EXECUTIVE
VIEWPOINT

Ricardo Garcia
Principal Portfolio Architect, Red Hat

Ricardo Garcia Cavero is a Principal Portfolio Architect at Red Hat with a SAP Basis background which allows him to create solutions for SAP customers that combine the Red Hat's offering to help them in their journey to the Digital Enterprise.

Jonathan Lake
SAP GTM Specialist Sales UK/Ireland Amazon Web Services

Jonathan Lake is the SAP GTM Specialist at AWS responsible for UK/Ireland. He has more than 20 years of experience in SAP and the associated ecosystem and has worked for companies like HP, DXC and for the past few years at AWS.

Taming SAP migration complexity with Red Hat® OpenShift® Service on AWS

Ricardo Garcia Cavero is a principal technical marketing manage for Red Hat . His background as an SAP basis engineer and solutions architect allows him to help customers in their journey to SAP's Intelligent Enterprise using Red Hat technologies. We spoke with Cavero about how Red Hat can help organizations address the challenges involved in SAP migration.

SAP to SAP S/4HANA migration is often described as a monumental challenge – can you explain the main issues? After launching SAP S/4HANA in 2015 it's revealing that SAP gave its customers a full decade to complete the migration from legacy SAP NetWeaver. For some installations, this process might even take longer. The single biggest reason for this long timescale is the complexity of handling custom code created over many years to add functions to legacy SAP. This is already a problem even without migration. SAP S/4HANA and SAP HANA migration – moving to a completely new architecture – simply multiplies the challenges several fold.

One of the core principles of SAP is the idea of keeping the core clean by minimizing the effect of lots of custom code building up over time. Meeting this has always proved difficult but SAP S/4HANA migration offers a once-in-a-generation opportunity to rationalize custom code. Nevertheless, this requires that organizations first analyze their code, working out what they should migrate and what they should discard. It sounds simple enough but can quickly become a major undertaking.

How does choosing Red Hat® OpenShift® help address this?
Once an organization has decided which custom code it wants to keep, it needs a platform to run it outside of SAP and on which new applications and extensions can be created. Red Hat OpenShift Service on AWS (ROSA) is perfectly designed to address this requirement. Its philosophy is to provide everything an organization needs for cloud-native Kubernetes development.

This works strategically because it allows organizations to integrate their SAP backbone within a multi-cloud architecture. For ease in deploying applications, it supports a wide range of programming languages, with runtimes and tools on hand so that developers can work with their preferred tools.

How does this integrate with other approaches such as SAP's tools?
Naturally, the custom code still needs to connect back to SAP running elsewhere. This is where integration becomes important. It's important to understand that Red Hat is not trying to compete here with SAP's own integration tools, for example the SAP Business Technology Platform (BTP) PaaS. Organizations with a narrow focus on SAP or not embracing cloud native will find that this is a

better approach for them. Instead, integrations can be created by utilizing the portfolio of tools provided by Red Hat's Application Services.

ROSA is a managed service. What type of customers are a good fit for this approach? OpenShift is a good option for running the custom code once it has been carved out of SAP, but it still requires a big team and a lot of specialized skills to manage it that not everyone has, as well as a lot of expensive infrastructure. SAP migration places huge demands on organizations who find that there are simply too many moving parts to look after. For these customers, a completely managed service with support provided jointly by Red Hat and AWS makes sense. It saves time, reduces complexity, and solves a lot of difficult problems in a single managed platform. It is also likely to save money.

In addition, keeping the SAP S/4HANA core clean reduces support complexity and cost, makes version upgrades less, and also reduces the infrastructure footprint for SAP S/4HANA which in turn reduces cost and reduces the carbon footprint.

OpenShift is also a good option for new areas of functionality for innovations beyond the SAP S/4HANA core. The ability to deploy new areas of functionality at scale and speed is a critical area for customers who need to differentiate in the marketplace. Organizations that aren't such a good fit for the ROSA approach would those that have already invested a lot in Red Hat OpenShift and have their own people in place to manage it.

What are the benefits of an API-first approach to SAP integration and how can this be put in place on ROSA? Obviously, there's no point in putting your applications in OpenShift if they can't talk to SAP. This is why integration is so critical. In fact, this can be done in a variety of ways, which is where things can become complex once again.

APIs provide a standardized approach to this problem that will work regardless of the language the applications were written in. But why choose an API-first approach in the context of SAP integration? Because often integrations with SAP were done with an Enterprise Service Bus (ESB) approach. For example, SAP offers ESB tools such as PI/PO, and although thousands of integrations are possible with this approach, it leads to ad-hoc integrations which can't be reused, as well as a huge amount of management complexity. APIs provide significantly more flexibility and reusability. This is why SAP recommends the API-first approach.

Learn more [about SAP and ROSA](#)