WHITE PAPER

INTERRA Information Technologies

Automating XaaS Using vRealize Automation 7.x (vRA)

October 2017

www.interrait.com

Table of Contents

Abstract

XaaS refers to "**Anything as a Service**", revealing how much more can be automated beyond infrastructure, leveraging vRealize Automation and vRealize Orchestrator (vRO).

The need of XaaS

Envision the need to empower clients to utilize distinct services on the vRA portal, where these may not be supported or available, out of the box in vRA. This gap can be plugged by XaaS, utilizing vRA/vRO.

Introduction

Blueprints are an important modeling tool in vRealize Automation capturing the specifications of the service, machine, application, resources (storage, networking etc.), for provisioning and managing their lifecycle.

In vRA, Blueprints are the specification of the resources, which hosts various types of components to be deployed (i.e. Software Component, or prebuilt content) including the lifecycle information.

The following resource elements can be provisioned from a blueprint:

Single/Multi machine blueprints

VM(s) like a vSphere machine, an AWS machine or a Hyper-V machine, with software components, which installs, configures and starts the software during the machine provisioning process, supporting the software lifecycle.

XaaS blueprints

In the vRA world, the blueprints entail the details of the machine being deployed. However, one can design any service with the help of vRO workflows and augment the self-service catalog by publishing it as a blueprint. In vRA, these blueprints are called "*Anything as a Service* (XaaS)" Blueprints.

XaaS are integrations of your vRO workflows that connect with third party systems, and are part of the deployment process.

There are several important components which come together in creating XaaS:

- XaaS Blueprint designer - Advance Service Designer in vRA for publishing of XaaS services to the self-service catalog
- vRA catalog management for organizing these services and publishing them
- vRO for automating the tasks as workflows and extending these to include third party systems.

Challenges before XaaS

As the device-centric computing era is almost gone, companies are focusing and becoming more prominent on Connectivity-centric computing, in which the content is available with cloud. Cloud Service Providers (CSP) are now more focused on automating the configuration, management and provisioning of resources along with the integration of 3rd party services, where a flexible environment is ensured for the efficient consumption and management of services.

Anything-as-a-Service provides access to service and its applications through self-service portals without IT intervention. Having multiple instances of services for testing and development is traditionally a complex operation. By quickly provisioning instances of services like databases, messaging, mail services Etc. one can reduce the time to create, test, deliver, and deploy new applications using the backend service instances.

vRA XaaS provides a robust architecture for automating a XaaS solution, where users quickly deploy and manage applications and compute services, thereby improving business agility and operational efficiency.

vRA XaaS delivers on-demand services to vRA users by creating catalog items and actions. Users can then control who can request these 3rd party services by using entitlements and approvals. It also simplifies the creation and management of services by using resource mapping modelling technique.

Consider a scenario, where one needs to integrate with 3rd party management systems like IPAM, CMDB, DNS etc. and perform the day 2 operations on the same through vRA.

XaaS blueprints are the solution to this.

vRA XaaS Design Fundamentals

Creating Custom Resources

Custom Resources are the assets or resources provisioned by the XaaS services and managed, e.g. VM. These can also be used for post-provisioning operations with the help of resource actions.

Managing Resources with Resource Actions and Resource Mappings

The resource actions can be created to perform activities on the provisioned resources. These post provisioning operations can manage the already provisioned resources. E.g. Reconfigure VM.

Resource mappings can also be created between the vRealize Automation catalog resource and the vRO inventory type to manage resources provisioned outside of XaaS.

Creating XaaS Blueprints

We can then design a XaaS blueprint, which serves as the UI for our Orchestrator workflow. After assigning the output to a custom resource (i.e. VM provisioned), it can be published to the service catalog on vRA.

Below flow diagram depicts the activities for creating vRA XaaS Blueprint:

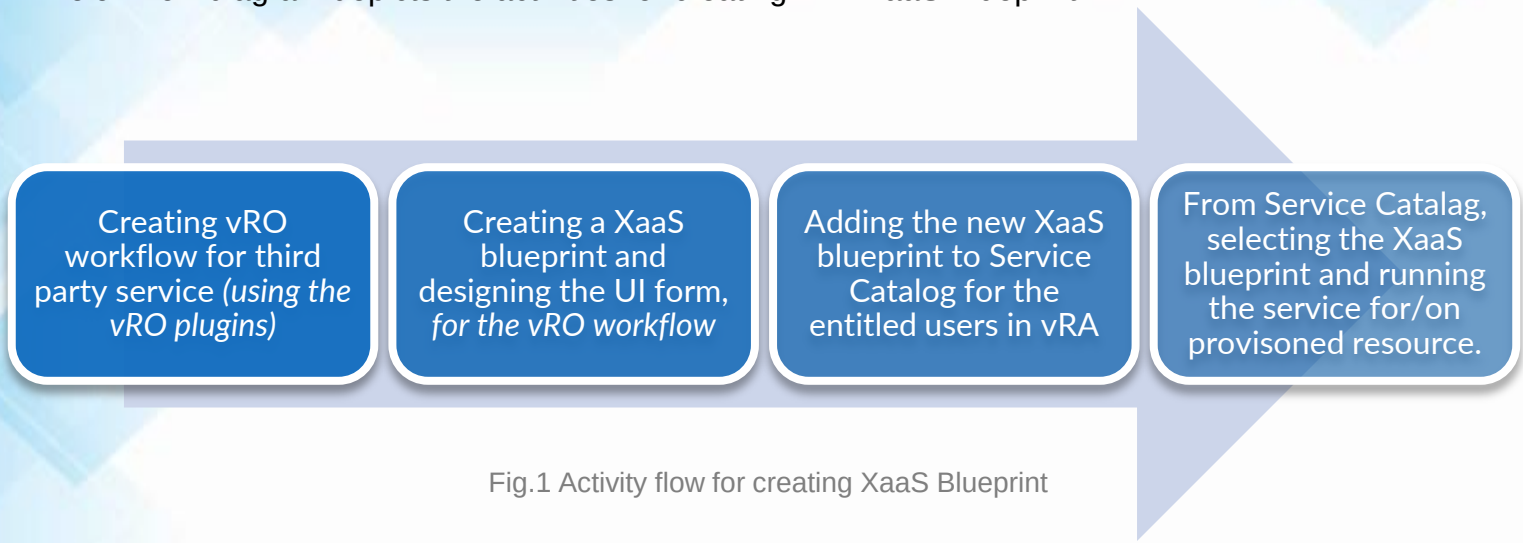


Fig.1 Activity flow for creating XaaS Blueprint

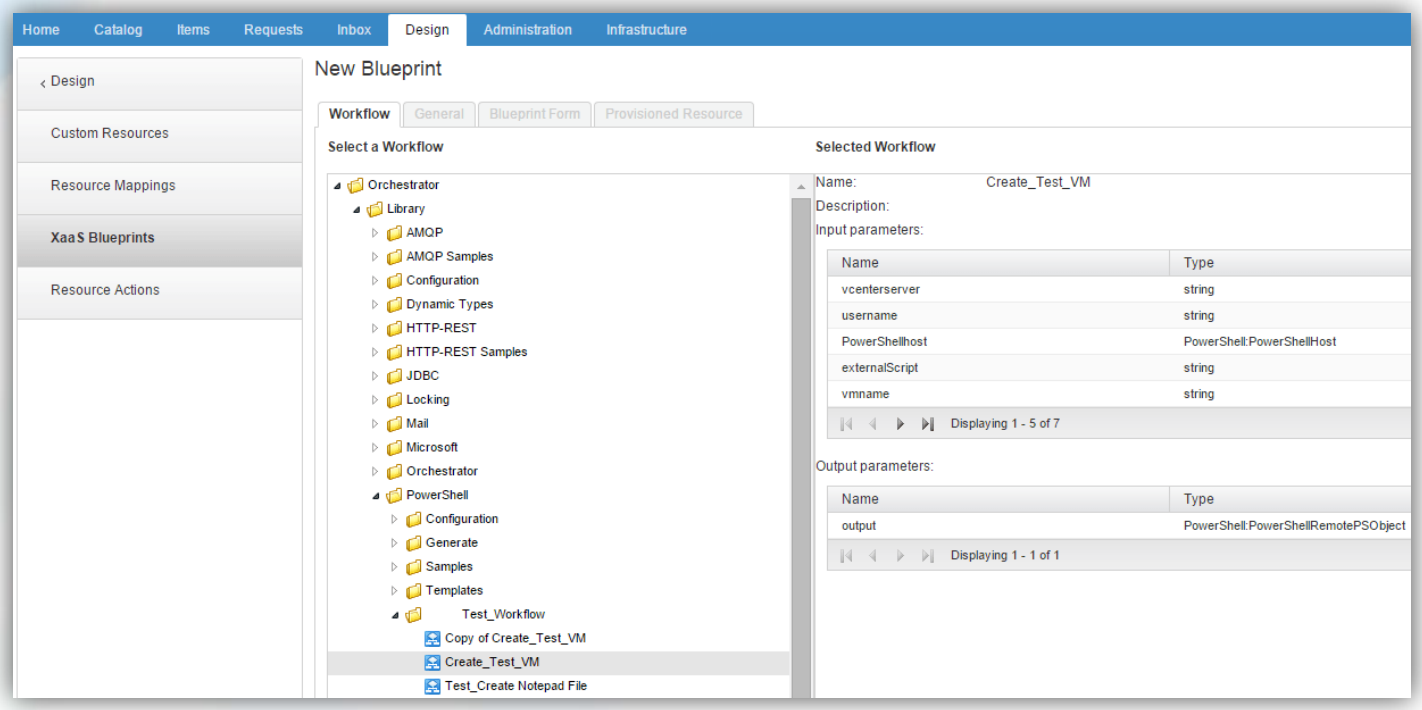


Fig.2 Creating a XaaS Blueprint in vRA from vRO workflow

XaaS helps in integrating the third party solution in vRA and providing them as a service through the self-service portal. Several vRO plug-ins exist for vRA, through which many products can be integrated in vRA and invoked from vRA portal. Some existing plug-ins include AWS, Puppet, Service Now, Storage plugins from NetApp etc.

XaaS can be the perfect solution in many scenarios listed below:

- Third party integrations like AWS, Puppet, Docker, Etc.
- Day 2 Operations like User administration such renaming the VM's provisioned, adding the users/groups to VM's for authentication, reconfiguring the provisioned resources, Etc.
- Automating notifications/warnings to users and admins through email.
- Providing Database as a Service.
- NSX Integrations.
- Backups and Storage.

Configuring vRA for XaaS

In order to create XaaS, we will need to configure vRA, which involves following tasks:

Assigning the appropriate roles

We need to have the XaaS Architect role to create and modify XaaS.

Configuring the vRO Endpoint

Although 7.x comes with inbuilt vRO and default vRO endpoint configured, we can also configure external vRO endpoint for the vRA.

In Order to configure vRO server, go to **Server Configuration** menu, and select to whether to use the default Orchestrator server or an external one.

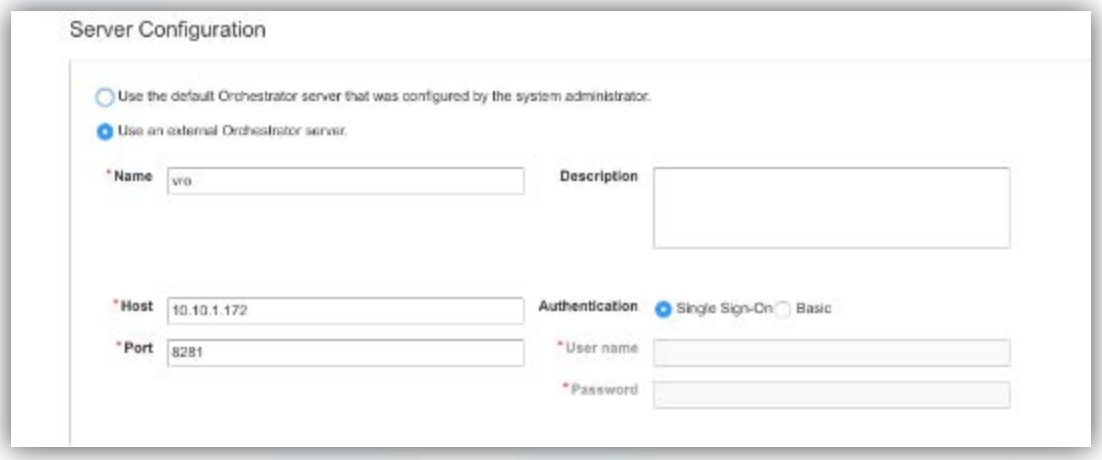


Fig. 3 configuring vRO Server in vRA

Configuring the Orchestrator plug-ins in vRO

Configuration of the endpoints for the vRO is required. It provides endpoints for many plug-ins:

Active Directory, HTTP-REST, PowerShell, SOAP, vCenter Server, Etc.

XaaS Use Cases

There are many use cases for XaaS. Let's take a closer look at some of these.

Use Case 1: XAAS Blueprint - Database as a Service (DBaaS)

Automation of Database as a Service, a service model, which provides an instance of a database, with access details, irrespective of where the database server is running. With vRA XaaS, DBaaS will be available as a catalog item on vRA service catalog, and users can directly request the item, to get the DB instance.

Designing the DBaaS

Using vRA we have implemented DBaaS, for different Databases like **MYSQL, MSSQL, Oracle** Database, etc. We have also implemented the Day2 Operations and Lifecycle management for NOSQL database like **MongoDB**.

vRA gives us the ability to provision VMs with Database service like MYSQL, MSSQL. We can perform day-2 operations by creating XaaS, which will automate the below tasks:

1. XaaS Blueprint for Creating the Database (MYSQL/MSSQL) instance.
2. XaaS Resource Action/Day 2 operation for Creating Schema with new DB User on the selected instance.

Implementing DBaaS

Developing a Blueprint

A Blueprint with a VM (vSphere/AWS) is designed along with the software components, which will install and configure the database i.e. MSSQL, MYSQL or Oracle DB 12c on a machine.

The Software Component takes the input parameters like Database Admin Username, Password, Port number etc. It can be deployed from a vRA service catalog, where users will provide the input details.

Creating a vRO Workflow

- vRO Workflows were created for Registration and Deregistration of Database Server to be used as a DBaaS.
- Using inbuilt vRO library plugins, a workflow is created for Registration and Deregistration of DB.
- It will accept the input parameters like Resource Item – VM, Username and Password for authentication.

- Another workflow for creating a User Schema with new user access is created, which will take inputs like Database instance, new schema name, schema username and password.

Creating XAAS Blueprints and Resource Actions for DBaaS

XAAS Blueprint created from above workflow will take below user inputs:

1. Provisioned Database Instance
2. Username/Password for new DB User.

There will be two Resource Actions, which will be required for pre and post operations of XAAS BP.

1. **Register Database Instance**, which will register the provisioned Instances of Databases to be used for DBaaS.
2. **De-Register Database Instance**, which will deregister the provisioned Instances of Databases.

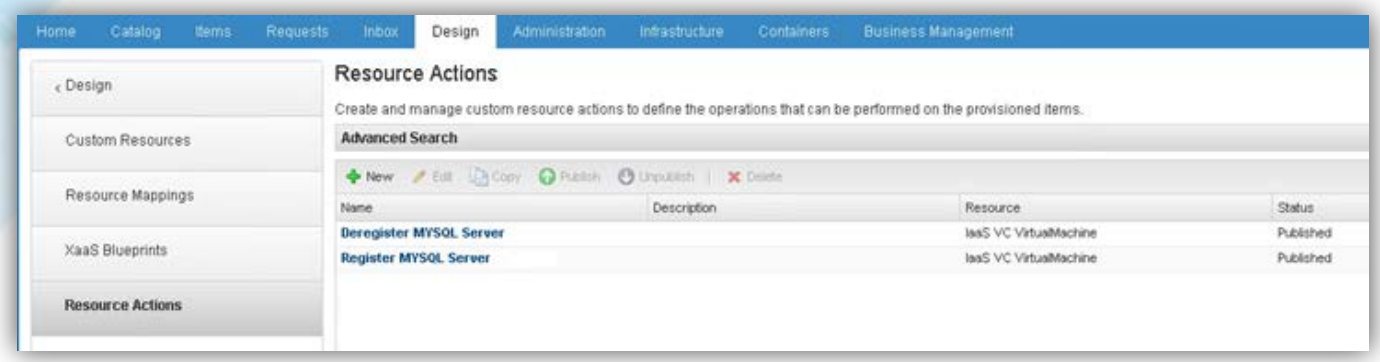


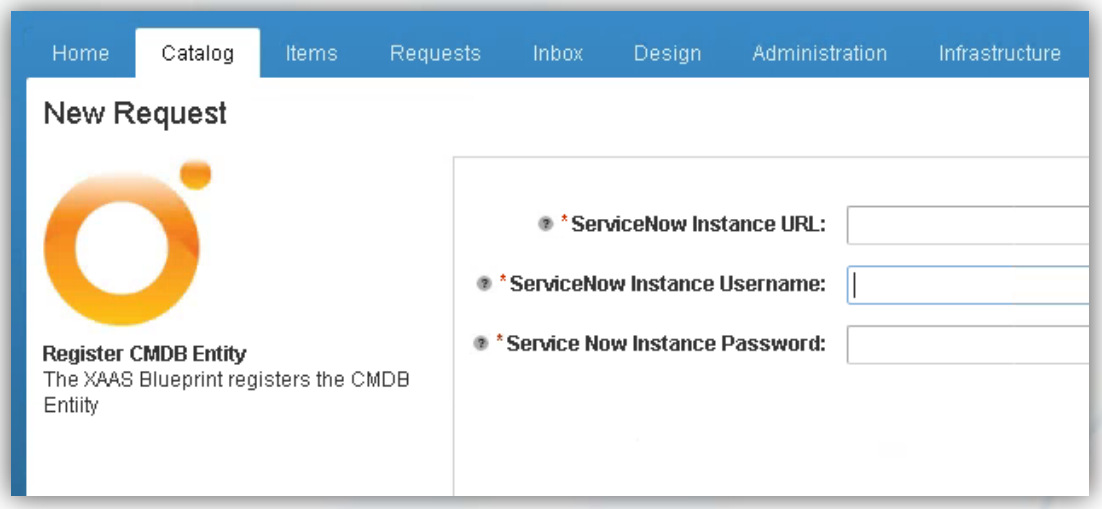
Fig. 4 Resource Action - Database as a Service for MYSQL

Use Case 2: XAAS Blueprint – Service Now Integration

Designing

With vRA, we have automated Incident generation, Insert, Update and Delete Operations on Service Now CMDB records for every provision done through vRA Blueprint. Day 2 operations have been created for Adding/Removing the Service Now-CMDB entity, Inserting and updating the CI record on registered/added CMDB entity.

Implementation



VRA XAAS BP and its **Workflows** for Registering and De-Registering the Service Now CMDB Entity.

Fig. 5 XaaS Blueprint for Service Now CMDB Instance Registration

Inputs– Service Now Instance, Service Now Instance Username, Service Now Instance Password, CMDB Entity (CI) on Service Now Instance

Use Case 3: Resource Actions for AWS

We have developed a resource action for interacting with AWS. This required automating the provisioning and configuration of an AWS EC2 Instance. Day 2 operations were created, for Adding, Updating, and Removing AWS accounts, Creating VPC's and Subnets etc.

Designing Automation for VPC/Subnet creation on AWS Instance

We used default library plugins from vRO as well as below items to automate the tasks using vRA Portal

- AWS Account/Instance with Access/Secret Keys.
- VRO-AWS Plugin for vRO

Implementation

Following things are configured and created **XAAS BP's for AWS on vRA:**

1. XaaS for Creating Subnet

It will create a subnet in the VPC. Inputs like VPC, Subnet Name, CIDR block for subnet, Availability Zones, etc are used for manual intervention.

A workflow on vRO is created and linked, which will take all the inputs shown above, along with the AWS Instance.

2. XaaS for Creating VPC

This XaaS BP will create a VPC , using the CIDR Block and Tenancy.

It takes following inputs like AWS Client, New VPC Name, etc.
A workflow is created and linked, which will take all the inputs shown above, along with the AWS Instance.

Fig. 6 XaaS Blueprint – Creating Subnet on AWS Instance

Conclusion

Using XaaS we are utilizing only one catalog item, instead of navigating through various third party tools or components, while managing multiple tools and objects effectively.

One of the other qualities of the XaaS is, its ability, to build dynamic request forms for integrated third party functions. The vRA clubbed with vRO help in input validation and have helped in implementation of several use cases.

We have now looked at the standard and the specialized methods for XaaS, with which it is possible to accomplish as well as optimize the business process automations.

References

https://docs.vmware.com/en/vRealize-Automation/7.0/com.vmware.vrealize.automation.doc/GUID-5AD4507B-57FC-44E1-9C5B-80818D2A7847.html