towards the DevOps and site reliability engineering disciplines has one **significant consequence for product owners: automation is becoming the preferred method for adopting new technologies**.

Go from manual and time-intensive to automated onboarding of users

If you want to show the value of your solution by onboarding as many end users as possible within one company, you've got to simplify the process. With automation, you can forget about one-on-one onboarding and leave it to Ansible playbooks.

This is exactly what Sensu did. With our help, they created Ansible integration for their product with the goal of easier onboarding of new users.

In the words of their CEO, Caleb Hailey:

"Once a new company starts using Sensu, we want as many people in the organization as possible to start using our product. If it's a kind of company where they are fully bought into Ansible as their primary method of adopting new technology to help advance their business, offering a high quality Ansible Collection helps us get new customers AND helps their usage to expand over time."

High-quality Ansible Collection can help **eliminate barriers to adoption and improve the reliability of on-boarding new technology**. But the benefits of offering customers a high-quality Ansible Collection extend beyond the day-1 operations. When appropriately designed, Ansible roles and modules can be of great help in day-2 operations.

Improve your solution engagement

If your solution delivers value on a daily basis, it's a sure sign that it's been completely adopted and embraced throughout the company.

Let's explain this with another example - Ansible integration for Atlassian Software Suite.

The most popular app from their Software Suite is Jira. It's used to create tickets, track development workflow, plan sprints, and pretty much anything else that an agile team could need.

Now, managing the workflow, even with a tool like Jira, can be quite a hassle. Ansible integration simplifies the use of Jira by providing us with a set of built-in tasks that can be used to create and manipulate tickets.

As a result, this sets up Jira for customer satisfaction and internal success.

From updating the app to operating and maintaining it, everything is taken care of by simple automation. Rather than wasting their valuable time and concentration on dealing with these simple tasks, users are fully supported and liberated.

What makes an Ansible Collection high quality

A high-quality Ansible Collection puts the user experience above all else. Its two primary purposes are:

1. Make it possible to write human-readable Ansible playbooks that describe the desired state.

2. Enforce the desired state from the Ansible playbook safely and reliably.

Because Ansible Collections provide essential building blocks for Ansible playbooks (modules and roles), readability and robustness are highly dependent on the quality of the Ansible content.

How to design high-quality Ansible modules

Trying to get developers to agree on a single definition of a good API is mission impossible. But this does not mean that we should stop trying. Here are three things we learned about the Ansible module's API design while developing Ansible Collections for almost two years.

Each module should address a relatively well-defined use case that will guide the development process. Just wrapping an underlying API is not the best option because they are often too low-level for end users.

The module's API should describe the operation's result, not the procedure itself. Such APIs drastically reduce the need for conditional processing in Ansible playbooks and make it easier to reason about the managed system's state.

If the Ansible module talks to a remote service, it should have its connection options separated from the rest of the parameters that describe the task's result. Alternatively, the module can use a custom Ansible connection plugin and ignore connection options entirely.

CHAPTER 3

The Value of Documentation and Integration Tests

It is a well-known fact that real programmers do not document their code. If it was hard to write, it should be hard to understand and use. But this attitude might prove itself to be a bit problematic if we would like to end up with a high-quality Ansible Collection.

Ansible module documentation reliability and compliance

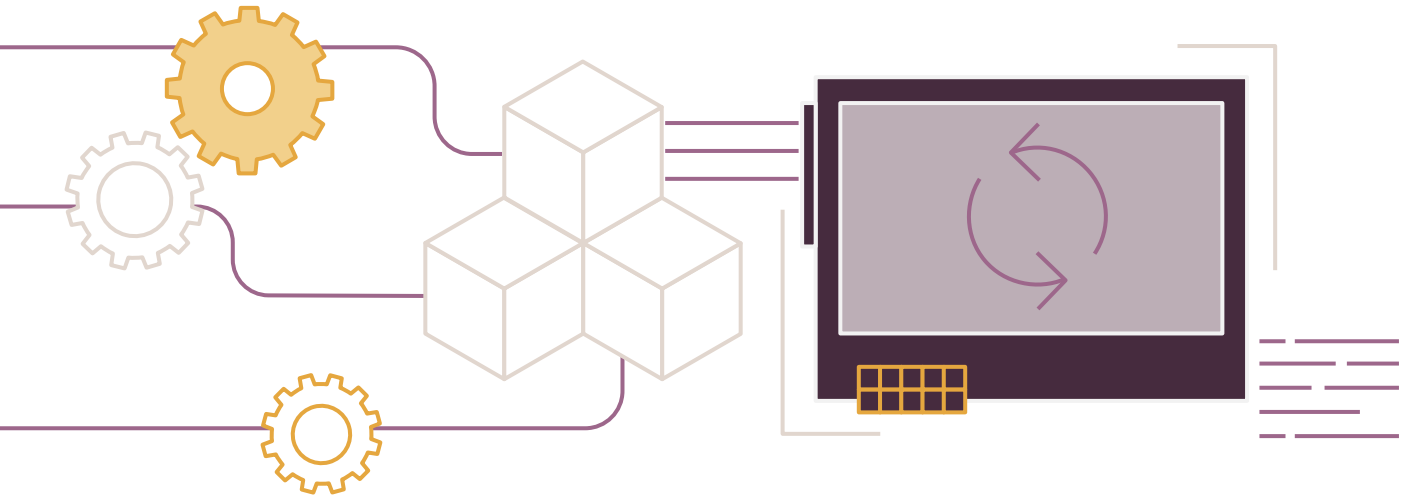
Module documentation is an entry point for Ansible playbook authors. So it is vitally important that we keep the said documentation in sync with the module implementation or risk getting angry bug reports. But here lies the problem.

Maintaining the synchronization between documentation and implementation is not as straightforward as it could be. Why? Because each Ansible module contains

two copies of parameter-related information. Module maintainers must first document each module parameter and then copy a slightly transformed description into the parameter validator.

Currently, there is no way to remove the information duplication from the Ansible modules.

In a perfect world, Ansible playbook authors would first read the API documentation and then write a task without making any mistakes. But unfortunately, we live in a world where mistakes do happen, and most people only read the documentation when something goes wrong.



Dealing with duplication - the Ansible way

So, how is Ansible currently dealing with the problem of information duplication? In short, it does not. What Ansible does offer is a way of detecting the desynchronization between the two copies of information.

The validate-modules sanity test will report the discrepancies between the API documentation and argument specification, but we still need to resolve them manually.

This detect-and-fix approach works pretty well if we are working with existing Ansible modules. Parameter changes are usually small in such scenarios, which keeps the error message count low. But things start to go downhill if we are writing new Ansible modules. In cases where the Ansible module contains a non-trivial amount of parameters, we can quickly end up with hundreds of errors.

Adding integration tests

If we were able to write perfect code all of the time, there would be no need for tests. But unfortunately, this is not how things work in real life. Any modestly useful software has deadlines attached, which usually means that developers need to strike a compromise between polish and delivery speed.

For us, the Ansible Collections authors, having a semi-decent Collection of integration tests has two main benefits:

1. We know that the tested code paths function as expected and produce desired results.

2. We can catch the breaking changes in the upstream product that we are trying to automate.

The second point is especially crucial in the Ansible world, where one team of developers is usually responsible for the upstream product, and a separate group maintains Ansible content.

Manually testing Ansible modules

The most straightforward integration test for an Ansible module is a playbook that has two tasks. The first task executes the operation and the second task validates the results of the first task.

And while manually testing modules is simple, it does not scale to more than a few modules. Usually, we would need to write a script that runs all of the tests. But luckily, Ansible comes bundled with a tool aptly called ansible-test that can do this for us.

Automate the automation tests

The ansible-test knows how to perform a wide variety of testing-related tasks, from linting module documentation and code to running unit and integration tests. But before we can use it, we must prepare a directory structure for it. Even though there's a bit more preparation, the benefit lies in automating the execution of these tests.

While having integration tests for modules is a great start, there are other things that we should test if we are serious about creating a robust Ansible Collection:

- testing the built-in documentation,
- linting the modules,
- writing unit tests,
- preparing integration tests for other kinds of Ansible plugins, and
- integrating with CI/CD.

CHAPTER 4

Should you build or outsource the development of Ansible Collections?

Although developing Ansible Collections internally gives you immediate insights and greater control, it comes at a cost, especially when you're lacking the proper resources. Contrary to what some people believe, Ansible integration shouldn't just be a side project for your developers - not if you really want to reap the benefits for your product. It needs to be well-planned, documented, executed, supported, and maintained to serve its purpose.

So, what are the key factors to consider before committing to outsourcing?

Allocation of limited resources

Quality of integrations

Ongoing costs

Certification of Ansible content

Let's take a closer look at each of them.

Allocation of limited resources: keep the focus on your core product

Let's face it, your product team is limited. Jira tickets are already piling up, and there's no way that you can hire someone immediately. On top of everything else that's going on, it's time to develop Ansible integration and get the initial traction from Ansible users.

You're faced with two choices:

- Dedicate a couple of people away from your core product
- Make Ansible integration an outside-of-work-hours project

Dedicating resources away from your product means that you'll have to delay the release of some core features and fixes. There's no way around this when there's a chunk taken out of your product team. This will negatively impact the release timeline and present a setback. The best-case scenario? A feature gets released a couple of days later. And the worst case? Your existing customers start to get irritated by the time it's taking you to fix simple bugs. This makes your competitors' solutions more appealing and increases the risk of churn. Instead of increasing conversions with Ansible integration, you'll end up with people questioning the quality of your core product.

By outsourcing the work, your team will continue to focus solely on the product. **You'll end up with a high-quality solution AND a high-quality Ansible integration.**

What about the option to develop Collection as a part of an outside-of-work-hours project? This leads us onto the next point.

High-quality Ansible integration = Happier customers

Creating an Ansible integration is not that challenging. Creating a high-quality collection? That's a whole different story. But why isn't any integration good enough? Because low-quality Ansible content won't provide a good user experience.

You can treat Ansible integration as a side project for your Dev team. However, this means that there won't be a project manager, the integration won't be carefully planned and designed, the development will be extended over a significant period of time, there'll be limited quality assurance, and you'll end up with exactly what you started with - a side project.

Ansible integration as a side project is often an **easy fix rather than an enrichment of your core product**. In contrast, **outsourcing content creation to Ansible experts will result in a high-quality Ansible Collection**.

How does the quality of Ansible Collections impact user experience?

You can identify high-quality Ansible content in the following ways:

- **Check documentation**: at the very least, it should contain a quick-start tutorial with installation instructions and reference documentation for Ansible modules and roles.
- **Assess playbook readability**: modules should have a consistent user interface and descriptive parameter names.
- **Test basic functionality**: check the basic functionality of each Ansible module - enforcing state instead of executing actions and supporting check mode are the bare minimum.
- **Peek at the tests**: checking the CI/CD configuration files should give you a general idea about the test suite's robustness - you are looking for integration and sanity tests.
- **Inspect the issue tracker**: a large number of reported defects with no response from maintainers is a bad sign.

How does the user experience tie into this? **The user experience defines the criteria for a high-quality Ansible Collection.** You need to provide a concise and robust getting-started guide to get your users up and

running as effortlessly as possible. You need to make sure your Ansible modules work as promised (by running tests) and address any issues found as soon as possible if you want users to trust your Ansible content. And you need to carefully design module interfaces to keep users' Ansible playbooks consistent, short, and readable.

Without high-quality content, your users won't benefit from the integration in a simple and efficient way. With that said, it's hard to develop a good integration without full-time commitment, user research, and planning.

Ongoing costs: maintaining and supporting your Ansible content

Are you reading this because your only concern is ROI?

Here's the truth: you won't financially benefit from developing Ansible integrations internally, if you treat it as a side-project. The integration itself might come at a lower cost (and quality), but **there's an ongoing commitment to maintaining and supporting the Collection.**

Maintenance

Ansible is continuously improving. If you want your users to reap the benefits of those improvements, as an Ansible Collection maintainer you need to follow Ansible updates and community announcements.

This means **dedicating extra hours each month** just to make sure that your content is aligned with the general Ansible rules and that it's adapted to newer versions. When an update is released, you'll need to adapt your content respectively, thus investing extra resources and (if done improperly) worsening the quality of your integration. With all the changes happening in the Ansible world, it can really be an intensive activity to keep up with the community.

When outsourced, **your content will always be up to date** without any added hassle on your end.

The users will experience the best possible integration and you won't need to expend your own resources.

Support

Things break, and that's ok if there's a dedicated team there to fix them. If not, be prepared for a bad experience. Since the whole point of integrating your product with Ansible is to create a better user experience, it'd be a shame if you dropped the ball on maintenance and support.

When outsourcing Ansible Collection to a team like XLAB Steampunk, maintenance and support are a part of the ongoing process. We'll make sure that everyone gets the best possible experience of your product via Ansible integration.

Certify your Ansible integration for enterprise use

All Ansible content can be published to Ansible Galaxy, as this is the place for all community-generated content. However, there is no guarantee for users that the content will be properly maintained and supported. Enterprise customers require enterprise support and Automation Hub is the place where this kind of Ansible content is available.

In order to publish your content on Automation Hub, you have to certify your Collection with Red Hat. Here's what it means to have a Certified Collection:

1. Your content needs to pass Red Hat quality requirements.
2. It needs to be compatible with supported versions of Ansible and your product.
3. It needs to be tested and validated.
4. There's shared support between Red Hat and its partners (vendors or XLAB Steampunk). When a customer opens a support ticket, it'll be resolved by the appropriate team.

Once certified, the content will **signal the market that it's an enterprise-grade Collection, which is critical for your enterprise customers**.

When outsourcing the creation of content, the Ansible experts will make sure that your Collection is ready for enterprise use.

Outsourcing will get you a high-quality Ansible Collection

Let's start by reviewing the advantages of creating content in-house:

- Total control over dedicated resources,
- immediate insight into the work,
- lower initial investment (but not necessarily),
- ... and that's about it.

On the other hand, outsourcing the development to experts like XLAB Steampunk will land you:

- A **high-quality integration** developed by a dedicated team of Ansible experts.
- An ongoing **maintenance and support** of the Collection without dedicating any of your internal resources.
- A time to **focus all your resources towards creating and improving your core product**.
- Enterprise-ready **Collection that can become part of the Red Hat Ansible Automation Platform**, easily available to all Ansible enterprise users.
- **Satisfied customers** that appreciate your efforts to deliver high-quality experience!

In the next Chapter, we'll explore the power of outsourcing in a real-life example.

CHAPTER 5

Customer Story - Sensu

In order to provide a great experience to its users, Sensu decided to integrate its product Sensu Go with Ansible. The team outsourced the integration to XLAB Steampunk for the best possible result. How did our collaboration work out for them? Did Sensu's Ansible Collection help them reach their goals?

For the purpose of this e-book, we chat with Caleb Hailey, co-founder and CEO of Sensu, who helps businesses solve their stickiest monitoring challenges.

Meet Sensu

Sensu is an Observability Pipeline that delivers Monitoring-as-Code for any cloud infrastructure. While some monitoring tools have been outpaced by high-velocity changes in their IT infrastructure, the Sensu monitoring solution empowers DevOps and SREs to automate their monitoring workflows, integrate existing best-of-breed tools and achieve instant visibility into their multi-cloud environments.

Goal: A great automation experience

Complexity in IT has increased immensely over the last decade, and organizations adopting DevOps practices are looking to manage that complexity through automation. Since Ansible plays a huge role in the world of automation, Sensu decided to add an Ansible Collection into its toolbox to offer an enhanced user experience.

"If that's how you want to onboard new technology in your organization, we want you to have a great experience onboarding Sensu. If Ansible is your preferred choice, we want you to have a great experience using your preferred tools," says Caleb.

They knew that the development of a high-quality Ansible Collection can be quite a challenge, so they chose to outsource it. Why did they choose XLAB Steampunk?

Recommended by Red Hat Ansible themselves

Sensu wanted to become an Ansible Certified partner and as they talked to the Ansible team about various levels of partnership, they asked who could help them to achieve their goals.

In fact, Caleb had a specific goal in mind: *"I wanted our collection to be on the shortlist of examples that Ansible team would point to and say 'If you want to do it right, this is what it should look like.' "*

"With those parameters in mind, they recommended XLAB Steampunk. Your name was on a very short list," says Caleb. *"I don't even remember any alternatives, as your name was on the top of the list. So we just said we were going to pick the number one."*

Sensu Collection sets an example at AnsibleFest 2019

Yes, that actually happened. Ansible Collections were a brand new thing. Red Hat presented them for the very first time at AnsibleFest 2019 and Certified Sensu Collection was available to users on the very same day. On top of that, the Ansible team recommended XLAB Steampunk to lead a session on how to build Ansible Collections and Sensu Collection was selected as an example of good practice.

"For me, that was the ultimate success. That was what we were going for and we got it. I can't speak on behalf of my own use as I'm not an Ansible user, so seeing Sensu highlighted as an example of how to implement the latest & greatest Ansible best practices was the ultimate litmus test for me," says Caleb.

But it's not about the praise, it's about users. So, what are the tangible results and benefits of adding an Ansible Collection to Sensu?

Ansible means you can expose more tools to more users

Ansible's ability to manage complexity with the help of automation brings benefits to vendors, as they can expose more tools to more users. And that helps Sensu.

"Once a new company starts using Sensu, we want as many people in the organization as possible to start using our product. If it's a kind of company where they are fully bought into Ansible as their primary method of adopting new technology to help advance their business, offering a high quality Ansible Collection helps us get new customers AND helps their usage to expand over time. We've seen notable results along these lines in the first year of our engagement together," says Caleb.

Since Sensu started publishing new Ansible content, they've had several new customers telling them: *"I tried your product a couple of months ago" but they didn't finish setting it up for one reason or another, "But then I saw that there was an Ansible Collection for it, so I tried again and I got it set up and started using it right away across thousands of devices."*

"Now more Ansible users are adopting Sensu, and some of them have already converted to paying customers," Caleb adds.

That's always fantastic to hear. However, there's always room for improvement and we wanted to know what Caleb thinks of our collaboration so far.

What is the experience of outsourcing development of Ansible integrations to XLAB Steampunk?

"You are patient and very easy to work with, even if we've had difficult requests at times. You understand our no-compromise desire to deliver a high-quality experience for our Ansible users. We can develop our product on our own, but to ensure that our product provides a great experience for Ansible users, we need input and feedback from a trusted partner. And you've been that for us," Caleb says.

"We don't always have a great experience when outsourcing projects like this. Consulting is hard. It often doesn't work out how we'd like it to. However, the dream of the ideal outsourcing outcome is still possible with the right partner. What I've learned from working with you is not to give up on consulting. It restored my faith a little bit," Caleb concludes.

The high praise really means a lot to our team. And what would Caleb say about XLAB Steampunk when talking to a friend?

"If you're trying to deliver a high-quality experience around your product to Ansible users, don't try to do it yourself. Hire XLAB Steampunk - you're going to get a better result."

CHAPTER 6

How to get started

Would you also like to integrate your product with Ansible? Don't know where to start? Well, follow these steps.

1. Is your product a good fit with Ansible?

There are several variables you need to consider. Here are the questions - answering "yes" to at least one of them will already make your product a great fit with Ansible:

1. Do your competitors have Ansible integration?
2. Is deployment of your product a multi-stage process?
3. Can users configure your product with a config file or through an API?
4. Are your customers already using Ansible?

2. What would you like to achieve?

Be as specific as possible when defining the use cases for your integration.

When defining them, think of users first. What would you like for them to gain with the integration? How is it going to help them better understand and leverage your solution?

These are the mandatory questions. Don't create an integration just for the sake of integration. Build it around your user's needs because it will help you onboard more users and expand your revenue stream.

3. Would you like to certify the Collection?

There are two options: your Collection can get certified OR it can live within Ansible Galaxy.

By becoming a Red Hat Ansible Automation Platform Certified Partner you offer your users a trusted, secure, and reliable way of automating your product since certified Ansible content undergoes Red Hat quality evaluation and testing. And most importantly, you assure them the content is maintained and supported.

This will consequently attract larger companies who demand enterprise-grade quality of integrations.

4. Contact XLAB Steampunk

The very last step is getting in contact with us.

Let us know about your plans and desires for Ansible integration and we can help you with your content development, support, and maintenance.

Reach out to us at steampunk@xlab.si.

XLAB
Steampunk

Contact:
+ 386 1 244 77 50
steampunk@xlab.si

www.steampunk.si

Follow us on: