

Referenz für benutzerdefinierte Eigenschaften

28. Dezember 2020

vRealize Automation 7.6

Die aktuellste technische Dokumentation finden Sie auf der VMware-Website unter:

<https://docs.vmware.com/de/>

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

VMware Global, Inc.
Zweigniederlassung Deutschland
Willy-Brandt-Platz 2
81829 München
Germany
Tel.: +49 (0) 89 3706 17 000
Fax: +49 (0) 89 3706 17 333
www.vmware.com/de

Copyright © 2008-2020 VMware, Inc. Alle Rechte vorbehalten. [Urheberrechts- und Markenhinweise](#).

Inhalt

Referenz für benutzerdefinierte Eigenschaften 5

Aktualisierte Informationen 6

1 Benutzerdefinierte Eigenschaften und das Eigenschaftenwörterbuch 7

Verwenden benutzerdefinierter Eigenschaften 7

Erstellen und Hinzufügen von benutzerdefinierten Eigenschaften und Eigenschaftsgruppen 8

Verwenden von Eigenschaften bei der Maschinenbereitstellung 9

Grundlegende Informationen zur Rangfolge benutzerdefinierter Eigenschaften 10

Benutzerdefinierte Eigenschaften gruppiert nach der Funktion 13

Benutzerdefinierte Eigenschaften für Bereitstellungen 15

Benutzerdefinierte Eigenschaften zum Benennen und Analysieren von Bereitstellungen 17

Benutzerdefinierte Eigenschaften für OpenStack-Endpoints 19

Benutzerdefinierte Eigenschaften für Klon-Blueprints 21

Benutzerdefinierte Eigenschaften für verknüpfte Klon-Blueprints 26

Benutzerdefinierte Eigenschaften für FlexClone-Blueprints 29

Benutzerdefinierte Eigenschaften für einfache Workflow-Blueprints 32

Benutzerdefinierte Eigenschaften für Linux Kickstart-Blueprints 34

Benutzerdefinierte Eigenschaften für SCCM-Blueprints 36

Benutzerdefinierte Eigenschaften für WIM-Blueprints 38

Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Blueprints 41

Benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit 46

Benutzerdefinierte Eigenschaften und Eigenschaftsgruppen für Container 58

Benutzerdefinierte Eigenschaften für die PXE-Bereitstellung 60

Benutzerdefinierte Eigenschaften für OVF-Import 63

Benutzerdefinierte Eigenschaften für den vRealize Automation-Gast-Agent 64

Benutzerdefinierte Eigenschaften für die Integration von BMC BladeLogic Configuration Manager 69

Benutzerdefinierte Eigenschaften für die Integration von HP Server Automation 71

Benutzerdefinierte Eigenschaften gruppiert nach dem Namen 74

Benutzerdefinierte Eigenschaften mit Unterstrich (_) 75

Benutzerdefinierte Eigenschaften A 77

Benutzerdefinierte Eigenschaften B 79

Benutzerdefinierte Eigenschaften C 80

Benutzerdefinierte Eigenschaften E 82

Benutzerdefinierte Eigenschaften H 85

Benutzerdefinierte Eigenschaften I 85

Benutzerdefinierte Eigenschaften L	87
Benutzerdefinierte Eigenschaften M	88
Benutzerdefinierte Eigenschaften N	88
Benutzerdefinierte Eigenschaften O	92
Benutzerdefinierte Eigenschaften P	93
Benutzerdefinierte Eigenschaften R	93
Benutzerdefinierte Eigenschaften S	94
Benutzerdefinierte Eigenschaften V	101
Benutzerdefinierte Eigenschaften X	131
Verwenden des Eigenschaftenwörterbuchs	131
Verwenden von Eigenschaftsdefinitionen	132
Verwenden von Eigenschaftsgruppen	158
Definieren der Einstellungen für Komponentenprofile	160
Konfigurieren von Komponentenprofil-Image-Einstellungen für Katalogbereitstellungen	160
Konfigurieren von Größeneinstellungen des Komponentenprofils für Katalogbereitstellungen	162

Referenz für benutzerdefinierte Eigenschaften

Referenz für benutzerdefinierte Eigenschaften stellt Informationen über die benutzerdefinierten Eigenschaften und deren Verwendung bereit, die bei der Verwendung von vRealize Automation verfügbar sind.

Diese Dokumentation ist zur Verwendung mit der vRealize Automation-Produktdokumentation gedacht, die über die vRealize Automation-Produktdokumentation unter <https://docs.vmware.com/de/vRealize-Automation/index.html> zur Verfügung steht.

Zielgruppe

Diese Information ist für IaaS-Administratoren, Fabric-Administratoren und Business-Gruppen-Manager von vRealize Automation gedacht. Dieser Inhalt wurde für erfahrene Windows- oder Linux-Systemadministratoren geschrieben, die mit der Virtualisierungstechnologie und den in *Grundlagen und Konzepte* beschriebenen Standardkonzepten vertraut sind.

Aktualisierte Informationen

Referenz für benutzerdefinierte Eigenschaften wird mit jeder Produktversion oder bei Bedarf aktualisiert.

Diese Tabelle enthält den Update-Verlauf für *Referenz für benutzerdefinierte Eigenschaften*.

Revision	Beschreibung
14. Februar 2019	Kleinere Updates.
9. September 2019	Kleinere Updates.
18. Juli 2019	VirtualMachine.Software.Execute, VirtualMachine.Admin.UseGuestAgent, VirtualMachine.Customize.WaitComplete und VirtualMachine.Admin.CustomizeGuestOSDelay in Benutzerdefinierte Eigenschaften V wurden aktualisiert.
1. März 2019	Benutzerdefinierte Eigenschaften V wurde aktualisiert.
25. Januar 2019	Benutzerdefinierte Eigenschaften mit Unterstrich (_) wurde aktualisiert.
13. November 2018	Kleinere Updates.
4. Oktober 2018	Kleinere Updates.
20. September 2018	Erstversion.

Benutzerdefinierte Eigenschaften und das Eigenschaftswörterbuch

1

Sie können die im Lieferumfang enthaltenen benutzerdefinierten vRealize Automation-Eigenschaften verwenden, um verschiedene Aspekte der Maschinenbereitstellung zu steuern. Sie können auch das Eigenschaftswörterbuch verwenden, um neue Eigenschaftsdefinitionen und -gruppen zu erstellen, die auf Ihre Bedürfnisse zugeschnitten sind.

Sie können Eigenschaften verwenden, um Werte hinzuzufügen bzw. vorhandene oder Standardwerte zu überschreiben und Netzwerk-, Plattform- und Gast-Agenteneinstellungen sowie viele andere bereitstellungsbezogene Parameter zu konfigurieren.

Dieses Kapitel enthält die folgenden Themen:

- [Verwenden benutzerdefinierter Eigenschaften](#)
- [Benutzerdefinierte Eigenschaften gruppiert nach der Funktion](#)
- [Benutzerdefinierte Eigenschaften gruppiert nach dem Namen](#)
- [Verwenden des Eigenschaftswörterbuchs](#)
- [Definieren der Einstellungen für Komponentenprofile](#)

Verwenden benutzerdefinierter Eigenschaften

Mithilfe benutzerdefinierter Eigenschaften von vRealize Automation können Sie Werte hinzufügen oder vorhandene Werte bzw. Standardwerte überschreiben, um Einstellungen für Netzwerk, Plattform, Gast-Agent und viele andere Bereitstellungsparameter zu konfigurieren.

Einige Eigenschaften werden durch Standardeinstellungen bestimmt, die Sie für alle Maschinen angeben müssen. Beispielsweise sind Werte für Arbeitsspeicher und Festplattengröße für alle Blueprints erforderlich. Sie können zusätzliche Eigenschaften einzeln oder in Form von Eigenschaftsgruppen in Blueprints und Reservierungen angeben.

Wenn Sie einem Blueprint oder einer Eigenschaftsgruppe eine Eigenschaft hinzufügen, können Sie sie als erforderliche Eigenschaft kennzeichnen. Ist eine Eigenschaft als erforderlich festgelegt, muss der Benutzer einen Wert für diese Eigenschaft angeben, wenn er eine Maschine anfordert, wie in den folgenden Beispielen veranschaulicht.

- Anfordern von Informationen zu mehreren Festplatten, die den zugeteilten Speicher der Maschine gemeinsam nutzen.

- Anfordern von Informationen zu Benutzern oder Gruppen, die einer lokalen Gruppe auf der Maschine hinzugefügt werden sollen.
- Anfordern des Hostnamens der Maschine.

Der Windows-Gast-Agent zeichnet Eigenschaftswerte auf der bereitgestellten Maschine in der Datei %SystemDrive%\VRMGuestAgent\site\workitem.xml auf.

Der Linux-Gast-Agent zeichnet Eigenschaftswerte auf der bereitgestellten Maschine in der Datei /usr/share/gugent/site/workitem.xml auf.

Erstellen und Hinzufügen von benutzerdefinierten Eigenschaften und Eigenschaftsgruppen

Sie können zur Steuerung von Maschinenbereitstellungen benutzerdefinierte Eigenschaften verwenden. Sie können bereitgestellte benutzerdefinierte Eigenschaften hinzufügen, aber auch eigene Eigenschaften und Eigenschaftsgruppen erstellen und hinzufügen.

Sie können Eigenschaften und Eigenschaftsgruppen zu allgemeinen Blueprints, Komponenten in einem Blueprint, Reservierungen und anderen vRealize Automation-Elementen hinzufügen, einschließlich einiger Endpoint-Typen. Sie können auch neue benutzerdefinierte Eigenschaften und Eigenschaftsgruppen erstellen.

Sie können Eigenschaften und Eigenschaftsgruppen beim Erstellen des Blueprints oder später, wenn sich der Blueprint im Status „Entwurf“ oder „Veröffentlicht“ befindet, hinzufügen. Alternativ können Sie benutzerdefinierte Eigenschaften und Eigenschaftsgruppen zu einzelnen Komponenten im Blueprint hinzufügen.

Benutzerdefinierte Eigenschaften auf Blueprint-Ebene haben Vorrang vor den auf der Ebene der Komponente konfigurierten benutzerdefinierten Eigenschaften. Informationen zur Rangfolge von benutzerdefinierten Eigenschaften finden Sie unter [Grundlegende Informationen zur Rangfolge benutzerdefinierter Eigenschaften](#).

Eigenschaften auf Blueprint-Ebene können Sie mithilfe der Seite „Blueprint-Eigenschaften“ bearbeiten.

Bei einer benutzerdefinierten Eigenschaft kann es optional erforderlich sein, dass der Benutzer beim Erstellen einer Maschinenanforderung eine bestimmte Eigenschaft angibt.

- Bei Namen und Werten benutzerdefinierter Eigenschaften wird in der Regel die Groß-/Kleinschreibung beachtet. Beispielsweise werden die benutzerdefinierte Eigenschaft `hostname` und die benutzerdefinierte Eigenschaft `HOSTNAME` als unterschiedliche benutzerdefinierte Eigenschaften betrachtet.
- Namen benutzerdefinierter Eigenschaften dürfen keine Leerzeichen enthalten. Verwenden Sie bei der Erstellung und Verwendung von benutzerdefinierten Eigenschaften keine Leerzeichen im Eigenschaftsnamen.
- Einige Namen benutzerdefinierter Eigenschaften sind reserviert und können nicht als Namen verwendet werden, wenn Sie neue benutzerdefinierte Eigenschaften erstellen. Die Eigenschaftsnamen `Encrypted` und `encrypted` sind beispielsweise reserviert.

Weitere Informationen zum Erstellen neuer benutzerdefinierter Eigenschaften und Eigenschaftsgruppen finden Sie unter [Verwenden des Eigenschaftenwörterbuchs](#).

Verwenden von Eigenschaften bei der Maschinenbereitstellung

Benutzerdefinierte Eigenschaften sind von vRealize Automation bereitgestellte Eigenschaften. Sie können auch eigene Eigenschaften definieren. Eigenschaften sind Name/Wert-Paare, mit denen Attribute einer Maschine festgelegt oder Standardspezifikationen überschrieben werden.

Sie können benutzerdefinierte Eigenschaften verwenden, um verschiedene Bereitstellungsmethoden, Maschinentypen und Maschinenoptionen zu steuern, wie in den folgenden Beispielen gezeigt:

- Geben Sie einen bestimmten Gastbetriebssystemtyp an.
- Aktivieren Sie die WIM-basierte Bereitstellung, bei der ein Windows Imaging File Format (WIM)-Image einer Referenzmaschine für die Bereitstellung neuer Maschinen verwendet wird.
- Passen Sie das Verhalten des Remotedesktopprotokolls bei der Verbindungsherstellung mit einer Maschine an.
- Registrieren Sie eine virtuelle Maschine mit einem XenDesktop Desktop Delivery Controller (DDC)-Server.
- Passen Sie die Systemspezifikationen einer virtuellen Maschine an, wie beispielsweise das Hinzufügen mehrerer Festplattenlaufwerke.
- Passen Sie das Gastbetriebssystem für eine Maschine an, beispielsweise durch Hinzufügen angegebener Benutzer in ausgewählten lokalen Gruppen.
- Geben Sie Netzwerk- und Sicherheitseinstellungen an.
- Fügen Sie zusätzliche Steuerungsoptionen (wie z. B. Dropdown-Menüs) hinzu, um dem Benutzer zum Zeitpunkt der Anforderung Eingabe- und Auswahloptionen zur Verfügung zu stellen.

Wenn Sie eine Eigenschaft zu einem Blueprint, einer Reservierung oder einem anderen Formular hinzufügen, können Sie angeben, ob die Eigenschaft verschlüsselt werden soll und ob der Benutzer bei der Bereitstellung zur Eingabe eines Werts aufgefordert werden soll. Diese Optionen können bei der Bereitstellung nicht überschrieben werden.

Ein Beispiel für das Hinzufügen von zusätzlichen Steuerungsoptionen zwecks dynamischer Festlegung einer benutzerdefinierten Eigenschaft basierend auf einer Benutzerauswahl aus einer Liste mit vordefinierten Optionen finden Sie im Blogbeitrag [Adding a Network Selection Drop-Down in vRA 7](#) (Hinzufügen einer Dropdown-Liste für die Netzwerkauswahl in vRA 7).

Eine in einem Blueprint angegebene Eigenschaft überschreibt dieselbe Eigenschaft, die in einer Eigenschaftsgruppe angegeben ist. Auf diese Weise kann ein Blueprint die meisten Eigenschaften in einer Eigenschaftsgruppe verwenden, während es für bestimmte Situationen Abweichungen von der Eigenschaftsgruppe gibt. Beispielsweise kann ein Blueprint, der eine Eigenschaftsgruppe für die standardmäßige Entwickler-Workstation integriert, die Einstellungen für Englisch (USA) in der Gruppe durch die Einstellungen für Englisch (GB) überschreiben.

Sie können Eigenschaften in Reservierungen und Business-Gruppen auf zahlreiche Maschinen anwenden. Ihre Verwendung ist in der Regel auf Zwecke beschränkt, die sich auf die jeweiligen Quellen beziehen, wie z. B. Ressourcenverwaltung. Die Merkmale der bereitzustellenden Maschine werden im Allgemeinen durch Hinzufügen von Eigenschaften zu Blueprints und Eigenschaftsgruppen angegeben.

Grundlegende Informationen zur Rangfolge benutzerdefinierter Eigenschaften

Ordnungsgemäß autorisierte Benutzer können benutzerdefinierte Eigenschaften für Blueprints, Endpoints, Business-Gruppen und Reservierungen angeben. Wenn dieselbe Eigenschaft in mehreren Quellen vorhanden ist, hält sich vRealize Automation beim Anwenden von Eigenschaften auf eine Maschine an eine bestimmte Rangfolge.

Die folgenden Elemente können um benutzerdefinierte Eigenschaften ergänzt werden, die auf bereitgestellte Maschinen angewendet werden:

- Eine Reservierung, um die benutzerdefinierten Eigenschaften auf alle Maschinen anzuwenden, die über diese Reservierung bereitgestellt werden.
- Eine Business-Gruppe, um die benutzerdefinierten Eigenschaften auf die Bereitstellung und alle Maschinen anzuwenden, die durch Business-Gruppenmitglieder bereitgestellt werden.
- Ein Blueprint, um die benutzerdefinierten Eigenschaften auf alle Maschinen anzuwenden, die über diesen Blueprint bereitgestellt werden.
- Eigenschaftsgruppen, die in einem Blueprint enthalten sein können, um alle benutzerdefinierten Eigenschaften in der Gruppe auf alle Maschinen anzuwenden, die über den Blueprint bereitgestellt werden.

Ein Blueprint kann eine oder mehrere Eigenschaftsgruppen enthalten.

- Eine Maschinenanforderung, um die benutzerdefinierten Eigenschaften auf die bereitgestellte Maschine anzuwenden.
- Eine Genehmigungsrichtlinie, wenn die Unterstützung für die erweiterte Genehmigung aktiviert ist, damit Genehmiger Werte für die Maschine, die genehmigt wird, eingeben müssen.

In der folgenden Liste wird die Rangfolge für benutzerdefinierte Eigenschaften aufgeführt. Eigenschaftswerte, die in einer Quelle weiter unten in der Liste angezeigt werden und Werte für dieselbe Eigenschaft überschreiben, die in Quellen weiter oben in der Liste angegeben sind.

Wenn ein Konflikt zwischen einem von vRealize Automation bereitgestellten benutzerdefinierten Eigenschaftsnamen und einem benutzerdefinierten Eigenschaftsnamen besteht, hat der von vRealize Automation bereitgestellte benutzerdefinierte Eigenschaftsname Vorrang.

- 1 Eigenschaftsgruppe
- 2 Blueprint
- 3 Business-Gruppe

- 4 Computing-Ressource
- 5 Reservierungen
- 6 Endpoint
- 7 Laufzeit

Benutzerdefinierte Eigenschaften von Eigenschaftsgruppen, Blueprints und Business-Gruppen werden zum Zeitpunkt der Anforderung zugewiesen, während die Eigenschaften von Computing-Ressourcen, Reservierungen und Endpoints während der Bereitstellung zugewiesen werden.

Diese Rangfolge wird wie folgt weiter aufgeschlüsselt:

- 1 Benutzerdefinierte Eigenschaften und Gruppen auf der Ebene des gesamten Blueprints
- 2 Benutzerdefinierte Eigenschaften und Gruppen auf der Komponentenebene
- 3 Benutzerdefinierte Eigenschaften für die Business-Gruppe
- 4 Benutzerdefinierte Eigenschaften für die Computing-Ressource
- 5 Benutzerdefinierte Eigenschaften für die Reservierung
- 6 Benutzerdefinierte Eigenschaften für den Endpoint
- 7 Benutzerdefinierte Eigenschaften auf der Anforderungsebene der verschachtelten Blueprints
- 8 Benutzerdefinierte Eigenschaften auf der Ebene der Komponentenanforderung

In den meisten Fällen hat eine Laufzeiteigenschaft Vorrang vor anderen Eigenschaften. Eine Laufzeiteigenschaft erfüllt die folgenden Bedingungen:

- Die benutzerdefinierte Eigenschaftsoption, die den Benutzer zur Eingabe auffordert, ist ausgewählt und gibt an, dass der Benutzer einen Wert für die Eigenschaft festlegen muss, wenn Maschinenbereitstellung angefordert wird.
- Ein Business-Gruppenmanager fordert Maschinenbereitstellung an und die Eigenschaft wird in der Liste mit den benutzerdefinierten Eigenschaften auf der Seite zur Bestätigung der Maschinenanforderung angezeigt.

Es gibt Ausnahmen zu den Vorrangregeln. Sie fügen beispielsweise die benutzerdefinierte Eigenschaft `VMware.VirtualCenter.Folder` zu einer Business-Gruppe hinzu, stellen einen Eigenschaftswert bereit und wählen die Option zum Anzeigen der Eigenschaft in der Anforderung nicht aus. Sie fügen dieselbe benutzerdefinierte Eigenschaft zu einem Blueprint hinzu und geben an, dass die Eigenschaft in der Anforderung angezeigt werden soll. Wenn die festgelegten Benutzer Bereitstellung aus dem Katalog anfordern, wird die Eigenschaft nicht im Kataloganforderungsformular angezeigt, da sich die Eigenschaft auf Reservierungsinformationen bezieht, die erst nach dem Beginn der Bereitstellung und nicht bei Bereitstellungsanforderung verfügbar sind.

Benutzerdefinierte Eigenschaftstypen

Sie können für geklonte Maschinen externe und aktualisierte vRealize Automation-Eigenschaftstypen verwenden. Sie dürfen für geklonte Maschinen keine internen oder schreibgeschützten Eigenschaftstypen verwenden.

Die folgenden benutzerdefinierten vRealize Automation-Eigenschaftstypen sind verfügbar.

- Intern

Der angegebene Wert wird nur in der Datenbank verwaltet. Beispielsweise wird die E-Mail-Adresse des Managers, der die Maschinenanfrage genehmigt hat, in der `VirtualMachine.Admin.Approver`-Eigenschaft aufgezeichnet, die Eigenschaft hat jedoch keinen Einfluss auf die Maschine.

- Schreibgeschützt

Der angegebene Wert wird auf der Maschine implementiert und kann nicht geändert werden. `VirtualMachine.Admin.UUID` gibt beispielsweise die UUID der Maschine an, die nicht geändert werden kann.

- Extern

Die externen Eigenschaften einer Maschine werden festgelegt, wenn die Virtualisierungsplattform die Maschine erstellt oder während der WinPE-Phase des Erstellungsvorgangs. Zum Festlegen dieser Eigenschaften müssen ihre Werte dem Proxy-Agent übermittelt werden. Dieser gibt sie an die Virtualisierungsplattform oder den Gast-Agent weiter, wodurch sie in der WinPE-Phase implementiert werden.

Der angegebene Wert wird auf der Maschine implementiert, aber nie aktualisiert. Wenn beispielsweise die Eigenschaft `VirtualMachine.Admin.AddOwnerToAdmins` auf „true“ festgelegt ist, wird der Besitzer der Maschine zur lokalen Administratorgruppe hinzugefügt. Wenn der Besitzer später aus dieser Gruppe entfernt wird, wird die Eigenschaft nicht auf „false“ aktualisiert.

- Aktualisiert

Der angegebene Wert wird auf der Maschine implementiert und wird mithilfe von Datenerfassung aktualisiert. Wenn beispielsweise die Computing-Ressource einer Maschine geändert wird, aktualisiert ein Proxy-Agent den Wert für die Eigenschaft `VirtualMachine.Admin.Hostname` der Maschine.

Interne und schreibgeschützte Eigenschaftstypen legen Attribute fest, die von der Vorlage bestimmt werden.

Sie können das vRealize Automation-Maschinenmenü verwenden, um alle reservierten benutzerdefinierten Eigenschaften zu ändern, mit Ausnahme der schreibgeschützten Eigenschaften `VirtualMachine.Admin.AgentID`, `VirtualMachine.Admin.UUID` und `VirtualMachine.Admin.Name`.

Benutzerdefinierte Eigenschaften gruppiert nach der Funktion

Mithilfe von benutzerdefinierten Eigenschaften können Sie zusätzliche vRealize Automation-Steurelemente bereitstellen.

Benutzerdefinierte Eigenschaften wurden hier nach der Funktion gruppiert. Nach dem Namen gruppierte benutzerdefinierte Eigenschaften finden Sie unter [Benutzerdefinierte Eigenschaften gruppiert nach dem Namen](#).

- [Benutzerdefinierte Eigenschaften für Bereitstellungen](#)

vRealize Automation bietet mehrere benutzerdefinierte Eigenschaften, die für die meisten Bereitstellungen gelten.

- [Benutzerdefinierte Eigenschaften zum Benennen und Analysieren von Bereitstellungen](#)

Wenn die Bereitstellung fehlschlägt, führt vRealize Automation ein Rollback aller im Katalogelement enthaltenen Ressourcen durch. Bei Bereitstellungen mit mehreren Komponenten können Sie eine benutzerdefinierte Eigenschaft verwenden, um die Standardeigenschaften zu überschreiben und um Informationen zum Debugging des Fehlers zu erhalten. Diese Eigenschaften werden idealerweise auf den gesamten Blueprint angewendet.

- [Benutzerdefinierte Eigenschaften für OpenStack-Endpoints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, die Sie bei der Konfiguration von OpenStack-Endpoints in vRealize Automation verwenden können.

- [Benutzerdefinierte Eigenschaften für Klon-Blueprints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für Klon-Blueprints bereitstellen können.

- [Benutzerdefinierte Eigenschaften für verknüpfte Klon-Blueprints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für verknüpfte Klon-Blueprints bereitstellen können.

- [Benutzerdefinierte Eigenschaften für FlexClone-Blueprints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für FlexClone-Blueprints bereitstellen können.

- [Benutzerdefinierte Eigenschaften für einfache Workflow-Blueprints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für einfache Workflow-Blueprints bereitstellen können.

- [Benutzerdefinierte Eigenschaften für Linux Kickstart-Blueprints](#)

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für Linux Kickstart-Blueprints bereitstellen können.

- **Benutzerdefinierte Eigenschaften für SCCM-Blueprints**

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für SCCM-Blueprints bereitstellen können.

- **Benutzerdefinierte Eigenschaften für WIM-Blueprints**

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für WIM-Blueprints bereitstellen können.

- **Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Blueprints**

Sie können bestimmte benutzerdefinierte Eigenschaften zu einer vCloud Air- oder vCloud Director-Maschinenkomponentendefinition in einem Blueprint hinzufügen.

- **Benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit**

Mit den benutzerdefinierten Eigenschaften von vRealize Automation für Netzwerke wird die Konfiguration für ein bestimmtes Netzwerkgerät auf einer Maschine festgelegt.

- **Benutzerdefinierte Eigenschaften und Eigenschaftsgruppen für Container**

Sie können einer Containerkomponente in einem vRealize Automation-Blueprint vordefinierte Eigenschaftsgruppen hinzufügen. Wenn Maschinen, die diese Eigenschaften enthalten, mithilfe eines Blueprints bereitgestellt werden, wird die bereitgestellte Maschine als Docker Container-Hostmaschine registriert.

- **Benutzerdefinierte Eigenschaften für die PXE-Bereitstellung**

PXE ist die einzige Bereitstellungsmethode, die für Cisco UCS Manager unterstützt wird. Mit dem Network Bootstrap-Programm können Sie zusammen mit benutzerdefinierten Eigenschaften von vRealize Automation die WIM-, SCCM- oder Linux Kickstart-Bereitstellung starten. Mithilfe benutzerdefinierter Eigenschaften können Sie auch Ihre eigenen PowerShell-Skripts aufrufen. Für die Linux Kickstart-Bereitstellung sind keine benutzerdefinierten Eigenschaften erforderlich.

- **Benutzerdefinierte Eigenschaften für OVF-Import**

Wenn Sie eine OVF-Datei in einen Blueprint importieren, können Sie mehrere Einstellungen als benutzerdefinierte Eigenschaften importieren und konfigurieren.

- **Benutzerdefinierte Eigenschaften für den vRealize Automation-Gast-Agent**

Wenn Sie den vRealize Automation-Gast-Agent in Ihren Vorlagen für das Klonen oder in WinPE installiert haben, können Sie mithilfe benutzerdefinierter Eigenschaften benutzerdefinierte Skripts innerhalb des Gastbetriebssystems einer bereitgestellten Maschine ausführen, nachdem die Maschine vollständig bereitgestellt wurde.

- **Benutzerdefinierte Eigenschaften für die Integration von BMC BladeLogic Configuration Manager**

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für die Integration von BMC BladeLogic Configuration Manager bereitstellen können.

- **Benutzerdefinierte Eigenschaften für die Integration von HP Server Automation**

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für die Integration von HP Server Automation bereitstellen können. Einige benutzerdefinierte Eigenschaften sind für die Integration von HP Server Automation erforderlich. Andere benutzerdefinierte Eigenschaften sind optional.

Benutzerdefinierte Eigenschaften für Bereitstellungen

vRealize Automation bietet mehrere benutzerdefinierte Eigenschaften, die für die meisten Bereitstellungen gelten.

Tabelle 1-1. Benutzerdefinierte Eigenschaften für Blueprints und Bereitstellungen

Benutzerdefinierte Eigenschaft	Beschreibung
<code>_debug_deployment</code>	<p>Mit Ausnahme von Skalierungsvorgängen, die teilweise erfolgreiche Bereitstellungen ermöglichen, wird standardmäßig die gesamte Bereitstellung gelöscht, wenn einzelne Ressourcen nicht bereitgestellt werden können. Sie können das Standardverhalten überschreiben, indem Sie den benutzerdefinierten Eigenschaftswert <code>_debug_deployment</code> auf „true“ festlegen. Wenn die Bereitstellung fehlschlägt, verhindert die benutzerdefinierte Debugging-Eigenschaft, dass ein Rollback der Ressourcen durchgeführt wird. Auf diese Weise können Sie ermitteln, welche Komponenten bei der Bereitstellung fehlgeschlagen sind.</p> <p>Mit anderen Worten: Wenn Sie <code>_debug_deployment</code> auf „true“ festlegen, können Sie die Anpassung und den ersten Start (z. B. Agent) einfacher debuggen, da durch die Einstellung sichergestellt ist, dass Maschinen nach einem Bereitstellungsfehler nicht gelöscht werden. Andernfalls ändert die Einstellung nichts direkt am Bereitstellungsvorgang und wirkt sich nicht direkt auf den Gast-Agent oder die Anpassung aus (z. B. Festlegen unserer Ergebnisse relativ zu einer vCenter-Anpassungsspezifikation).</p> <p>Hinweis: Auf ein fehlgeschlagenes Katalogelement kann normalerweise nicht zugegriffen werden, da bei einem Fehler sofort ein Rollback durchgeführt wird. Wenn <code>_debug_deployment</code> jedoch auf „true“ festgelegt ist, behandelt vRealize Automation die andernfalls fehlgeschlagene Bereitstellung als teilweise erfolgreich und ermöglicht somit den Zugriff.</p> <p>Um die benutzerdefinierte Eigenschaft auf einen Blueprint anzuwenden, fügen Sie <code>_debug_deployment</code> beim Erstellen oder Bearbeiten eines Blueprints über die Registerkarte Eigenschaften zur Seite Blueprint-Eigenschaften hinzu. Die <code>_debug_deployment</code>-Eigenschaft wird auf der Softwarebereitstellungsebene verwendet, und nicht auf der Ebene der Gast-Agent- oder Maschinenbereitstellung.</p> <p>Sie können vRealize Automation auch so konfigurieren, dass virtuelle Maschinen nach dem Fehlschlagen der Bereitstellung nicht gelöscht werden, indem Sie Einstellungen in der Datei <code>VRMAgent.exe.config</code> verwenden.</p>
<code>_deploymentName</code>	<p>Wenn diese Eigenschaft einem Blueprint hinzugefügt wird, können Sie damit einen benutzerdefinierten Namen für die Bereitstellung angeben, indem Sie den Wert von <code>_deploymentName</code> auf Ihre benutzerdefinierte Zeichenfolge festlegen. Wenn mehrere Instanzen dieser Bereitstellung in einer Einzelanforderung bereitgestellt werden, wird der benutzerdefinierte Name als Präfix verwendet. Wenn Sie</p>

Tabelle 1-1. Benutzerdefinierte Eigenschaften für Blueprints und Bereitstellungen (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
	<p>möchten, dass Benutzer eigene Bereitstellungsnamen angeben, legen Sie diese Eigenschaft so fest, dass Überschreibungen zulässig sind. Die folgenden zwei Vorbehalte sind für die Nutzung erforderlich:</p> <ul style="list-style-type: none"> ■ Sie müssen diese Eigenschaft auf Blueprint-Ebene hinzufügen, nicht auf Komponentenebene. Klicken Sie beispielsweise beim Erstellen oder Bearbeiten eines Blueprints auf die Registerkarte Eigenschaften und wählen Sie anschließend Benutzerdefinierte Eigenschaften > Neu aus, um die <code>_deploymentName</code>-Eigenschaft dem Blueprint hinzuzufügen. Fügen Sie die Eigenschaft keiner Maschine oder anderen Komponente im Blueprint hinzu. ■ Sie müssen diese Eigenschaft als separate Eigenschaft und nicht als Mitglied einer Eigenschaftsgruppe hinzufügen.

Benutzerdefinierte Eigenschaften zum Benennen und Analysieren von Bereitstellungen

Wenn die Bereitstellung fehlschlägt, führt vRealize Automation ein Rollback aller im Katalogelement enthaltenen Ressourcen durch. Bei Bereitstellungen mit mehreren Komponenten können Sie eine benutzerdefinierte Eigenschaft verwenden, um die Standardeigenschaften zu überschreiben und um Informationen zum Debugging des Fehlers zu erhalten. Diese Eigenschaften werden idealerweise auf den gesamten Blueprint angewendet.

Tabelle 1-2. Benutzerdefinierte Eigenschaften für Analyse-Bereitstellungen

Benutzerdefinierte Eigenschaft	Beschreibung
<code>_debug_deployment</code>	<p>Mit Ausnahme von Skalierungsvorgängen, die teilweise erfolgreiche Bereitstellungen ermöglichen, wird standardmäßig die gesamte Bereitstellung gelöscht, wenn einzelne Ressourcen nicht bereitgestellt werden können. Sie können das Standardverhalten überschreiben, indem Sie den benutzerdefinierten Eigenschaftswert <code>_debug_deployment</code> auf „true“ festlegen. Wenn die Bereitstellung fehlschlägt, verhindert die benutzerdefinierte Debugging-Eigenschaft, dass ein Rollback der Ressourcen durchgeführt wird. Auf diese Weise können Sie ermitteln, welche Komponenten bei der Bereitstellung fehlgeschlagen sind.</p> <p>Hinweis: Auf ein fehlgeschlagenes Katalogelement kann normalerweise nicht zugegriffen werden, da bei einem Fehler sofort ein Rollback durchgeführt wird. Wenn <code>_debug_deployment</code> jedoch auf „true“ festgelegt ist, behandelt vRealize Automation die andernfalls fehlgeschlagene Bereitstellung als teilweise erfolgreich und ermöglicht somit den Zugriff.</p> <p>Mit anderen Worten: Wenn Sie <code>_debug_deployment</code> auf „true“ festlegen, können Sie die Anpassung und den ersten Start (z. B. Agent) einfacher debuggen, da durch die Einstellung sichergestellt ist, dass Maschinen nach einem Bereitstellungsfehler nicht gelöscht werden. Andernfalls ändert die Einstellung nichts direkt am Bereitstellungsvorgang und wirkt sich nicht direkt auf den Gast-Agent oder die Anpassung aus (z. B. Festlegen unserer Ergebnisse relativ zu einer vCenter-Anpassungsspezifikation).</p> <p>Um die benutzerdefinierte Eigenschaft auf einen Blueprint anzuwenden, fügen Sie <code>_debug_deployment</code> beim Erstellen oder Bearbeiten eines Blueprints über die Registerkarte Eigenschaften zur Seite Blueprint-Eigenschaften hinzu. Die <code>_debug_deployment</code>-Eigenschaft wird auf der Softwarebereitstellungsebene verwendet, und nicht auf der Ebene der Gast-Agent- oder Maschinenbereitstellung.</p> <p>Sie können vRealize Automation auch so konfigurieren, dass virtuelle Maschinen nach dem Fehlschlagen der Bereitstellung nicht gelöscht werden, indem Sie Einstellungen in der Datei <code>VRMAgent.exe.config</code> verwenden.</p>
<code>_deploymentName</code>	<p>Wenn diese Eigenschaft einem Blueprint hinzugefügt wird, können Sie damit einen benutzerdefinierten Namen für die Bereitstellung angeben, indem Sie den Wert von <code>_deploymentName</code> auf Ihre benutzerdefinierte Zeichenfolge festlegen. Wenn mehrere Instanzen dieser Bereitstellung in einer Einzelanforderung bereitgestellt werden, wird der benutzerdefinierte Name als Präfix verwendet. Wenn Sie</p>

Tabelle 1-2. Benutzerdefinierte Eigenschaften für Analyse-Bereitstellungen (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
	<p>möchten, dass Benutzer eigene Bereitstellungsnamen angeben, legen Sie diese Eigenschaft so fest, dass Überschreibungen zulässig sind. Die folgenden zwei Vorbehalte sind für die Nutzung erforderlich:</p> <ul style="list-style-type: none"> ■ Sie müssen diese Eigenschaft auf Blueprint-Ebene hinzufügen, nicht auf Komponentenebene. Klicken Sie beispielsweise beim Erstellen oder Bearbeiten eines Blueprints auf die Registerkarte Eigenschaften und wählen Sie anschließend Benutzerdefinierte Eigenschaften > Neu aus, um die <code>_deploymentName</code>-Eigenschaft dem Blueprint hinzuzufügen. Fügen Sie die Eigenschaft keiner Maschine oder anderen Komponente im Blueprint hinzu. ■ Sie müssen diese Eigenschaft als separate Eigenschaft und nicht als Mitglied einer Eigenschaftsgruppe hinzufügen.

Benutzerdefinierte Eigenschaften für OpenStack-Endpoints

vRealize Automation enthält benutzerdefinierte Eigenschaften, die Sie bei der Konfiguration von OpenStack-Endpoints in vRealize Automation verwenden können.

Tabelle 1-3. Benutzerdefinierte Eigenschaften für Openstack-Endpoints

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.Admin.ConnectAddress.Regex	<p>Wird von einem vRealize Automation-Administrator für die Definition eines regulären Ausdrucks verwendet, um eine passende IP-Adresse für Terminalverbindungen wie etwa eine RDP-Verbindung zuzuordnen. Wenn eine passende IP-Adresse gefunden wird, wird sie unter der benutzerdefinierten Eigenschaft VirtualMachine.Admin.ConnectAddress gespeichert. Andernfalls wird die erste verfügbare IP-Adresse zugewiesen.</p> <p>Beispielsweise kann durch Festlegen des Eigenschaftswerts auf 10.10.0. eine IP-Adresse in einem 10.10.0.*-Subnetz ausgewählt werden, das der virtuellen Maschine zugewiesen ist. Wenn das Subnetz nicht zugewiesen wurde, wird die Eigenschaft ignoriert. Diese Eigenschaft ist für die Verwendung mit OpenStack verfügbar.</p>
VirtualMachine.NetworkN.AdditionAddressM	<p>Definiert eine zusätzliche IP-Adresse <i>M</i>, die einer OpenStack-Instanz für das Netzwerk <i>N</i> zugeteilt wird, mit Ausnahme des IP-Adresssatzes, der durch die Eigenschaft VirtualMachine.NetworkN angegeben wird. Address. angegeben wird. Weitere Adressen werden auf der Registerkarte „Netzwerk“ in der Spalte „Zusätzliche Adressen“ angezeigt.</p> <p>Diese Eigenschaft wird von der Zustandsdatenerfassung der OpenStack-Maschine verwendet. Für diese Eigenschaft werden nur vom OpenStack-Endpoint Daten erfasst, aber sie gilt nicht speziell für OpenStack und kann von anderen Endpoint-Typen für die Lebenszyklus-Erweiterbarkeit verwendet werden.</p> <p>Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-3. Benutzerdefinierte Eigenschaften für Openstack-Endpoints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VMware.Endpoint.Openstack.IdentityProvider.Domain.Name	<p>Ermöglicht es, dass vRealize Automation die erforderliche Authentifizierung des KeystoneV3-Domänennamens unterstützt. Wenn Keystone V3 aktiviert ist, können Sie die Eigenschaft verwenden, um dem OpenStack-Endpoint eine bestimmte Domäne zur Authentifizierung mit einem Keystone V3 OpenStack-Identitätsanbieter zuzuweisen.</p> <ul style="list-style-type: none"> ■ Fügen Sie für neue Endpoints die benutzerdefinierte Eigenschaft hinzu, um eine bestimmte Domäne zuzuweisen. ■ Fügen Sie für aktualisierte oder migrierte Endpoints die benutzerdefinierte Eigenschaft nur dann hinzu, wenn die Datenerfassung nach einem Upgrade oder einer Migration fehlschlägt.
VMware.Endpoint.Openstack.IdentityProvider.Version	<p>Gibt die Version des OpenStack-Identitätsanbieters (Keystone) an, die beim Authentifizieren eines OpenStack-Endpoints verwendet werden soll. Konfigurieren Sie den Wert 3 für die Authentifizierung mit dem OpenStack-Identitätsanbieter Keystone V3. Wenn Sie einen anderen Wert verwenden oder diese benutzerdefinierte Eigenschaft nicht verwenden, wird für die Authentifizierung standardmäßig Keystone V2 verwendet.</p>

Benutzerdefinierte Eigenschaften für Klon-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für Klon-Blueprints bereitstellen können.

Tabelle 1-4. Benutzerdefinierte Eigenschaften für Klon-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.StorageReservationPolicy</code>	<p>Gibt die Speicherreservierungsrichtlinie an, die für die Suche nach Speicher für die Festplatte <i>N</i> verwendet werden soll. Hiermit wird außerdem die benannte Speicherreservierungsrichtlinie einem Volume zugewiesen. Um diese Eigenschaft zu verwenden, ersetzen Sie die Volume-Nummer im Eigenschaftsnamen durch <i>N</i> und geben Sie den Namen einer Speicherreservierungsrichtlinie als Wert an. Diese Eigenschaft entspricht dem Namen der im Blueprint angegebenen Speicherreservierungsrichtlinie. Festplatten müssen sequenziell nummeriert werden. Diese Eigenschaft kann für alle virtuellen Reservierungen und vCloud-Reservierungen verwendet werden. Für physische, Amazon- oder OpenStack-Reservierungen kann diese Eigenschaft nicht verwendet werden.</p>
<code>VirtualMachine.NetworkN.NetworkProfileName</code>	<p>Gibt den Namen eines Netzwerkprofils an, aus dem dem Netzwerkgerät <i>N</i> eine statische IP-Adresse zugewiesen werden soll oder aus dem der statische IP-Adressbereich bezogen werden soll, der dem Netzwerkgerät <i>N</i> einer geklonten Maschine zugewiesen werden kann. Dabei steht <i>N</i>=0 für das erste Gerät, 1 für das zweite Gerät usw.</p> <p>Das Netzwerkprofil, auf das die Eigenschaft verweist, wird zum Zuweisen einer IP-Adresse verwendet. Die Eigenschaft bestimmt das Netzwerk, an das die Maschine angeschlossen ist, basierend auf der Reservierung.</p> <p>Die Änderung dieses Eigenschaftswerts hat nach der Zuweisung des Netzwerks keine Auswirkung auf die erwarteten IP-Adresswerte für die angegebenen Maschinen.</p> <p>Mit der WIM-basierten Bereitstellung für virtuelle Maschinen können Sie mithilfe dieser Eigenschaft ein Netzwerkprofil und eine Netzwerkschnittstelle angeben. Sie können aber auch den Abschnitt „Netzwerk“ der Seite „Virtuelle Reservierung“ verwenden.</p> <p>Die folgenden Attribute des Netzwerkprofils sind für die Zuweisung von statischen IP-Adressen in einem Klon-Blueprint verfügbar:</p> <ul style="list-style-type: none"> ■ <code>VirtualMachine.NetworkN.SubnetMask</code> ■ <code>VirtualMachine.NetworkN.Gateway</code> ■ <code>VirtualMachine.NetworkN.PrimaryDns</code> ■ <code>VirtualMachine.NetworkN.SecondaryDns</code> ■ <code>VirtualMachine.NetworkN.PrimaryWins</code> ■ <code>VirtualMachine.NetworkN.SecondaryWins</code> ■ <code>VirtualMachine.NetworkN.DnsSuffix</code> ■ <code>VirtualMachine.NetworkN.DnsSearchSuffixes</code> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer</p>

Tabelle 1-4. Benutzerdefinierte Eigenschaften für Klon-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>Linux.ExternalScript.Name</code>	<p>Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p> <p>Gibt den Namen eines optionalen Anpassungsskripts an, wie beispielsweise <code>config.sh</code>, das der Linux-Gast-Agent nach der Installation des Betriebssystems ausführt. Diese Eigenschaft ist für über Vorlagen geklonte Linux-Maschinen verfügbar, auf denen der Linux-Agent installiert ist.</p> <p>Wenn Sie ein externes Skript angeben, müssen Sie auch mithilfe der Eigenschaften <code>Linux.ExternalScript.LocationType</code> und <code>Linux.ExternalScript.Path</code> dessen Speicherort definieren.</p>
<code>Linux.ExternalScript.LocationType</code>	<p>Gibt den Speicherorttyp des in der Eigenschaft <code>Linux.ExternalScript.Name</code> benannten Anpassungsskripts an. Mögliche Werte sind „local“ oder „nfs“.</p> <p>Darüber hinaus müssen Sie mit der Eigenschaft <code>Linux.ExternalScript.Path</code> den Skriptspeicherort angeben. Wenn der Speicherorttyp „nfs“ lautet, sollten Sie auch die Eigenschaft <code>Linux.ExternalScript.Server</code> verwenden.</p>
<code>Linux.ExternalScript.Server</code>	<p>Gibt den Namen des NFS-Servers an, wie beispielsweise „lab-ad.lab.local“, auf dem das in <code>Linux.ExternalScript.Name</code> angegebene externe Linux-Anpassungsskript gespeichert ist.</p>
<code>Linux.ExternalScript.Path</code>	<p>Gibt den lokalen Pfad zum Linux-Anpassungsskript oder den Exportpfad zur Linux-Anpassung auf dem NFS-Server an. Dieser Wert muss mit einem Schrägstrich beginnen und darf den Dateinamen nicht enthalten, wie beispielsweise <code>/scripts/linux/config.sh</code>.</p>

Wenn Ihre Systemadministratoren den Gast-Agent installiert haben, um Skripts auszuführen, die benutzerdefinierte Eigenschaften akzeptieren und bereitgestellte Maschinen anpassen, können Sie mithilfe von benutzerdefinierten Eigenschaften weitere Anpassungen an geklonten Maschinen vornehmen, die den Gast-Agent verwenden.

Tabelle 1-5. Benutzerdefinierte Eigenschaften zum Anpassen geklonter Maschinen mit einem Gast-Agent

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.AllowLogin</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um den Maschinenbesitzer zur Gruppe der lokalen Remotedesktopbenutzer, gemäß der Eigenschaft <code>VirtualMachine.Admin.Owner</code> , hinzuzufügen.
<code>VirtualMachine.Admin.UseGuestAgent</code>	Wenn der Gast-Agent als Dienst in einer Vorlage für das Klonen installiert ist, legen Sie diese Eigenschaft im Maschinen-Blueprint auf „True“ fest, um den Gast-Agent-Dienst auf Maschinen, die anhand dieser Vorlage geklonet werden, zu aktivieren. Beim Starten der Maschine wird der Gast-Agent-Dienst gestartet. Legen Sie diese Eigenschaft auf „False“ fest, um den Gast-Agent zu deaktivieren. Mit der Einstellung „False“ verwendet der erweiterte Klon-Workflow den Gast-Agent nicht für Aufgaben des Gastbetriebssystems, wodurch dessen Funktionalität auf <code>VMwareCloneWorkflow</code> reduziert wird. Wenn diese Option nicht angegeben ist oder auf einen anderen Wert als „False“ festgelegt ist, sendet der erweiterte Klon-Workflow Arbeitselemente an den Gast-Agent.
<code>VirtualMachine.DiskN.Active</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine aktiv ist. Legen Sie diese Eigenschaft auf „False“ fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine nicht aktiv ist.
<code>VirtualMachine.DiskN.Label</code>	Gibt die Bezeichnung für die Festplatte <i>N</i> einer Maschine an. Für die Festplattenbezeichnung sind maximal 32 Zeichen zulässig. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent wird hiermit die Bezeichnung der Festplatte <i>N</i> einer Maschine im Gastbetriebssystem angegeben.
<code>VirtualMachine.DiskN.Letter</code>	Gibt den Laufwerkbuchstaben oder Mount-Punkt der Festplatte <i>N</i> einer Maschine an. Der Standardwert ist C. Um beispielsweise den Buchstaben D für die Festplatte 1 anzugeben, definieren Sie die benutzerdefinierte Eigenschaft als <code>VirtualMachine.Disk1.Letter</code> und geben Sie den Wert „D“ ein. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent gibt dieser Wert den Laufwerkbuchstaben oder Einhängepunkt an, unter dem die zusätzliche Festplatte <i>N</i> vom Gast-Agent im Gastbetriebssystem gemountet wird.

Tabelle 1-5. Benutzerdefinierte Eigenschaften zum Anpassen geklonter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.CustomizeGuestOSDelay</code>	Gibt an, wie lange nach Abschluss der Anpassung gewartet werden soll, bevor die Anpassung des Gastbetriebssystems gestartet wird. Für diesen Wert ist das Format HH:MM:SS erforderlich. Wenn dieser Wert nicht festgelegt wird, wird der Standardwert von einer Minute (00:01:00) verwendet. Wenn Sie diese benutzerdefinierte Eigenschaft nicht angeben, schlägt die Bereitstellung möglicherweise fehl, falls die virtuelle Maschine neu gestartet wird, bevor Arbeitselemente des Gast-Agents abgeschlossen sind.
<code>VirtualMachine.Customize.WaitComplete</code>	Legen Sie diese Eigenschaft auf „True“ fest, um zu verhindern, dass der Bereitstellungsworkflow Arbeitselemente an den Gast-Agent sendet, bevor alle Anpassungen abgeschlossen sind. Legen Sie diese Eigenschaft auf „False“ fest, um die Erstellung von Arbeitselementen zu ermöglichen, bevor die Anpassung abgeschlossen ist.
<code>VirtualMachine.SoftwareN.Name</code>	Gibt den beschreibenden Namen der Softwareanwendung <i>N</i> oder eines Skripts an, die bzw. das während der Bereitstellung installiert oder ausgeführt werden soll. Dies ist eine optionale und rein informative Eigenschaft. Sie hat keine echte Funktion für den erweiterten Klon-Workflow oder den Gast-Agent, ist aber hilfreich für die benutzerdefinierte Softwareauswahl in einer Benutzeroberfläche oder für Berichte zur Softwarenutzung.
<code>VirtualMachine.SoftwareN.ScriptPath</code>	Gibt den vollständigen Pfad zum Installationsskript einer Anwendung an. Bei dem Pfad muss es sich um einen gültigen absoluten Pfad wie er im Gastbetriebssystem angezeigt wird handeln und er muss den Namen der Skriptdatei enthalten. Sie können benutzerdefinierte Eigenschaftswerte als Parameter an das Skript übergeben, indem Sie <code>{CustomPropertyName}</code> in der Pfadzeichenfolge einfügen. Angenommen, Sie haben eine benutzerdefinierte Eigenschaft <code>ActivationKey</code> mit dem Wert 1234. In diesem Fall lautet der Skriptpfad <code>D:\InstallApp.bat -key {ActivationKey}</code> . Der Gast-Agent führt den Befehl <code>D:\InstallApp.bat -key 1234</code> aus. Ihre Skriptdatei kann dann so programmiert werden, dass dieser Wert akzeptiert und verwendet wird.

Tabelle 1-5. Benutzerdefinierte Eigenschaften zum Anpassen geklonter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.SoftwareN.ISOName</code>	Gibt den Pfad und den Dateinamen der ISO-Datei in Bezug auf das Stammverzeichnis des Datenspeichers an. Das Format lautet <i>/Ordnername/Unterordnername/Dateiname.iso</i> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.
<code>VirtualMachine.SoftwareN.ISOLocation</code>	Gibt den Speicherpfad an, der die ISO-Imagedatei enthält, die von der Anwendung oder dem Skript verwendet werden soll. Formatieren Sie den in der Hostreservierung angezeigten Pfad, wie beispielsweise <code>netapp-1:it_nfs_1</code> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.

Benutzerdefinierte Eigenschaften für verknüpfte Klon-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für verknüpfte Klon-Blueprints bereitstellen können.

Bestimmte benutzerdefinierte Eigenschaften von vRealize Automation sind für die Verwendung mit verknüpften Klon-Blueprints erforderlich.

Tabelle 1-6. Benutzerdefinierte Eigenschaften für verknüpfte Klon-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.Storage</code>	Gibt den Datenspeicher an, auf dem die Festplatte <i>N</i> der Maschine platziert werden soll, wie beispielsweise <code>DATASTORE01</code> . Mithilfe dieser Eigenschaft wird auch ein einzelner Datenspeicher zu einem verknüpften Klon-Blueprint hinzugefügt. <i>N</i> ist der Index (beginnend mit 0) des zuzuweisenden Volumes. Geben Sie den Namen des Datenspeichers ein, der dem Volume zugewiesen werden soll. Dies ist der Datenspeichername, der auf der Seite „Computing-Ressource bearbeiten“ im Feld „Speicherpfad“ angezeigt wird. Festplatten müssen sequenziell nummeriert werden.
<code>VirtualMachine.DiskN.StorageReservationPolicy</code>	Gibt die Speicherreservierungsrichtlinie an, die für die Suche nach Speicher für die Festplatte <i>N</i> verwendet werden soll. Hiermit wird außerdem die benannte Speicherreservierungsrichtlinie einem Volume zugewiesen. Um diese Eigenschaft zu verwenden, ersetzen Sie die Volume-Nummer im Eigenschaftsnamen durch <i>N</i> und geben Sie den Namen einer Speicherreservierungsrichtlinie als Wert an. Diese Eigenschaft entspricht dem Namen der im Blueprint angegebenen Speicherreservierungsrichtlinie. Festplatten müssen sequenziell nummeriert werden. Diese Eigenschaft kann für alle virtuellen Reservierungen und vCloud-Reservierungen verwendet werden. Für physische, Amazon- oder OpenStack-Reservierungen kann diese Eigenschaft nicht verwendet werden.

Tabelle 1-6. Benutzerdefinierte Eigenschaften für verknüpfte Klon-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.Label</code>	Gibt die Bezeichnung für die Festplatte <i>N</i> einer Maschine an. Für die Festplattenbezeichnung sind maximal 32 Zeichen zulässig. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent wird hiermit die Bezeichnung der Festplatte <i>N</i> einer Maschine im Gastbetriebssystem angegeben.
<code>VirtualMachine.DiskN.Letter</code>	Gibt den Laufwerkbuchstaben oder Mount-Punkt der Festplatte <i>N</i> einer Maschine an. Der Standardwert ist C. Um beispielsweise den Buchstaben D für die Festplatte 1 anzugeben, definieren Sie die benutzerdefinierte Eigenschaft als <code>VirtualMachine.Disk1.Letter</code> und geben Sie den Wert „D“ ein. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent gibt dieser Wert den Laufwerkbuchstaben oder Einhängepunkt an, unter dem die zusätzliche Festplatte <i>N</i> vom Gast-Agent im Gastbetriebssystem gemountet wird.
<code>MaximumProvisionedMachines</code>	Gibt die maximale Anzahl von verknüpften Klonen für einen Snapshot der virtuellen Maschine an. Der Standardwert ist „unlimited“.
<code>Linux.ExternalScript.Name</code>	Gibt den Namen eines optionalen Anpassungsskripts an, wie beispielsweise <code>config.sh</code> , das der Linux-Gast-Agent nach der Installation des Betriebssystems ausführt. Diese Eigenschaft ist für über Vorlagen geklonte Linux-Maschinen verfügbar, auf denen der Linux-Agent installiert ist. Wenn Sie ein externes Skript angeben, müssen Sie auch mithilfe der Eigenschaften <code>Linux.ExternalScript.LocationType</code> und <code>Linux.ExternalScript.Path</code> dessen Speicherort definieren.
<code>Linux.ExternalScript.LocationType</code>	Gibt den Speicherorttyp des in der Eigenschaft <code>Linux.ExternalScript.Name</code> benannten Anpassungsskripts an. Mögliche Werte sind „local“ oder „nfs“. Darüber hinaus müssen Sie mit der Eigenschaft <code>Linux.ExternalScript.Path</code> den Skriptspeicherort angeben. Wenn der Speicherorttyp „nfs“ lautet, sollten Sie auch die Eigenschaft <code>Linux.ExternalScript.Server</code> verwenden.
<code>Linux.ExternalScript.Server</code>	Gibt den Namen des NFS-Servers an, wie beispielsweise „lab-ad.lab.local“, auf dem das in <code>Linux.ExternalScript.Name</code> angegebene externe Linux-Anpassungsskript gespeichert ist.
<code>Linux.ExternalScript.Path</code>	Gibt den lokalen Pfad zum Linux-Anpassungsskript oder den Exportpfad zur Linux-Anpassung auf dem NFS-Server an. Dieser Wert muss mit einem Schrägstrich beginnen und darf den Dateinamen nicht enthalten, wie beispielsweise <code>/scripts/linux/config.sh</code> .

Wenn Sie den Gast-Agent zum Anpassen geklonter Maschinen installiert haben, werden Sie bestimmte benutzerdefinierte Eigenschaften öfter als andere verwenden.

Tabelle 1-7. Benutzerdefinierte Eigenschaften zum Anpassen geklonter Maschinen mit einem Gast-Agent

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.Admin.UseGuestAgent	Wenn der Gast-Agent als Dienst in einer Vorlage für das Klonen installiert ist, legen Sie diese Eigenschaft im Maschinen-Blueprint auf „True“ fest, um den Gast-Agent-Dienst auf Maschinen, die anhand dieser Vorlage geklonet werden, zu aktivieren. Beim Starten der Maschine wird der Gast-Agent-Dienst gestartet. Legen Sie diese Eigenschaft auf „False“ fest, um den Gast-Agent zu deaktivieren. Mit der Einstellung „False“ verwendet der erweiterte Klon-Workflow den Gast-Agent nicht für Aufgaben des Gastbetriebssystems, wodurch dessen Funktionalität auf VMwareCloneWorkflow reduziert wird. Wenn diese Option nicht angegeben ist oder auf einen anderen Wert als „False“ festgelegt ist, sendet der erweiterte Klon-Workflow Arbeitselemente an den Gast-Agent.
VirtualMachine.Admin.CustomizeGuestOSDelay	Gibt an, wie lange nach Abschluss der Anpassung gewartet werden soll, bevor die Anpassung des Gastbetriebssystems gestartet wird. Für diesen Wert ist das Format HH:MM:SS erforderlich. Wenn dieser Wert nicht festgelegt wird, wird der Standardwert von einer Minute (00:01:00) verwendet. Wenn Sie diese benutzerdefinierte Eigenschaft nicht angeben, schlägt die Bereitstellung möglicherweise fehl, falls die virtuelle Maschine neu gestartet wird, bevor Arbeitselemente des Gast-Agents abgeschlossen sind.

Tabelle 1-7. Benutzerdefinierte Eigenschaften zum Anpassen geklonter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Customize.WaitComplete</code>	Legen Sie diese Eigenschaft auf „True“ fest, um zu verhindern, dass der Bereitstellungsworkflow Arbeitselemente an den Gast-Agent sendet, bevor alle Anpassungen abgeschlossen sind. Legen Sie diese Eigenschaft auf „False“ fest, um die Erstellung von Arbeitselementen zu ermöglichen, bevor die Anpassung abgeschlossen ist.
<code>VirtualMachine.SoftwareN.ScriptPath</code>	<p>Gibt den vollständigen Pfad zum Installationsskript einer Anwendung an. Bei dem Pfad muss es sich um einen gültigen absoluten Pfad wie er im Gastbetriebssystem angezeigt wird handeln und er muss den Namen der Skriptdatei enthalten.</p> <p>Sie können benutzerdefinierte Eigenschaftswerte als Parameter an das Skript übergeben, indem Sie <code>{CustomPropertyName}</code> in der Pfadzeichenfolge einfügen. Angenommen, Sie haben eine benutzerdefinierte Eigenschaft <code>ActivationKey</code> mit dem Wert 1234. In diesem Fall lautet der Skriptpfad <code>D:\InstallApp.bat -key {ActivationKey}</code>. Der Gast-Agent führt den Befehl <code>D:\InstallApp.bat -key 1234</code> aus. Ihre Skriptdatei kann dann so programmiert werden, dass dieser Wert akzeptiert und verwendet wird.</p>

Benutzerdefinierte Eigenschaften für FlexClone-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für FlexClone-Blueprints bereitstellen können.

Tabelle 1-8. Benutzerdefinierte Eigenschaften für FlexClone-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.NetworkN.NetworkProfileName	<p>Gibt den Namen eines Netzwerkprofils an, aus dem dem Netzwerkgerät <i>N</i> eine statische IP-Adresse zugewiesen werden soll oder aus dem der statische IP-Adressbereich bezogen werden soll, der dem Netzwerkgerät <i>N</i> einer geklonten Maschine zugewiesen werden kann. Dabei steht <i>N=0</i> für das erste Gerät, 1 für das zweite Gerät usw.</p> <p>Das Netzwerkprofil, auf das die Eigenschaft verweist, wird zum Zuweisen einer IP-Adresse verwendet. Die Eigenschaft bestimmt das Netzwerk, an das die Maschine angeschlossen ist, basierend auf der Reservierung.</p> <p>Die Änderung dieses Eigenschaftswerts hat nach der Zuweisung des Netzwerks keine Auswirkung auf die erwarteten IP-Adresswerte für die angegebenen Maschinen.</p> <p>Mit der WIM-basierten Bereitstellung für virtuelle Maschinen können Sie mithilfe dieser Eigenschaft ein Netzwerkprofil und eine Netzwerkschnittstelle angeben. Sie können aber auch den Abschnitt „Netzwerk“ der Seite „Virtuelle Reservierung“ verwenden.</p> <p>Die folgenden Attribute des Netzwerkprofils sind für die Zuweisung von statischen IP-Adressen in einem Klon-Blueprint verfügbar:</p> <ul style="list-style-type: none"> ■ VirtualMachine.NetworkN.SubnetMask ■ VirtualMachine.NetworkN.Gateway ■ VirtualMachine.NetworkN.PrimaryDns ■ VirtualMachine.NetworkN.SecondaryDns ■ VirtualMachine.NetworkN.PrimaryWins ■ VirtualMachine.NetworkN.SecondaryWins ■ VirtualMachine.NetworkN.DnsSuffix ■ VirtualMachine.NetworkN.DnsSearchSuffixes <p>Die benutzerdefinierten Eigenschaften VirtualMachine.Network<i>N</i> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p>
Linux.ExternalScript.Name	<p>Gibt den Namen eines optionalen Anpassungsskripts an, wie beispielsweise <code>config.sh</code>, das der Linux-Gast-Agent nach der Installation des Betriebssystems ausführt. Diese Eigenschaft ist für über Vorlagen geklonte Linux-Maschinen verfügbar, auf denen der Linux-Agent installiert ist.</p> <p>Wenn Sie ein externes Skript angeben, müssen Sie auch mithilfe der Eigenschaften Linux.ExternalScript.LocationType und Linux.ExternalScript.Path dessen Speicherort definieren.</p>

Tabelle 1-8. Benutzerdefinierte Eigenschaften für FlexClone-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>Linux.ExternalScript.LocationType</code>	Gibt den Speicherorttyp des in der Eigenschaft <code>Linux.ExternalScript.Name</code> benannten Anpassungsskripts an. Mögliche Werte sind „local“ oder „nfs“. Darüber hinaus müssen Sie mit der Eigenschaft <code>Linux.ExternalScript.Path</code> den Skriptspeicherort angeben. Wenn der Speicherorttyp „nfs“ lautet, sollten Sie auch die Eigenschaft <code>Linux.ExternalScript.Server</code> verwenden.
<code>Linux.ExternalScript.Server</code>	Gibt den Namen des NFS-Servers an, wie beispielsweise „lab-ad.lab.local“, auf dem das in <code>Linux.ExternalScript.Name</code> angegebene externe Linux-Anpassungsskript gespeichert ist.
<code>Linux.ExternalScript.Path</code>	Gibt den lokalen Pfad zum Linux-Anpassungsskript oder den Exportpfad zur Linux-Anpassung auf dem NFS-Server an. Dieser Wert muss mit einem Schrägstrich beginnen und darf den Dateinamen nicht enthalten, wie beispielsweise <code>/scripts/linux/config.sh</code> .

Wenn Sie den Gast-Agent installiert haben, um geklonte Maschinen anzupassen, werden in der Tabelle „Benutzerdefinierte Eigenschaften zum Anpassen von FlexClone-Maschinen mit einem Gast-Agent“ die am häufigsten verwendeten benutzerdefinierten Eigenschaften für Ihre Situation beschrieben.

Tabelle 1-9. Benutzerdefinierte Eigenschaften zum Anpassen von FlexClone-Maschinen mit einem Gast-Agent

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.UseGuestAgent</code>	Wenn der Gast-Agent als Dienst in einer Vorlage für das Klonen installiert ist, legen Sie diese Eigenschaft im Maschinen-Blueprint auf „True“ fest, um den Gast-Agent-Dienst auf Maschinen, die anhand dieser Vorlage geklont werden, zu aktivieren. Beim Starten der Maschine wird der Gast-Agent-Dienst gestartet. Legen Sie diese Eigenschaft auf „False“ fest, um den Gast-Agent zu deaktivieren. Mit der Einstellung „False“ verwendet der erweiterte Klon-Workflow den Gast-Agent nicht für Aufgaben des Gastbetriebssystems, wodurch dessen Funktionalität auf VMwareCloneWorkflow reduziert wird. Wenn diese Option nicht angegeben ist oder auf einen anderen Wert als „False“ festgelegt ist, sendet der erweiterte Klon-Workflow Arbeitselemente an den Gast-Agent.
<code>VirtualMachine.DiskN.Label</code>	Gibt die Bezeichnung für die Festplatte <i>N</i> einer Maschine an. Für die Festplattenbezeichnung sind maximal 32 Zeichen zulässig. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent wird hiermit die Bezeichnung der Festplatte <i>N</i> einer Maschine im Gastbetriebssystem angegeben.

Tabelle 1-9. Benutzerdefinierte Eigenschaften zum Anpassen von FlexClone-Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.Letter</code>	Gibt den Laufwerkbuchstaben oder Mount-Punkt der Festplatte <i>N</i> einer Maschine an. Der Standardwert ist C. Um beispielsweise den Buchstaben D für die Festplatte 1 anzugeben, definieren Sie die benutzerdefinierte Eigenschaft als <code>VirtualMachine.Disk1.Letter</code> und geben Sie den Wert „D“ ein. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent gibt dieser Wert den Laufwerkbuchstaben oder Einhängepunkt an, unter dem die zusätzliche Festplatte <i>N</i> vom Gast-Agent im Gastbetriebssystem gemountet wird.
<code>VirtualMachine.Admin.CustomizeGuestOSDelay</code>	Gibt an, wie lange nach Abschluss der Anpassung gewartet werden soll, bevor die Anpassung des Gastbetriebssystems gestartet wird. Für diesen Wert ist das Format HH:MM:SS erforderlich. Wenn dieser Wert nicht festgelegt wird, wird der Standardwert von einer Minute (00:01:00) verwendet. Wenn Sie diese benutzerdefinierte Eigenschaft nicht angeben, schlägt die Bereitstellung möglicherweise fehl, falls die virtuelle Maschine neu gestartet wird, bevor Arbeitselemente des Gast-Agents abgeschlossen sind.
<code>VirtualMachine.Customize.WaitComplete</code>	Legen Sie diese Eigenschaft auf „True“ fest, um zu verhindern, dass der Bereitstellungsworkflow Arbeitselemente an den Gast-Agent sendet, bevor alle Anpassungen abgeschlossen sind. Legen Sie diese Eigenschaft auf „False“ fest, um die Erstellung von Arbeitselementen zu ermöglichen, bevor die Anpassung abgeschlossen ist.
<code>VirtualMachine.SoftwareN.ScriptPath</code>	Gibt den vollständigen Pfad zum Installationsskript einer Anwendung an. Bei dem Pfad muss es sich um einen gültigen absoluten Pfad wie er im Gastbetriebssystem angezeigt wird handeln und er muss den Namen der Skriptdatei enthalten. Sie können benutzerdefinierte Eigenschaftswerte als Parameter an das Skript übergeben, indem Sie <code>{CustomPropertyName}</code> in der Pfadzeichenfolge einfügen. Angenommen, Sie haben eine benutzerdefinierte Eigenschaft <code>ActivationKey</code> mit dem Wert 1234. In diesem Fall lautet der Skriptpfad <code>D:\InstallApp.bat -key {ActivationKey}</code> . Der Gast-Agent führt den Befehl <code>D:\InstallApp.bat -key 1234</code> aus. Ihre Skriptdatei kann dann so programmiert werden, dass dieser Wert akzeptiert und verwendet wird.

Benutzerdefinierte Eigenschaften für einfache Workflow-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für einfache Workflow-Blueprints bereitstellen können.

Tabelle 1-10. Benutzerdefinierte Eigenschaften für einfache Workflow-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.CDRom.Attach</code>	Legen Sie diese Eigenschaft auf „False“ fest, um die Maschine ohne CD-ROM-Gerät bereitzustellen. Der Standardwert ist „True“.
<code>VirtualMachine.Admin.ThinProvision</code>	Legt fest, ob Thin Provisioning auf ESX-Computing-Ressourcen verwendet wird. Festplatten-bereitstellung wird vom zugrunde liegenden Speicher abstrahiert. Legen Sie diese Eigenschaft auf „True“ fest, um Thin Provisioning zu verwenden. Legen Sie diese Eigenschaft auf „False“ fest, um die Standardbereitstellung zu verwenden. Diese Eigenschaft dient für die virtuelle Bereitstellung.
<code>VirtualMachine.DiskN.StorageReservationPolicy</code>	Gibt die Speicherreservierungsrichtlinie an, die für die Suche nach Speicher für die Festplatte <i>N</i> verwendet werden soll. Hiermit wird außerdem die benannte Speicherreservierungsrichtlinie einem Volume zugewiesen. Um diese Eigenschaft zu verwenden, ersetzen Sie die Volume-Nummer im Eigenschaftsnamen durch <i>N</i> und geben Sie den Namen einer Speicherreservierungsrichtlinie als Wert an. Diese Eigenschaft entspricht dem Namen der im Blueprint angegebenen Speicherreservierungsrichtlinie. Festplatten müssen sequenziell nummeriert werden. Diese Eigenschaft kann für alle virtuellen Reservierungen und vCloud-Reservierungen verwendet werden. Für physische, Amazon- oder OpenStack-Reservierungen kann diese Eigenschaft nicht verwendet werden.
<code>VirtualMachine.Storage.AllocationType</code>	<p>Speichert erfasste Gruppen in einem einzigen Datenspeicher. Eine verteilte Umgebung speichert Festplatten im Round Robin-Verfahren. Geben Sie einen der folgenden Werte an:</p> <ul style="list-style-type: none"> ■ Erfasst <p>Alle Festplatten zusammenhalten.</p> ■ Verteilt <p>Ermöglicht die Platzierung der Festplatten in jedem Datenspeicher oder Datenspeicher-Cluster, der in der Reservierung verfügbar ist.</p> <p>Ein Beispiel dafür, wie die <code>VirtualMachine.Storage.AllocationType</code>-Eigenschaft zum Erstellen von Datenspeicher-Clustern verwendet wird, finden Sie im Blog Keeping Multiple Disks Together (Mehrere Festplatten zusammenhalten).</p>
<code>VirtualMachine.Storage.Name</code>	Identifiziert den Speicherpfad, in dem sich die Maschine befindet. Der Standardwert ist der in der Reservierung angegebene Wert, der zum Bereitstellen der Maschine verwendet wurde.

Tabelle 1-10. Benutzerdefinierte Eigenschaften für einfache Workflow-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Storage.ReserveMemory</code>	<p>Legen Sie diese Eigenschaft für die Verwaltung der vSwap-Speicherzuteilung auf „True“ fest, um die Verfügbarkeit sicherzustellen und die Zuteilung in der Reservierung festzulegen. Die vSwap-Zuteilung wird beim Erstellen oder Neukonfigurieren einer virtuellen Maschine berücksichtigt. Die vSwap-Zuteilungsprüfung ist nur für vSphere-Endpoints verfügbar.</p> <hr/> <p>Hinweis Wenn Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.Storage.ReserveMemory</code> beim Erstellen oder Neukonfigurieren der Maschine in vRealize Automation nicht angeben, ist die Verfügbarkeit von Auslagerungsspeicherplatz nicht sichergestellt. Wenn Sie diese Eigenschaft für eine bereits bereitgestellte Maschine hinzufügen und die zugeteilte Reservierung voll ist, überschreitet der in der Reservierung zugeteilte Speicher möglicherweise den tatsächlich zugeteilten Speicher.</p>
<code>VMware.Hardware.Version</code>	<p>Gibt die VM-Hardwareversion an, die für vSphere-Einstellungen verwendet werden soll. Unterstützte Werte sind derzeit „vmx-04“, „vmx-07“, „vmx-08“, „vmx-09“ und „vmx-10“. Diese Eigenschaft betrifft VM-Erstellungs- und VM-Aktualisierungsworkflows und ist nur für einfache Workflow-Blueprints verfügbar.</p>

Benutzerdefinierte Eigenschaften für Linux Kickstart-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für Linux Kickstart-Blueprints bereitstellen können.

Bestimmte benutzerdefinierte Eigenschaften von vRealize Automation sind für die Verwendung mit Linux Kickstart-Blueprints erforderlich.

Tabelle 1-11. Erforderliche benutzerdefinierte Eigenschaften für Linux Kickstart-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VMware.VirtualCenter.OperatingSystem</code>	<p>Gibt die Version des vCenter Server-Gastbetriebssystems (<code>VirtualMachineGuestOsIdentifizier</code>) an, mit der vCenter Server die Maschine erstellt. Diese Betriebssystemversion muss mit der Betriebssystemversion übereinstimmen, die auf der bereitgestellten Maschine installiert werden soll. Administratoren können Eigenschaftsgruppen mithilfe mehrerer Eigenschaftensätze erstellen, wie beispielsweise <code>VMware[OS_Version]Properties</code>. Diese Eigenschaftensätze sind vordefiniert und enthalten die korrekten Werte für <code>VMware.VirtualCenter.OperatingSystem</code>. Diese Eigenschaft dient für die virtuelle Bereitstellung.</p> <p>Weitere Informationen hierzu finden Sie im Abschnitt zum Enumerationstyp <code>VirtualMachineGuestOsIdentifizier</code> in der vSphere API/SDK-Dokumentation. Eine Liste der aktuell akzeptierten Werte finden Sie in der vCenter Server-Dokumentation.</p>
<code>Image.ISO.Location</code>	<p>Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Speicherort des ISO-Images an, von dem aus gestartet werden soll. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.</p>
<code>Image.ISO.Name</code>	<p>Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Namen des ISO-Images an, von dem aus gestartet werden soll, wie beispielsweise „/ISO/Microsoft/WinPE.iso“. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.</p>
<code>Image.ISO.UserName</code>	<p>Gibt den Benutzernamen für den Zugriff auf die CIFS-Freigabe im Format <code>username@domain</code> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.</p>
<code>Image.ISO.Password</code>	<p>Gibt das Kennwort für die Eigenschaft <code>Image.ISO.UserName</code> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.</p>

Optionale benutzerdefinierte Eigenschaften sind verfügbar.

Tabelle 1-12. Optionale benutzerdefinierte Eigenschaften für Linux Kickstart-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.Admin.ThinProvision	Legt fest, ob Thin Provisioning auf ESX-Computing-Ressourcen verwendet wird. Festplatten-bereitstellung wird vom zugrunde liegenden Speicher abstrahiert. Legen Sie diese Eigenschaft auf „True“ fest, um Thin Provisioning zu verwenden. Legen Sie diese Eigenschaft auf „False“ fest, um die Standardbereitstellung zu verwenden. Diese Eigenschaft dient für die virtuelle Bereitstellung.
Machine.SSH	<p>Legen Sie diese Eigenschaft auf „True“ fest, um die Option Verbindungsherstellung mithilfe von SSH auf der vRealize Automation-Seite „Elemente“ für Linux-Maschinen zu aktivieren, die über diesen Blueprint bereitgestellt wurden. Wenn diese Eigenschaft auf „True“ festgelegt ist und die Option Verbinden via RDP oder SSH der Maschine im Blueprint aktiviert ist, wird für alle Linux-Maschinen, die über diesen Blueprint bereitgestellt werden, die Option Verbindungsherstellung mithilfe von SSH für berechnete Benutzer angezeigt.</p> <p>Gemäß der Option Verbindungsherstellung mithilfe von SSH muss der Browser über ein Plug-In verfügen, das SSH unterstützt, z. B. der SSH-Terminalclient „FireSSH“ für Mozilla Firefox und Google Chrome. Wenn das Plug-In vorhanden ist, wird bei Auswahl von Verbindungsherstellung mithilfe von SSH eine SSH-Konsole mit der Aufforderung angezeigt, Administratoranmeldedaten einzugeben.</p>

Benutzerdefinierte Eigenschaften für SCCM-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für SCCM-Blueprints bereitstellen können.

Bestimmte benutzerdefinierte Eigenschaften sind für die Verwendung mit SCCM-Blueprints erforderlich.

Tabelle 1-13. Erforderliche benutzerdefinierte Eigenschaften für SCCM-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
Image.ISO.Location	Bei den Werten für diese Eigenschaft wird die Groß-/ Kleinschreibung beachtet. Gibt den Speicherort des ISO-Images an, von dem aus gestartet werden soll. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.
Image.ISO.Name	Bei den Werten für diese Eigenschaft wird die Groß-/ Kleinschreibung beachtet. Gibt den Namen des ISO-Images an, von dem aus gestartet werden soll, wie beispielsweise „/ISO/Microsoft/WinPE.iso“. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.
Image.ISO.UserName	Gibt den Benutzernamen für den Zugriff auf die CIFS-Freigabe im Format <i>username@domain</i> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
Image.ISO.Password	Gibt das Kennwort für die Eigenschaft <code>Image.ISO.UserName</code> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
SCCM.Collection.Name	Gibt den Namen der SCCM-Kollektion an, die die Aufgabenabfolge für die Betriebssystembereitstellung enthält.
SCCM.Server.Name	Gibt den vollqualifizierten Domänennamen des SCCM-Servers an, auf dem die Kollektion gespeichert ist, wie beispielsweise „lab-sccm.lab.local“.
SCCM.Server.SiteCode	Gibt den Standortcode des SCCM-Servers an.
SCCM.Server.UserName	Gibt einen Benutzernamen mit Administratorzugriff auf den SCCM-Server an.
SCCM.Server.Password	Gibt das Kennwort für die Eigenschaft <code>SCCM.Server.UserName</code> an.

Bestimmte benutzerdefinierte Eigenschaften werden häufig mit SCCM-Blueprints verwendet.

Tabelle 1-14. Häufig verwendete benutzerdefinierte Eigenschaften für SCCM-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
SCCM.CustomVariable.Name	Gibt den Wert einer benutzerdefinierten Variablen an, wobei <i>Name</i> der Name einer benutzerdefinierten Variable ist, die für die SCCM-Aufgabenabfolge verfügbar gemacht werden soll, nachdem die bereitgestellte Maschine bei der SCCM-Kollektion registriert wurde. Dieser Wert wird durch die von Ihnen ausgewählte benutzerdefinierte Variable bestimmt. Wenn Ihre Integration dies erfordert, können Sie mithilfe von SCCM.RemoveCustomVariablePrefix das Präfix SCCM.CustomVariable. aus Ihrer benutzerdefinierten Variablen entfernen.
SCCM.RemoveCustomVariablePrefix	Legen Sie diese Eigenschaft auf <i>true</i> fest, um das Präfix SCCM.CustomVariable. aus benutzerdefinierten SCCM-Variablen zu entfernen, die Sie mithilfe der benutzerdefinierten Eigenschaft SCCM.CustomVariable.Name erstellt haben.

Benutzerdefinierte Eigenschaften für WIM-Blueprints

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für WIM-Blueprints bereitstellen können.

Bestimmte benutzerdefinierte Eigenschaften von vRealize Automation sind für WIM-Blueprints erforderlich.

Tabelle 1-15. Erforderliche benutzerdefinierte Eigenschaften für WIM-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
Image.ISO.Location	Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Speicherort des ISO-Images an, von dem aus gestartet werden soll. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.
Image.ISO.Name	Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Namen des ISO-Images an, von dem aus gestartet werden soll, wie beispielsweise „/ISO/Microsoft/WinPE.iso“. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.

Tabelle 1-15. Erforderliche benutzerdefinierte Eigenschaften für WIM-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>Image.ISO.UserName</code>	Gibt den Benutzernamen für den Zugriff auf die CIFS-Freigabe im Format <i>username@domain</i> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
<code>Image.ISO.Password</code>	Gibt das Kennwort für die Eigenschaft <code>Image.ISO.UserName</code> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
<code>Image.Network.Letter</code>	Gibt den Laufwerksbuchstaben an, dem der WIM-Image-Pfad auf der bereitgestellten Maschine zugeordnet wird. Der Standardwert ist „K“.
<code>Image.WIM.Path</code>	Gibt den UNC-Pfad zur WIM-Datei an, aus der bei der WIM-basierten Bereitstellung ein Image extrahiert wird. Das Pfadformat ist <code>\\server\share\$</code> , beispielsweise <code>\\lab-ad\dfs\$</code> .
<code>Image.WIM.Name</code>	Gibt den Namen der WIM-Datei an, deren Speicherort mit der Eigenschaft <code>Image.WIM.Path</code> festgelegt wird.
<code>Image.WIM.Index</code>	Gibt den verwendeten Index zum Extrahieren des korrekten Images aus der WIM-Datei an.
<code>Image.Network.User</code>	Gibt den Benutzernamen an, mit dem der WIM-Image-Pfad (<code>Image.WIM.Path</code>) einem Netzlaufwerk auf der bereitgestellten Maschine zugeordnet werden soll. Dies ist in der Regel ein Domänenkonto mit Zugriff auf die Netzwerkfreigabe.
<code>Image.Network.Password</code>	Gibt das Kennwort für die Eigenschaft <code>Image.Network.User</code> an.
<code>VirtualMachine.Admin.Owner</code>	Gibt den Benutzernamen des Maschinenbesitzers an.
<code>VMware.VirtualCenter.OperatingSystem</code>	Gibt die Version des vCenter Server-Gastbetriebssystems (<code>VirtualMachineGuestOsIdentifier</code>) an, mit der vCenter Server die Maschine erstellt. Diese Betriebssystemversion muss mit der Betriebssystemversion übereinstimmen, die auf der bereitgestellten Maschine installiert werden soll. Administratoren können Eigenschaftsgruppen mithilfe mehrerer Eigenschaftensätze erstellen, wie beispielsweise <code>VMware[OS_Version]Properties</code> . Diese Eigenschaftensätze sind vordefiniert und enthalten die korrekten Werte für <code>VMware.VirtualCenter.OperatingSystem</code> . Diese Eigenschaft dient für die virtuelle Bereitstellung. Weitere Informationen hierzu finden Sie im Abschnitt zum Enumerationstyp <code>VirtualMachineGuestOsIdentifier</code> in der vSphere API/SDK-Dokumentation. Eine Liste der aktuell akzeptierten Werte finden Sie in der vCenter Server-Dokumentation.

Optionale benutzerdefinierte Eigenschaften sind für WIM-Blueprints ebenfalls verfügbar.

Tabelle 1-16. Häufig verwendete benutzerdefinierte Eigenschaften für WIM-Blueprints

Benutzerdefinierte Eigenschaft	Beschreibung
<p><i>SysPrep.Section.Key</i></p> <ul style="list-style-type: none"> ■ <i>SysPrep.GuiUnattended.AdminPassword</i> ■ <i>SysPrep.GuiUnattended.EncryptedAdminPassword</i> ■ <i>SysPrep.GuiUnattended.TimeZone</i> 	<p>Gibt Informationen an, die der SysPrep-Antwortdatei auf Maschinen während der WinPE-Phase der Bereitstellung hinzugefügt werden sollen. Bereits in der SysPrep-Antwortdatei vorhandene Informationen werden von diesen benutzerdefinierten Eigenschaften überschrieben. <i>Section</i> steht für den Namen des Abschnitts der SysPrep-Antwortdatei, wie beispielsweise „GuiUnattended“ oder „UserData“. <i>Key</i> steht für einen Schlüsselnamen im Abschnitt. Um beispielsweise als Zeitzone einer bereitgestellten Maschine Westpazifik (Normalzeit) festzulegen, definieren Sie die benutzerdefinierte Eigenschaft <i>GuiUnattended.UserData.TimeZone</i> und legen Sie „275“ als Wert fest.</p> <p>Eine vollständige Aufstellung der Abschnitte, Schlüssel und akzeptierten Werte finden Sie in der Dokumentation zum Systemvorbereitungsdienstprogramm (SysPrep) für Windows.</p> <p>Die folgenden <i>Section.Key</i>-Kombinationen können für die WIM-basierte Bereitstellung angegeben werden:</p> <ul style="list-style-type: none"> ■ <i>GuiUnattended</i> <ul style="list-style-type: none"> ■ <i>AdminPassword</i> ■ <i>EncryptedAdminPassword</i> ■ <i>TimeZone</i> ■ <i>UserData</i> <ul style="list-style-type: none"> ■ <i>ProductKey</i> ■ <i>FullName</i> ■ <i>ComputerName</i> ■ <i>OrgName</i> ■ <i>Identification</i> <ul style="list-style-type: none"> ■ <i>DomainAdmin</i> ■ <i>DomainAdminPassword</i> ■ <i>JoinDomain</i> ■ <i>JoinWorkgroup</i>
<i>Sysprep.Identification.DomainAdmin</i>	Gibt einen Benutzernamen mit Administratorzugriff auf die Zieldomäne in Active Directory an. Schließen Sie die Benutzerdomäne nicht in die Anmeldedaten ein, die Sie an vCloud Director oder vCloud Air senden.
<i>Sysprep.Identification.DomainAdminPassword</i>	Gibt das Kennwort für die Eigenschaft <i>Sysprep.Identification.DomainAdmin</i> an.
<i>Sysprep.Identification.JoinDomain</i>	Gibt den Namen der Domäne an, der in Active Directory beigetreten werden soll.
<i>Sysprep.Identification.JoinWorkgroup</i>	Gibt den Namen der Arbeitsgruppe an, der beigetreten werden soll, wenn keine Domäne verwendet wird.

Tabelle 1-16. Häufig verwendete benutzerdefinierte Eigenschaften für WIM-Blueprints (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
SysPrep.UserData.ComputerName	Gibt einen Maschinennamen an, wie beispielsweise „lab-client005“.
SysPrep.UserData.FullName	Gibt den vollständigen Namen eines Benutzers an.
SysPrep.UserData.OrgName	Gibt den Organisationsnamen des Benutzers an.
SysPrep.UserData.ProductKey	Gibt den Windows-Product Key an.
VirtualMachine.Admin.ThinProvision	Legt fest, ob Thin Provisioning auf ESX-Computing-Ressourcen verwendet wird. Festplatten-bereitstellung wird vom zugrunde liegenden Speicher abstrahiert. Legen Sie diese Eigenschaft auf „True“ fest, um Thin Provisioning zu verwenden. Legen Sie diese Eigenschaft auf „False“ fest, um die Standardbereitstellung zu verwenden. Diese Eigenschaft dient für die virtuelle Bereitstellung.

Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Blueprints

Sie können bestimmte benutzerdefinierte Eigenschaften zu einer vCloud Air- oder vCloud Director-Maschinenkomponentendefinition in einem Blueprint hinzufügen.

Verwenden Sie für vSphere-Maschinenkomponenten mit zugeordnetem NSX die Einstellung für Netzwerk, Sicherheit und Lastausgleich auf der Benutzeroberfläche. Für Maschinenkomponenten, die nicht über eine Registerkarte **Netzwerk** oder **Sicherheit** verfügen, können Sie benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit wie z. B.

VirtualMachine.Network0.Name zur entsprechenden Registerkarte **Eigenschaften** in der Design-Arbeitsfläche hinzufügen. Die Eigenschaften von NSX-Netzwerk-, -Sicherheits- und -Lastausgleichsdiensten gelten nur für vSphere-Maschinen.

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche

Benutzerdefinierte Eigenschaft	Beschreibung
Sysprep.Identification.DomainAdmin	Gibt einen Benutzernamen mit Administratorzugriff auf die Zieldomäne in Active Directory an. Schließen Sie die Benutzerdomäne nicht in die Anmeldedaten ein, die Sie an vCloud Director oder vCloud Air senden.
Sysprep.Identification.DomainAdminPassword	Gibt das Kennwort für die Eigenschaft Sysprep.Identification.DomainAdmin an.
Sysprep.Identification.JoinDomain	Gibt den Namen der Domäne an, der in Active Directory beigetreten werden soll.

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.DiskN.IsFixed	<p>Deaktiviert die Bearbeitung einer bestimmten Festplatte bei der Neukonfiguration einer Maschine. Legen Sie diese Eigenschaft auf „True“ fest, um die Anzeige der Kapazitätsbearbeitungsoption für ein bestimmtes Volume zu deaktivieren. Für den Wert „True“ wird die Groß-/Kleinschreibung berücksichtigt. Der Wert <i>N</i> ist der 0-basierte Index der Festplatte.</p> <p>Alternativ können Sie die benutzerdefinierte Eigenschaft VirtualMachine.DiskN.IsFixed in der VirtualMachineProperties-Tabelle der Datenbank auf „True“ festlegen oder mithilfe der Repository-API einen URI-Wert angeben, wie beispielsweise <code>.../Repository/Data/ManagementModelEntities.svc/VirtualMachines(guid'60D93A8A-F541-4CE0-A6C6-78973AC0F1D2')/VirtualMachineProperties</code>.</p>
VirtualMachine.DiskN.StorageReservationPolicy	<p>Gibt die Speicherreservierungsrichtlinie an, die für die Suche nach Speicher für die Festplatte <i>N</i> verwendet werden soll. Hiermit wird außerdem die benannte Speicherreservierungsrichtlinie einem Volume zugewiesen. Um diese Eigenschaft zu verwenden, ersetzen Sie die Volume-Nummer im Eigenschaftsnamen durch <i>N</i> und geben Sie den Namen einer Speicherreservierungsrichtlinie als Wert an. Diese Eigenschaft entspricht dem Namen der im Blueprint angegebenen Speicherreservierungsrichtlinie. Festplatten müssen sequenziell nummeriert werden. Diese Eigenschaft kann für alle virtuellen Reservierungen und vCloud-Reservierungen verwendet werden. Für physische, Amazon- oder OpenStack-Reservierungen kann diese Eigenschaft nicht verwendet werden.</p>
VirtualMachine.EULA.AcceptAll	<p>Legen Sie diese Eigenschaft auf „True“ fest, um anzugeben, dass alle Lizenzvereinbarungen (EULAs) für die VM-Vorlagen der vCloud Air- oder vCloud Director-Endpoints während der Bereitstellung akzeptiert werden.</p>

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.NetworkN.Name	<p>Gibt den Namen des Netzwerks an, mit dem eine Verbindung hergestellt werden soll. Beispielsweise das Netzwerkgerät <i>N</i>, mit dem eine Maschine verbunden wird. Dies entspricht einer Netzwerkkarte (Network Interface Card, NIC).</p> <p>Standardmäßig wird ein Netzwerk aus den in der Reservierung verfügbaren Netzwerkpfaden zugewiesen, in denen die Maschine bereitgestellt wird. Siehe auch <code>VirtualMachine.NetworkN.AddressType</code>.</p> <p>Sie können sicherstellen, dass ein Netzwerkgerät mit einem bestimmten Netzwerk verbunden wird, indem Sie für diese Eigenschaft den Namen eines Netzwerks in einer verfügbaren Reservierung festlegen. Wenn Sie beispielsweise als Eigenschaften <i>N</i>= 0 und 1 festlegen, erhalten Sie zwei NICs und deren zugewiesenen Wert, vorausgesetzt das Netzwerk ist in der zugeordneten Reservierung ausgewählt.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p> <p>Ein Beispiel dafür, wie Sie diese benutzerdefinierte Eigenschaft verwenden können, um <code>VirtualMachine.Network0.Name</code> basierend auf der Auswahl des Konsumenten aus einer Liste der vordefinierten verfügbaren Netzwerke dynamisch festzulegen, finden Sie im Blog Adding a Network Selection Drop-Down in vRA 7 (Hinzufügen eines Dropdown-Menüs für die Netzwerkauswahl in vRA 7).</p> <p>Informationen hierzu finden Sie unter Benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit.</p>

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.NetworkN.AddressType</code>	<p>Gibt an, wie die Zuteilung der IP-Adresse an den Netzanbieter übermittelt wird, wobei „NetworkN“ die Netzwerknummer (beginnend bei 0) ist. Die folgenden Werte sind verfügbar:</p> <ul style="list-style-type: none"> ■ DHCP ■ Statisch ■ MANUAL (nur verfügbar für vCloud Air und vCloud Director) <p>Mit dieser Eigenschaft können Sie vCloud Air-, vCloud Director- und vSphere-Maschinenkomponenten im Blueprint konfigurieren. Weitere Informationen finden Sie auch unter <code>VirtualMachine.NetworkN.Name</code>. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>
<code>VirtualMachine.Reconfigure.DisableHotCpu</code>	<p>Legen Sie diese Eigenschaft auf „True“ fest, um anzugeben, dass mit der Option für die Maschinenneukonfiguration die angegebene Maschine neu gestartet wird. Standardmäßig wird mit der Option für die Maschinenneukonfiguration die Maschine nicht neu gestartet.</p> <p>Beim Hinzufügen von CPU, Arbeitsspeicher oder Speicher bei laufendem Betrieb schlägt die Option für die Maschinenneukonfiguration fehl und die Maschine wird nicht neu gestartet, außer die Einstellung <code>Hot Add</code> ist in vSphere für die Maschine oder Vorlage aktiviert. Sie können</p> <p><code>VirtualMachine.Reconfigure.DisableHotCpu=true</code> einer Maschinenkomponente in einem vRealize Automation-Blueprint hinzufügen, um die Einstellung <code>Hot Add</code> zu deaktivieren und den Neustart der Maschine unabhängig von der vSphere-Einstellung <code>Hot Add</code> zu erzwingen. Diese benutzerdefinierte Eigenschaft ist nur für Maschinentypen verfügbar, die die Hardwareneukonfiguration unterstützen, nämlich vSphere, vCloud Air und vCloud Director.</p>

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VCloud.Lease.Sync.TimeBufferMins	<p>Gibt einen ganzzahligen Schwellenwert für eine Computing-Ressource an, sodass die Lease-Synchronisierung zwischen vCloud Director und vRealize Automation nur für mit vCloud Director oder vCloud Air bereitgestellte Maschinen ausgeführt wird, die in vCloud Director oder vCloud Air in diesem Zeitraum ablaufen sollen. Falls ein Konflikt festgestellt wird, wird der Leasewert synchronisiert und an die in vRealize Automation definierte Leasedauer angepasst. Der Standardwert für VCloud.Lease.Sync.TimeBufferMins beträgt 720 Minuten (12 Stunden). Wenn VCloud.Lease.Sync.TimeBufferMins nicht vorhanden ist, wird der Standardwert verwendet. Wenn beispielsweise die Standardwerte verwendet werden, führt vRealize Automation den Workflow zur Lease-Synchronisierungsprüfung alle 45 Minuten aus, was dem Workflow-Standardwert entspricht. Nur die Leases von Maschinen, die innerhalb von 12 Stunden ablaufen sollen, werden geändert und an die in vRealize Automation definierte Leasedauer angepasst.</p>
VCloud.Owner.UseEndpointAccount	<p>Legen Sie diese Eigenschaft auf „True“ fest, um das Endpoint-Konto als vCloud Air- oder vCloud Director-Maschinenbesitzer für Bereitstellungs- und Importvorgänge zuzuweisen. Für Vorgänge zum Ändern des Besitzers wird der Besitzer nicht auf dem Endpoint geändert. Wenn diese Option nicht angegeben ist oder auf „False“ festgelegt ist, ist der vRealize Automation-Besitzer der Maschinenbesitzer.</p>
VCloud.Template.MakeIdenticalCopy	<p>Legen Sie diese Eigenschaft auf „True“ fest, um eine identische Kopie der vCloud Air- oder vCloud Director-Vorlage für die Maschinenbereitstellung zu klonen. Die Maschine wird als identische Kopie der Vorlage bereitgestellt. In der Vorlage angegebene Einstellungen, einschließlich des Speicherpfads, ersetzen im Blueprint angegebene Einstellungen. Die einzigen Änderungen gegenüber der Vorlage sind die Namen der geklonten Maschinen, die anhand des im Blueprint angegebenen Maschinenpräfixes generiert werden.</p> <p>vCloud Air- oder vCloud Director-Maschinen, die als identische Kopien bereitgestellt werden, können Netzwerke und Speicherprofile verwenden, die in der vRealize Automation-Reservierung nicht verfügbar sind. Um nicht berücksichtigte Reservierungszuteilungen zu vermeiden, sollten Sie sicherstellen, dass das in der Vorlage angegebene Speicherprofil oder Netzwerk in der Reservierung verfügbar ist.</p>

Tabelle 1-17. Benutzerdefinierte Eigenschaften für vCloud Air- und vCloud Director-Maschinenkomponenten in der Design-Arbeitsfläche (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VMware.SCSI.Sharing	<p>Gibt an, in welchem Modus der VMware-SCSI-Bus der Maschine gemeinsam verwendet wird. Mögliche Werte basieren auf dem ENUM-Wert <code>VirtualSCSISharing</code> und enthalten „noSharing“, „physicalSharing“ und „virtualSharing“.</p> <p>Die Eigenschaft <code>VMware.SCSI.Sharing</code> ist für die Verwendung mit dem <code>CloneWorkflow</code>-Bereitstellungsworkflow nicht verfügbar. Wenn Sie den <code>CloneWorkflow</code>-Bereitstellungsworkflow beim Konfigurieren der Maschinenkomponente auf der Design-Arbeitsfläche des Blueprints angeben, können Sie die Eigenschaft <code>VMware.SCSI.Sharing</code> nicht verwenden.</p>
VMware.SCSI.Type	<p>Für vCloud Air-, vCloud Director - oder vSphere-Maschinenkomponenten in Blueprints wird hiermit der SCSI-Maschinentyp mithilfe eines der folgenden Werte, bei denen die Groß-/Kleinschreibung beachtet wird, angegeben:</p> <ul style="list-style-type: none"> ■ <code>buslogic</code> <p>Verwenden Sie die BusLogic-Emulation für die virtuelle Festplatte.</p> ■ <code>lsilogic</code> <p>Verwenden Sie die LSILogic-Emulation für die virtuelle Festplatte (Standardwert).</p> ■ <code>lsilogicsas</code> <p>Verwenden Sie die LSILogic SAS 1068-Emulation für die virtuelle Festplatte.</p> ■ <code>pvscsi</code> <p>Verwenden Sie die Para-Virtualisierungs-Emulation für die virtuelle Festplatte.</p> ■ <code>Keine</code> <p>Verwenden Sie diese Option, wenn kein SCSI-Controller für diese Maschine vorhanden ist.</p> <p>Die Eigenschaft <code>VMware.SCSI.Type</code> ist für die Verwendung mit dem <code>CloneWorkflow</code>-Bereitstellungsworkflow nicht verfügbar. Wenn Sie den <code>CloneWorkflow</code>-Bereitstellungsworkflow beim Konfigurieren der Maschinenkomponente auf der Design-Arbeitsfläche des Blueprints angeben, können Sie die Eigenschaft <code>VMware.SCSI.Type</code> nicht verwenden.</p>

Benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit

Mit den benutzerdefinierten Eigenschaften von vRealize Automation für Netzwerke wird die Konfiguration für ein bestimmtes Netzwerkgerät auf einer Maschine festgelegt.

Verwenden Sie für vSphere-Maschinenkomponenten mit zugeordnetem NSX die Einstellung für Netzwerk, Sicherheit und Lastausgleich auf der Benutzeroberfläche. Für Maschinenkomponenten, die nicht über eine Registerkarte **Netzwerk** oder **Sicherheit** verfügen, können Sie benutzerdefinierte Eigenschaften für Netzwerk und Sicherheit wie z. B.

`VirtualMachine.Network0.Name` zur entsprechenden Registerkarte **Eigenschaften** in der Design-Arbeitsfläche hinzufügen. Die Eigenschaften von NSX-Netzwerk-, -Sicherheits- und -Lastausgleichsdiensten gelten nur für vSphere-Maschinen.

Netzwerkzuweisungen werden bei der Maschinenzuteilung vorgenommen. vRealize Automation ruft Netzwerkinformationen vom Blueprint ab. Wenn Sie mehrere Netzwerke zuweisen möchten, verwenden Sie die benutzerdefinierte Eigenschaft `VirtualMachine.NetworkN.Name` für Ihren Maschinen-Blueprint. Wenn Sie keine benutzerdefinierten Eigenschaften angeben, wird nur ein Netzwerk zugewiesen, das mithilfe eines Round Robin-Verfahrens in Verbindung mit der gewählten Reservierung ausgewählt wird.

Hinweis Mit Ausnahme der folgenden Eigenschaften gelten die in der Tabelle aufgeführten Eigenschaften nicht für Amazon Web Services:

- `agent.download.url`
 - `software.agent.service.url`
 - `software.ebs.url`
-

Hinweis Netzwerkspezifische benutzerdefinierte Eigenschaften, die auf bedarfsgesteuerte Netzwerke verweisen, werden nicht unterstützt. Sie können beispielsweise keine benutzerdefinierten Netzwerkeigenschaften wie `VirtualMachine.Network0.NetworkProfileName`, bedarfsgesteuerte NAT- und geroutete bedarfsgesteuerte Netzwerkkomponenten verwenden.

Standardmäßig wird mit der Eigenschaft `VirtualMachine.Network0.Name` ein Netzwerkgerät für eine Maschine konfiguriert. Mithilfe der benutzerdefinierten Eigenschaft `VirtualMachine.NetworkN.Name` können Sie zusätzliche Netzwerkgeräte konfigurieren, wobei *N* für die Netzwerknummer steht.

Netzwerkeigenschaften müssen fortlaufend durchnummeriert werden, beginnend mit 0. Wenn Sie beispielsweise benutzerdefinierte Eigenschaften nur für `VirtualMachine.Network0` und `VirtualMachine.Network2` angeben, werden die Eigenschaften für `VirtualMachine.Network2` ignoriert, da das vorausgehende Netzwerk, `VirtualMachine.Network1`, nicht angegeben wurde.

Obwohl die allgemeine Unterstützung für vCloud Networking and Security beendet wurde, sind die benutzerdefinierten VCNS-Eigenschaften nach wie vor zu NSX-Zwecken gültig. Weitere Informationen hierzu finden Sie im [Knowledgebase-Artikel 2144733](#).

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke

Benutzerdefinierte Eigenschaft	Beschreibung
<code>agent.download.url</code>	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für Ihre Software-Agent-Datei angegeben. Beispielsweise</p> <p><code>https://Private_IP:1443/software-service/resources/nobel-agent.jar</code></p> <p>Gibt die URL für den VPN-Agent in Ihrer Bereitstellung an. Das URL-Format lautet <code>https:// Private_IP:1443/software-service//resources/noble-agent.jar</code></p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.agent.service.url</code> und <code>software.ebs.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie die PAT- oder NAT-Übersetzung und die Portweiterleitung verwenden.</p>
<code>NSX.Edge.ApplianceSize</code>	<p>Gibt die zulässigen Größentypen der NSX-Edge-Appliance für die bereitgestellte Maschine oder die Bereitstellung an. Folgende Optionen stehen zur Verfügung:</p> <ul style="list-style-type: none"> ■ compact <p>Für kleine Bereitstellungen, POCs und Verwendung in einzelnen Diensten.</p> <ul style="list-style-type: none"> ■ CPU = 1 ■ RAM = 512 MB ■ Festplatte = 512 MB ■ large <p>Für kleine bis mittlere Bereitstellungen oder Bereitstellungen mit mehreren Mandanten.</p> <ul style="list-style-type: none"> ■ CPU = 2 ■ RAM = 1 GB ■ Festplatte = 512 MB ■ quadlarge <p>Für Bereitstellungen mit Equal-Cost Mlti-Path Routing (ECMP) mit hohem Durchsatz oder mit hoher Leistung und Firewall.</p> <ul style="list-style-type: none"> ■ CPU = 4 ■ RAM = 1 GB ■ Festplatte = 512 MB ■ xlarge <p>Für Bereitstellungen mit L7-Lastausgleich und dedizierte Core-Bereitstellungen.</p> <ul style="list-style-type: none"> ■ CPU = 6 ■ RAM = 8 GB ■ Festplatte = 4,5 GB (4 GB Swap) <p>Weitere Informationen finden Sie unter den Systemanforderungen für NSX.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
NSX.Edge.HighAvailability	<p>Wenn der Wert auf „True“ (NSX.Edge.HighAvailability=true) festgelegt ist, wird der Hochverfügbarkeitsmodus auf der NSX Edge-Maschine aktiviert, die über den Blueprint bereitgestellt wird.</p> <p>Bei Verwendung mit NSX.Edge.HighAvailability.PortGroup=port_group_name können Sie mit dieser Eigenschaft einen NSX Edge während der Blueprint-Erstellung konfigurieren.</p> <p>Sie können diese Eigenschaft zu einer NSX-Komponente für den Lastausgleichsdienst im vRealize Automation-Blueprint oder direkt zum vRealize Automation-Blueprint hinzufügen.</p> <p>Muss in Verbindung mit NSX.Edge.HighAvailability.PortGroup=port_group_name verwendet werden.</p>
NSX.Edge.HighAvailability.PortGroup	<p>Erstellt eine mit dem angegebenen Namen der Portgruppe verbundene interne Schnittstelle oder interne vNIC, z. B. NSX.Edge.HighAvailability.PortGroup=VM Network, wobei VM Network eine Portgruppe mit einem Distributed Switch mit Hochverfügbarkeit (vLAN-gestützt) oder eine NSX-Portgruppe mit einem logischen Switch ist. NSX im Hochverfügbarkeitsmodus erfordert mindestens eine interne Schnittstelle oder vNIC.</p> <p>Bei Verwendung mit NSX.Edge.HighAvailability=true können Sie mit dieser Eigenschaft Hochverfügbarkeit für einen NSX Edge während der Blueprint-Erstellung konfigurieren.</p> <p>Wenn Sie bei aktivierter Hochverfügbarkeit One-Arm-Lastausgleich verwenden, müssen Sie für die Hochverfügbarkeit eine getrennte Portgruppe angeben.</p> <p>Hinweis Das angegebene Portgruppennetzwerk kann nicht Mitglied des Reservierungspools sein, da die Verwendung der Portgruppe durch die Eigenschaft mit der regulären Nutzung der Portgruppe durch die Bereitstellung konfliktiert, was den folgenden Fehler hervorruft:</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>Portgroup must be unique within an Edge...</p> </div> <p>Muss in Verbindung mit NSX.Edge.HighAvailability=true verwendet werden.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
NSX.Validation.Disable.Single.Edge.Uplink	<p>Wenn der Wert auf „true“ festgelegt ist, ist die NSX-Validierung deaktiviert, die folgende Bedingungen überprüft:</p> <ul style="list-style-type: none"> ■ Alle bedarfsgesteuerten NAT-Netzwerke auf der Blueprint-Quelle im gleichen externen Netzwerk. ■ Alle weitergeleiteten bedarfsgesteuerten Netzwerke auf dem Blueprint, die das gleiche externe Netzwerk wie die VIP-Quelle des Lastausgleichs verwenden. ■ Alle bedarfsgesteuerten Lastausgleichskomponenten auf dem Blueprint haben VIPs auf dem gleichen externen Netzwerk oder bedarfsgesteuerte Netzwerke, die vom gleichen externen Netzwerk gesichert werden. <p>Wenn diese Validierungsprüfung deaktiviert ist, kann eine Bereitstellung zwar erfolgreich sein, der Zugriff auf einige Netzwerkkomponenten ist u. U. jedoch nicht möglich.</p> <p>Falls nicht vorhanden oder auf „false“ gesetzt, ist die Validierung aktiviert (Standard).</p> <p>Ein einzelner NSX Edge unterstützt nur ein externes Netzwerk als dessen Uplink-Netzwerk. Mehrere IP-Adressen von demselben externen Netzwerk werden unterstützt. Ein Blueprint kann eine beliebige Anzahl von externen oder bedarfsgesteuerten Netzwerkkomponenten enthalten; NSX unterstützt jedoch nur ein externes Netzwerk als Uplink-Netzwerk.</p> <p>Diese Eigenschaft kann nur auf der Blueprint-Ebene angegeben werden. Sie kann nicht für eine Komponente auf der Blueprint-Arbeitsfläche angegeben werden.</p>
NSX.Validation.Disable.Blueprint.NSXT	<p>Wenn der Wert auf „true“ festgelegt ist, wird die gesamte NSX-T-Validierung für die Blueprint-Aktion Fertig stellen deaktiviert.</p> <p>Wenn nicht vorhanden oder wenn der Wert auf „false“ festgelegt ist, ist die NSX-T-Validierungsprüfung aktiviert (Standard).</p> <p>Wenn beispielsweise im Blueprint überlappende Subnetze vorhanden sind, wird eine Fehlermeldung angezeigt, wenn Sie im Blueprint auf Fertig stellen klicken, und die Überlappung verhindert, dass der Blueprint fertig gestellt wird, obwohl Sie ihn speichern können. Wenn Sie den Blueprint fertig stellen möchten, können Sie unter Verwendung der Seite Blueprint-Eigenschaften <code>NSX.Validation.Disable.Blueprint.NSXT</code> hinzufügen und anschließend den Blueprint fertig stellen.</p> <p>Die Eigenschaft deaktiviert nur NSX-T-Validierungen für die Blueprint-Aktion Fertig stellen.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>software.agent.service.url</code>	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für die vRealize Automation-Softwaredienst-API angegeben. Beispielsweise <code>https://Private_IP:1443/software-service/api</code>.</p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.ebs.url</code> und <code>agent.download.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie PAT oder NAT und die Portweiterleitung verwenden.</p>
<code>software.ebs.url</code>	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für den vRealize Automation-Ereignisbrokerdienst angegeben. Beispielsweise <code>https://Private_IP:1443/event-broker-service/api</code>.</p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.agent.service.url</code> und <code>agent.download.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie PAT oder NAT und die Portweiterleitung verwenden.</p>
<code>VirtualMachine.NetworkN.Address</code>	<p>Gibt die IP-Adresse des Netzwerkgeräts <i>N</i> in einer mit einer statischen IP-Adresse bereitgestellten Maschine an.</p> <p>Informationen für Amazon finden Sie unter <code>Amazon.elasticIpAddress.ipAddress</code>.</p>
<code>VirtualMachine.NetworkN.MacAddressType</code>	<p>Gibt an, ob die MAC-Adresse des Netzwerkgeräts <i>N</i> generiert wird („generated“) oder benutzerdefiniert („static“) ist. Diese Eigenschaft ist für das Klonen verfügbar. Der Standardwert lautet „generated“. Mit dem Wert „static“ müssen Sie auch <code>VirtualMachine.NetworkN.MacAddress</code> verwenden, um die MAC-Adresse anzugeben.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.NetworkN.MacAddress	<p>Gibt die MAC-Adresse des Netzwerkgeräts <i>N</i> an. Diese Eigenschaft ist für das Klonen verfügbar.</p> <p>Wenn <code>VirtualMachine.NetworkN.MacAddressType</code> den Wert „generated“ hat, enthält diese Eigenschaft die generierte Adresse.</p> <p>Wenn <code>VirtualMachine.NetworkN.MacAddressType</code> den Wert „static“ hat, enthält diese Eigenschaft die MAC-Adresse. Für virtuelle Maschinen, die auf ESX Server-Hosts bereitgestellt werden, muss die Adresse innerhalb des von VMware angegebenen Bereichs liegen. Weitere Informationen finden Sie in der vSphere-Dokumentation.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.NetworkN.Name	<p>Gibt den Namen des Netzwerks an, mit dem eine Verbindung hergestellt werden soll. Beispielsweise das Netzwerkgerät <i>N</i>, mit dem eine Maschine verbunden wird. Dies entspricht einer Netzwerkkarte (Network Interface Card, NIC).</p> <p>Standardmäßig wird ein Netzwerk aus den in der Reservierung verfügbaren Netzwerkpfaden zugewiesen, in denen die Maschine bereitgestellt wird. Siehe auch <code>VirtualMachine.NetworkN.AddressType</code>.</p> <p>Sie können sicherstellen, dass ein Netzwerkgerät mit einem bestimmten Netzwerk verbunden wird, indem Sie für diese Eigenschaft den Namen eines Netzwerks in einer verfügbaren Reservierung festlegen. Wenn Sie beispielsweise als Eigenschaften <i>N</i>= 0 und 1 festlegen, erhalten Sie zwei NICs und deren zugewiesenen Wert, vorausgesetzt das Netzwerk ist in der zugeordneten Reservierung ausgewählt.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p> <p>Ein Beispiel dafür, wie Sie diese benutzerdefinierte Eigenschaft verwenden können, um <code>VirtualMachine.Network0.Name</code> basierend auf der Auswahl des Konsumenten aus einer Liste der vordefinierten verfügbaren Netzwerke dynamisch festzulegen, finden Sie im Blog Adding a Network Selection Drop-Down in vRA 7 (Hinzufügen eines Dropdown-Menüs für die Netzwerkauswahl in vRA 7).</p>
VirtualMachine.NetworkN.PortID	<p>Gibt die für das Netzwerkgerät <i>N</i> zu verwendende Port-ID an, wenn eine dvPort-Gruppe mit einem vSphere Distributed Switch verwendet wird.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.NetworkN.NetworkProfileName	<p data-bbox="810 268 1426 449">Gibt den Namen eines Netzwerkprofils an, aus dem dem Netzwerkgerät <i>N</i> eine statische IP-Adresse zugewiesen werden soll oder aus dem der statische IP-Adressbereich bezogen werden soll, der dem Netzwerkgerät <i>N</i> einer geklonten Maschine zugewiesen werden kann. Dabei steht <i>N=0</i> für das erste Gerät, 1 für das zweite Gerät usw.</p> <p data-bbox="810 466 1426 583">Das Netzwerkprofil, auf das die Eigenschaft verweist, wird zum Zuweisen einer IP-Adresse verwendet. Die Eigenschaft bestimmt das Netzwerk, an das die Maschine angeschlossen ist, basierend auf der Reservierung.</p> <p data-bbox="810 600 1426 718">Die Änderung dieses Eigenschaftswerts hat nach der Zuweisung des Netzwerks keine Auswirkung auf die erwarteten IP-Adresswerte für die angegebenen Maschinen.</p> <p data-bbox="810 735 1426 877">Mit der WIM-basierten Bereitstellung für virtuelle Maschinen können Sie mithilfe dieser Eigenschaft ein Netzwerkprofil und eine Netzwerkschnittstelle angeben. Sie können aber auch den Abschnitt „Netzwerk“ der Seite „Virtuelle Reservierung“ verwenden.</p> <p data-bbox="810 894 1426 982">Die folgenden Attribute des Netzwerkprofils sind für die Zuweisung von statischen IP-Adressen in einem Klon-Blueprint verfügbar:</p> <ul data-bbox="810 999 1426 1281" style="list-style-type: none"> ■ VirtualMachine.NetworkN.SubnetMask ■ VirtualMachine.NetworkN.Gateway ■ VirtualMachine.NetworkN.PrimaryDns ■ VirtualMachine.NetworkN.SecondaryDns ■ VirtualMachine.NetworkN.PrimaryWins ■ VirtualMachine.NetworkN.SecondaryWins ■ VirtualMachine.NetworkN.DnsSuffix ■ VirtualMachine.NetworkN.DnsSearchSuffixes <p data-bbox="810 1297 1426 1575">Die benutzerdefinierten Eigenschaften VirtualMachine.Network<i>N</i> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p> <p data-bbox="810 1591 1426 1801">Sie können diese benutzerdefinierte Eigenschaft nicht verwenden, um einen Profilnamen für eine bedarfsgesteuerte NAT oder ein bedarfsgesteuertes geroutetes Netzwerk zu definieren. Da Profilnamen für bedarfsgesteuerte Netzwerke bei Zuteilungszeit generiert werden (bei der Bereitstellung), sind deren Namen beim Erstellen oder Bearbeiten des Blueprints unbekannt. Um</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<ul style="list-style-type: none"> ■ VirtualMachine.NetworkN.SubnetMask ■ VirtualMachine.NetworkN.Gateway ■ VirtualMachine.NetworkN.PrimaryDns ■ VirtualMachine.NetworkN.SecondaryDns ■ VirtualMachine.NetworkN.PrimaryWins ■ VirtualMachine.NetworkN.SecondaryWins ■ VirtualMachine.NetworkN.DnsSuffix ■ VirtualMachine.NetworkN.DnsSearchSuffixes 	<p>Informationen von bedarfsgesteuerten Netzwerken von NSX anzugeben, verwenden Sie die entsprechende Netzwerkkomponente in der Blueprint-Design-Arbeitsfläche für Ihre vSphere-Maschinenkomponenten.</p> <p>Konfiguriert Attribute des in <code>VirtualMachine.NetworkN.NetworkProfileName</code> angegebenen Netzwerkprofils.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p>
<code>VCNS.LoadBalancerEdgePool.Names.name</code>	<p>Gibt die NSX-Lastausgleichspools an, denen die virtuelle Maschine während der Bereitstellung zugewiesen wird. Die virtuelle Maschine wird allen Dienstports von allen angegebenen Pools zugewiesen. Bei dem Wert handelt es sich um einen <i>Edge/Pool</i>-Namen oder eine durch Kommas getrennte Liste von <i>Edge/Pool</i>-Namen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt.</p> <p>Hinweis Sie können die IP-Adresse einer Maschine einem vorhandenen Lastausgleichsdienst mithilfe der benutzerdefinierten Eigenschaft <code>VCNS.LoadBalancerEdgePool.Names</code> hinzufügen. vRealize Automation und NSX verwenden das erste Element des angegebenen Edge-Lastausgleichsdienst-Pools, um den Port des neuen Elements zu ermitteln und die Porteinstellungen zu überwachen. Für NSX 6.2 muss jedoch die Porteinstellung für das Element nicht angegeben werden. Um Bereitstellungsfehler zu vermeiden, wenn <code>VCNS.LoadBalancerEdgePool.Names</code> mit NSX 6.2 zum Hinzufügen einer Maschine zu einem bestehenden Pool verwendet wird, müssen Sie einen Portwert für das erste Element des Lastausgleichsdienst-Pools in NSX angeben.</p> <p>Durch Anfügen eines Namens können Sie mehrere Versionen einer benutzerdefinierten Eigenschaft erstellen. Beispielsweise werden mit den folgenden Eigenschaften Lastausgleichspools aufgelistet, die zur allgemeinen Verwendung und für Maschinen mit hohen, mäßigen und niedrigen Leistungsanforderungen eingerichtet sind:</p> <ul style="list-style-type: none"> ■ <code>VCNS.LoadBalancerEdgePool.Names</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.moderate</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.high</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.low</code>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VCNS.SecurityGroup.Names.name	<p>Gibt die NSX-Sicherheitsgruppe(n) an, der bzw. denen die virtuelle Maschine während der Bereitstellung zugewiesen wird. Bei dem Wert handelt es sich um einen Sicherheitsgruppennamen oder eine durch Kommas getrennte Liste von Sicherheitsgruppennamen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt.</p> <p>Durch Anfügen eines Namens können Sie mehrere Versionen der Eigenschaft erstellen, die separat oder kombiniert verwendet werden können. Beispielsweise können mit den folgenden Eigenschaften Sicherheitsgruppen aufgelistet werden, die zur allgemeinen Verwendung, für die Vertriebsmitarbeiter und für den Support gedacht sind:</p> <ul style="list-style-type: none"> ■ VCNS.SecurityGroup.Names ■ VCNS.SecurityGroup.Names.sales ■ VCNS.SecurityGroup.Names.support

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VCNS.SecurityTag.Names.name	<p>Gibt das NSX-Sicherheitstag bzw. die -Sicherheitstags an, dem bzw. denen die virtuelle Maschine während der Bereitstellung zugeordnet wird. Bei dem Wert handelt es sich um einen Sicherheits-Tag-Namen oder eine durch Kommas getrennte Liste von Sicherheits-Tag-Namen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt. Durch Anfügen eines Namens können Sie mehrere Versionen der Eigenschaft erstellen, die separat oder kombiniert verwendet werden können. Beispielsweise können mit den folgenden Eigenschaften Sicherheits-Tags aufgelistet werden, die zur allgemeinen Verwendung, für die Vertriebsmitarbeiter und für den Support gedacht sind:</p> <ul style="list-style-type: none"> ■ VCNS.SecurityTag.Names ■ VCNS.SecurityTag.Names.sales ■ VCNS.SecurityTag.Names.support
VMware.Endpoint.NSX.HideDiscoveredSecurityObjects	<p>Legen Sie die Option auf „true“ fest, um neu erkannte Sicherheitsobjekte im aktiven Mandanten für die NSX-Endpoints zu verbergen, mit denen die Sicherheitsobjekte verbunden sind. Anderenfalls stehen alle neuen Sicherheitsobjekte nach der Datenerfassung in allen Mandanten zur Verfügung, vorausgesetzt, dass das Sicherheitsobjekt für einen Endpoint gilt, in dem Sie über eine Reservierung verfügen. Mit dieser Option können Sie vermeiden, dass Benutzer auf Sicherheitsobjekte zugreifen, wenn Sie diese Objekte einem einzelnen Mandanten zuweisen oder für alle Mandanten maskieren möchten. Legen Sie die Option auf „false“ fest, um zu „Global“ zurück zu wechseln, wodurch alle neuen Sicherheitsobjekte nach der Datenerfassung in allen Mandanten zur Verfügung stehen, vorausgesetzt, dass das Sicherheitsobjekt für einen Endpoint gilt, in dem Sie eine Reservierung besitzen. Damit dies wirksam wird, fügt der Fabric-Administrator die benutzerdefinierte Eigenschaft VMware.Endpoint.NSX.HideDiscoveredSecurityObjects dem zugehörigen NSX-Endpoint hinzu, der mit einem vSphere-Endpoint verknüpft ist. Die Einstellung gilt für die nächste Erfassung von Bestandslistendaten. Vorhandene Sicherheitsobjekte bleiben unverändert. Zum Ändern der Mandanteneinstellung eines Sicherheitsobjekts, für das bereits Daten erfasst wurden, wie z. B. vorhandene Sicherheitsobjekte nach dem Upgrade auf die aktuelle Version von vRealize Automation, können Sie die Einstellung „Mandanten-ID“ des Sicherheitsobjekts</p>

Tabelle 1-18. Konfiguration benutzerdefinierter Eigenschaften für Netzwerke (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
	<p>programmgesteuert mithilfe der REST API von vRealize Automation oder mithilfe von vRealize CloudClient bearbeiten. Die verfügbaren Einstellungen unter „Mandanten-ID“ für den NSX-Endpoint lauten wie folgt:</p> <ul style="list-style-type: none"> ■ "<code><global></code>" – das Sicherheitsobjekt steht allen Mandanten zur Verfügung. Dies ist die Standardeinstellung für vorhandene Sicherheitsobjekte nach dem Upgrade auf diese Version und für alle neuen Sicherheitsobjekte, die Sie erstellen. ■ "<code><unscoped></code>" – das Sicherheitsobjekt steht keinem Mandanten zur Verfügung. Nur der Systemadministrator kann auf das Sicherheitsobjekt zugreifen. Diese Einstellung eignet sich für die Definition von Sicherheitsobjekten, die irgendwann einem bestimmten Mandanten zugewiesen werden sollen. ■ "<code>Tenant_id_name</code>" – das Sicherheitsobjekt steht nur einem einzelnen benannten Mandanten zur Verfügung.

Benutzerdefinierte Eigenschaften und Eigenschaftsgruppen für Container

Sie können einer Containerkomponente in einem vRealize Automation-Blueprint vordefinierte Eigenschaftsgruppen hinzufügen. Wenn Maschinen, die diese Eigenschaften enthalten, mithilfe eines Blueprints bereitgestellt werden, wird die bereitgestellte Maschine als Docker Container-Hostmaschine registriert.

Container für vRealize Automation stellt die folgenden beiden Eigenschaftsgruppen von containerspezifischen benutzerdefinierten Eigenschaften bereit. Wenn Sie einem Blueprint eine Containerkomponente hinzufügen, können Sie diese Eigenschaftsgruppen zum Container hinzufügen, um bereitgestellte Maschinen als Container-Hosts zu registrieren.

- Container-Hosteigenschaften mit Zertifikatsauthentifizierung
- Container-Hosteigenschaften mit Benutzer-/Kennwortauthentifizierung

Diese Eigenschaftsgruppen sind in vRealize Automation sichtbar, wenn Sie **Verwaltung > Eigenschaftsdiktionär > Eigenschaftsgruppen** wählen.

Da Eigenschaftsgruppen von allen Mandanten gemeinsam genutzt werden, ziehen Sie das Klonen und Anpassen Ihrer Eigenschaften in Betracht, wenn Sie in einer Umgebung mit mehreren Mandanten arbeiten. Wenn Sie die Eigenschaftsgruppen und Eigenschaften in den Gruppen eindeutig benennen, können Sie sie bearbeiten, um benutzerdefinierte Werte zur Verwendung in einem bestimmten Mandanten zu definieren.

Die am häufigsten verwendeten Eigenschaften sind `Container.Auth.PublicKey` und `Container.Auth.PrivateKey`. Mit diesen stellt der Container-Administrator das Clientzertifikat zum Authentifizieren mit dem Container-Host bereit.

Tabelle 1-19. ContainerBenutzerdefinierte Eigenschaften

Eigenschaft	Beschreibung
<code>containers.ipam.driver</code>	Nur für die Verwendung mit Containern. Gibt den IPAM-Treiber an, der zu verwenden ist, wenn eine Container-Netzwerkkomponente zu einem Blueprint hinzugefügt wird. Welche Werte unterstützt werden, hängt von den Treibern ab, die in der Container-Hostumgebung installiert sind, in der sie verwendet werden. Ein unterstützter Wert wäre z. B. <code>infoblox</code> oder <code>calico</code> , je nachdem, welche IPAM-Plug-Ins auf dem Container-Host installiert sind.
<code>containers.network.driver</code>	Nur für die Verwendung mit Containern. Gibt den Netzwerktreiber an, der zu verwenden ist, wenn eine Container-Netzwerkkomponente zu einem Blueprint hinzugefügt wird. Welche Werte unterstützt werden, hängt von den Treibern ab, die in der Container-Hostumgebung installiert sind, in der sie verwendet werden. Standardmäßig gehören zu den von Docker bereitgestellten Netzwerktreibern Bridge-, Overlay- und Macvlan-Treiber, wobei bei den von Virtual Container Host (VCH) bereitgestellten Netzwerktreibern der Bridge-Treiber enthalten ist. Netzwerktreiber von Drittanbietern, wie z. B. <code>weave</code> und <code>calico</code> , stehen möglicherweise ebenfalls zur Verfügung, je nachdem, welche Netzwerk-Plug-Ins auf dem Container-Host installiert sind.
<code>Container</code>	Nur für die Verwendung mit Containern. Der Standardwert ist <code>App.Docker</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.
<code>Container.Auth.User</code>	Nur für die Verwendung mit Containern. Gibt den Benutzernamen für das Herstellen einer Verbindung zum Container-Host an.
<code>Container.Auth.Password</code>	Nur für die Verwendung mit Containern. Gibt entweder das Kennwort für den Benutzernamen oder für den zu verwendenden öffentlichen bzw. privaten Schlüssel an. Verschlüsselter Eigenschaftswert wird unterstützt.
<code>Container.Auth.PublicKey</code>	Nur für die Verwendung mit Containern. Gibt den öffentlichen Schlüssel für das Herstellen einer Verbindung zum Container-Host an.
<code>Container.Auth.PrivateKey</code>	Nur für die Verwendung mit Containern. Gibt den privaten Schlüssel für das Herstellen einer Verbindung zum Container-Host an. Verschlüsselter Eigenschaftswert wird unterstützt.
<code>Container.Connection.Protocol</code>	Nur für die Verwendung mit Containern. Gibt das Kommunikationsprotokoll an. Der Standardwert ist <code>API</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.
<code>Container.Connection.Scheme</code>	Nur für die Verwendung mit Containern. Gibt das Kommunikationsschema an. Die Standardeinstellung ist <code>https</code> .

Tabelle 1-19. ContainerBenutzerdefinierte Eigenschaften (Fortsetzung)

Eigenschaft	Beschreibung
Container.Connection.Port	Nur für die Verwendung mit Containern. Gibt den Container-Verbindungs-Port an. Die Standardeinstellung ist 2376.
Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineActivated	Nur für die Verwendung mit Containern. Gibt die Event-Broker-Eigenschaft an, um alle Container-Eigenschaften freizulegen, und wird für das Registrieren eines bereitgestellten Hosts verwendet. Der Standardwert ist Container* und erforderlich. Ändern Sie diese Eigenschaft nicht.
Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.Disposing	Nur für die Verwendung mit Containern. Gibt die Event-Broker-Eigenschaft an, um alle der oben genannten Container-Eigenschaften freizulegen, und wird für das Aufheben der Registrierung eines bereitgestellten Hosts verwendet. Der Standardwert ist Container* und erforderlich. Ändern Sie diese Eigenschaft nicht.

Benutzerdefinierte Eigenschaften für die PXE-Bereitstellung

PXE ist die einzige Bereitstellungsmethode, die für Cisco UCS Manager unterstützt wird. Mit dem Network Bootstrap-Programm können Sie zusammen mit benutzerdefinierten Eigenschaften von vRealize Automation die WIM-, SCCM- oder Linux Kickstart-Bereitstellung starten. Mithilfe benutzerdefinierter Eigenschaften können Sie auch Ihre eigenen PowerShell-Skripts aufrufen. Für die Linux Kickstart-Bereitstellung sind keine benutzerdefinierten Eigenschaften erforderlich.

Benutzerdefinierte Eigenschaften für die Bereitstellung mit PowerShell-Skripts

Mit diesen Eigenschaften können Sie PowerShell-Skripts aufrufen.

Tabelle 1-20. Benutzerdefinierte Eigenschaften zum Aufrufen von PowerShell-Skripts

Benutzerdefinierte Eigenschaft	Beschreibung
Pxe.Setup.ScriptName	Gibt ein benutzerdefiniertes EPI PowerShell-Skript an, das auf der Maschine ausgeführt werden soll, bevor sie vom PXE Network Bootstrap-Programm gestartet wird. Dieser Wert ist der Name, der dem Skript beim Hochladen in den Model Manager zugewiesen ist, wie beispielsweise setup.ps1.
Pxe.Clean.ScriptName	Gibt den Namen eines im vRealize Automation Model Manager installierten EPI PowerShell-Skripts an, das auf der Maschine nach deren Bereitstellung ausgeführt werden soll. Dieser Wert ist der Name, der dem Skript beim Hochladen in den Model Manager zugewiesen ist, wie beispielsweise clean.ps1.

Benutzerdefinierte Eigenschaften für die PXE- und SCCM-Bereitstellung

Diese Eigenschaften können Sie für die PXE- und SCCM-Bereitstellung verwenden.

Tabelle 1-21. Benutzerdefinierte Eigenschaften für die PXE- und SCCM-Bereitstellung

Benutzerdefinierte Eigenschaft	Beschreibung
SCCM.Collection.Name	Gibt den Namen der SCCM-Kollektion an, die die Aufgabenabfolge für die Betriebssystembereitstellung enthält.
SCCM.Server.Name	Gibt den vollqualifizierten Domännennamen des SCCM-Servers an, auf dem die Kollektion gespeichert ist, wie beispielsweise „lab-sccm.lab.local“.
SCCM.Server.SiteCode	Gibt den Standortcode des SCCM-Servers an.
SCCM.Server.UserName	Gibt einen Benutzernamen mit Administratorzugriff auf den SCCM-Server an.
SCCM.Server.Password	Gibt das Kennwort für die Eigenschaft SCCM.Server.UserName an.
SCCM.CustomVariable.	Gibt den Wert einer benutzerdefinierten Variablen an, wobei <i>Name</i> der Name einer benutzerdefinierten Variable ist, die für die SCCM-Aufgabenabfolge verfügbar gemacht werden soll, nachdem die bereitgestellte Maschine bei der SCCM-Kollektion registriert wurde. Dieser Wert wird durch die von Ihnen ausgewählte benutzerdefinierte Variable bestimmt. Wenn Ihre Integration dies erfordert, können Sie mithilfe von SCCM.RemoveCustomVariablePrefix das Präfix SCCM.CustomVariable. aus Ihrer benutzerdefinierten Variablen entfernen.

Benutzerdefinierte Eigenschaften für die PXE- und WIM-Bereitstellung

Diese Eigenschaften können Sie für die PXE- und WIM-Bereitstellung verwenden.

Tabelle 1-22. Benutzerdefinierte Eigenschaften für die PXE- und WIM-Bereitstellung

Benutzerdefinierte Eigenschaft	Beschreibung
Image.Network.Letter	Gibt den Laufwerksbuchstaben an, dem der WIM-Image-Pfad auf der bereitgestellten Maschine zugeordnet wird. Der Standardwert ist „K“.
Image.WIM.Path	Gibt den UNC-Pfad zur WIM-Datei an, aus der bei der WIM-basierten Bereitstellung ein Image extrahiert wird. Das Pfadformat ist \\server\share\$, beispielsweise \\lab-ad\dfs\$.
Image.WIM.Name	Gibt den Namen der WIM-Datei an, deren Speicherort mit der Eigenschaft Image.WIM.Path festgelegt wird.
Image.WIM.Index	Gibt den verwendeten Index zum Extrahieren des korrekten Images aus der WIM-Datei an.
Image.Network.User	Gibt den Benutzernamen an, mit dem der WIM-Image-Pfad (Image.WIM.Path) einem Netzlaufwerk auf der bereitgestellten Maschine zugeordnet werden soll. Dies ist in der Regel ein Domänenkonto mit Zugriff auf die Netzwerkfreigabe.

Tabelle 1-22. Benutzerdefinierte Eigenschaften für die PXE- und WIM-Bereitstellung (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
Image.Network.Password	Gibt das Kennwort für die Eigenschaft Image.Network.User an.
SysPrep.Section.Key ■ SysPrep.GuiUnattended.AdminPassword ■ SysPrep.GuiUnattended.EncryptedAdminPassword ■ SysPrep.GuiUnattended.TimeZone	<p>Gibt Informationen an, die der SysPrep-Antwortdatei auf Maschinen während der WinPE-Phase der Bereitstellung hinzugefügt werden sollen. Bereits in der SysPrep-Antwortdatei vorhandene Informationen werden von diesen benutzerdefinierten Eigenschaften überschrieben. <i>Section</i> steht für den Namen des Abschnitts der SysPrep-Antwortdatei, wie beispielsweise „GuiUnattended“ oder „UserData“. <i>Key</i> steht für einen Schlüsselnamen im Abschnitt. Um beispielsweise als Zeitzone einer bereitgestellten Maschine Westpazifik (Normalzeit) festzulegen, definieren Sie die benutzerdefinierte Eigenschaft GuiUnattended.UserData.TimeZone und legen Sie „275“ als Wert fest.</p> <p>Eine vollständige Aufstellung der Abschnitte, Schlüssel und akzeptierten Werte finden Sie in der Dokumentation zum Systemvorbereitungsdienstprogramm (SysPrep) für Windows.</p> <p>Die folgenden <i>Section.Key</i>-Kombinationen können für die WIM-basierte Bereitstellung angegeben werden:</p> <ul style="list-style-type: none"> ■ GuiUnattended <ul style="list-style-type: none"> ■ AdminPassword ■ EncryptedAdminPassword ■ TimeZone ■ UserData <ul style="list-style-type: none"> ■ ProductKey ■ FullName ■ ComputerName ■ OrgName ■ Identification <ul style="list-style-type: none"> ■ DomainAdmin ■ DomainAdminPassword ■ JoinDomain ■ JoinWorkgroup
Sysprep.Identification.DomainAdmin	Gibt einen Benutzernamen mit Administratorzugriff auf die Zieldomäne in Active Directory an. Schließen Sie die Benutzerdomäne nicht in die Anmeldedaten ein, die Sie an vCloud Director oder vCloud Air senden.
Sysprep.Identification.DomainAdminPassword	Gibt das Kennwort für die Eigenschaft Sysprep.Identification.DomainAdmin an.
Sysprep.Identification.JoinDomain	Gibt den Namen der Domäne an, der in Active Directory beigetreten werden soll.
Sysprep.Identification.JoinWorkgroup	Gibt den Namen der Arbeitsgruppe an, der beigetreten werden soll, wenn keine Domäne verwendet wird.

Tabelle 1-22. Benutzerdefinierte Eigenschaften für die PXE- und WIM-Bereitstellung (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>SysPrep.UserData.ComputerName</code>	Gibt einen Maschinennamen an, wie beispielsweise „lab-client005“.
<code>SysPrep.UserData.FullName</code>	Gibt den vollständigen Namen eines Benutzers an.
<code>SysPrep.UserData.OrgName</code>	Gibt den Organisationsnamen des Benutzers an.
<code>SysPrep.UserData.ProductKey</code>	Gibt den Windows-Product Key an.

Benutzerdefinierte Eigenschaften für OVF-Import

Wenn Sie eine OVF-Datei in einen Blueprint importieren, können Sie mehrere Einstellungen als benutzerdefinierte Eigenschaften importieren und konfigurieren.

Tabelle 1-23. Benutzerdefinierte Eigenschaften für Blueprints, in die eine OVF-Datei importiert wird

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VMware.Ovf.Thumbprint</code>	<p>Wenn sich die OVF-Datei auf einem HTTPS-Server befindet, der über ein Zertifikat verfügt, speichert diese Eigenschaft den Fingerabdruckwert dieses Zertifikats und wird zum Überprüfen des Zertifikats verwendet. Dies spielt keine Rolle, wenn die OVF-Datei auf einem HTTP-Server gehostet wird. Die Eigenschaft wird automatisch erstellt, wenn Sie eine OVF-Datei mit dem Bereitstellungsworkflow <code>ImportOvfWorkFlow</code> in der Benutzeroberfläche der Blueprint-Komponente importieren. Wenn Sie den Blueprint programmatisch mit vRealize Automation-REST APIs oder vRealize CloudClient erstellen, müssen Sie die Eigenschaft manuell erstellen.</p> <hr/> <p>Hinweis Der Fingerabdruck kann in einem kommagetrennten Format zur Unterstützung einer Zertifikatskette gespeichert werden.</p> <hr/> <p>Wenn <code>VMware.Ovf.TrustAllCertificates</code> vorhanden ist, wird die Eigenschaft <code>VMware.Ovf.Thumbprint</code> ignoriert.</p>
<code>VMware.Ovf.TrustAllCertificates</code>	<p>Wenn diese Eigenschaft vorhanden und auf „true“ festgelegt ist, wird die Eigenschaft <code>VMware.Ovf.Thumbprint</code> ignoriert, und beim Importieren einer OVF unter Verwendung des Bereitstellungsworkflows <code>ImportOvfWorkFlow</code> wird keine Zertifikatsvalidierung durchgeführt.</p>
<code>VMware.Ovf.Configuration.X</code>	<p>Eine OVF-Datei kann benutzerkonfigurierbare Eigenschaften enthalten, zum Beispiel eine Eigenschaft, die das Root-Kennwort einer von der OVF-Datei bereitgestellten VM festlegt. Wenn Sie eine OVF-Datei in einen Blueprint importieren, werden alle benutzerkonfigurierbaren Eigenschaften, die in der OVF-Datei definiert sind, analysiert und in benutzerdefinierte Eigenschaften der Form <code>VMware.Ovf.Configuration.X</code> konvertiert, wobei X der Name der benutzerkonfigurierbaren Eigenschaft aus der OVF-Datei ist.</p>

Benutzerdefinierte Eigenschaften für den vRealize Automation-Gast-Agent

Wenn Sie den vRealize Automation-Gast-Agent in Ihren Vorlagen für das Klonen oder in WinPE installiert haben, können Sie mithilfe benutzerdefinierter Eigenschaften benutzerdefinierte Skripts innerhalb des Gastbetriebssystems einer bereitgestellten Maschine ausführen, nachdem die Maschine vollständig bereitgestellt wurde.

Tabelle 1-24. Benutzerdefinierte Eigenschaften zum Anpassen bereitgestellter Maschinen mit einem Gast-Agent

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.AddOwnerToAdmins</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um den Besitzer der Maschine, gemäß der Eigenschaft <code>VirtualMachine.Admin.Owner</code> , zur lokalen Administratorengruppe auf der Maschine hinzuzufügen. Diese Eigenschaft ist für die Bereitstellung mittels Klonen nicht verfügbar.
<code>VirtualMachine.Admin.AllowLogin</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um den Maschinenbesitzer zur Gruppe der lokalen Remotedesktopbenutzer, gemäß der Eigenschaft <code>VirtualMachine.Admin.Owner</code> , hinzuzufügen.
<code>VirtualMachine.Admin.UseGuestAgent</code>	Wenn der Gast-Agent als Dienst in einer Vorlage für das Klonen installiert ist, legen Sie diese Eigenschaft im Maschinen-Blueprint auf „True“ fest, um den Gast-Agent-Dienst auf Maschinen, die anhand dieser Vorlage geklont werden, zu aktivieren. Beim Starten der Maschine wird der Gast-Agent-Dienst gestartet. Legen Sie diese Eigenschaft auf „False“ fest, um den Gast-Agent zu deaktivieren. Mit der Einstellung „False“ verwendet der erweiterte Klon-Workflow den Gast-Agent nicht für Aufgaben des Gastbetriebssystems, wodurch dessen Funktionalität auf <code>VMwareCloneWorkflow</code> reduziert wird. Wenn diese Option nicht angegeben ist oder auf einen anderen Wert als „False“ festgelegt ist, sendet der erweiterte Klon-Workflow Arbeitselemente an den Gast-Agent.
<code>VirtualMachine.DiskN.Active</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine aktiv ist. Legen Sie diese Eigenschaft auf „False“ fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine nicht aktiv ist.
<code>VirtualMachine.DiskN.Label</code>	Gibt die Bezeichnung für die Festplatte <i>N</i> einer Maschine an. Für die Festplattenbezeichnung sind maximal 32 Zeichen zulässig. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent wird hiermit die Bezeichnung der Festplatte <i>N</i> einer Maschine im Gastbetriebssystem angegeben.
<code>VirtualMachine.DiskN.Letter</code>	Gibt den Laufwerkbuchstaben oder Mount-Punkt der Festplatte <i>N</i> einer Maschine an. Der Standardwert ist C. Um beispielsweise den Buchstaben D für die Festplatte 1 anzugeben, definieren Sie die benutzerdefinierte Eigenschaft als <code>VirtualMachine.Disk1.Letter</code> und geben Sie den Wert „D“ ein. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent gibt dieser Wert den Laufwerkbuchstaben oder Einhängpunkt an, unter dem die zusätzliche Festplatte <i>N</i> vom Gast-Agent im Gastbetriebssystem gemountet wird.

Tabelle 1-24. Benutzerdefinierte Eigenschaften zum Anpassen bereitgestellter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.FS</code>	Für die Verwendung mit Windows-Gast-Agent (gugent). Gibt das Dateisystem der Festplatte <i>N</i> der Maschine an. Mögliche Optionen sind NTFS (Standardwert), FAT und FAT32. Ein Beispiel für die Verwendung finden Sie im Windows-Agent-Skript <code>10_setupdisks.bat</code> .
<code>VirtualMachine.DiskN.FileSystem</code>	Für die Verwendung mit Linux-Gast-Agent (gugent). Gibt das Dateisystem der Festplatte <i>N</i> der Maschine an. Die Optionen sind ext3, ext4 und XFS. Ein Beispiel für die Verwendung finden Sie im Linux-Agent-Skript <code>30_DiskSetup.sh</code> .
<code>VirtualMachine.Admin.CustomizeGuestOSDelay</code>	Gibt an, wie lange nach Abschluss der Anpassung gewartet werden soll, bevor die Anpassung des Gastbetriebssystems gestartet wird. Für diesen Wert ist das Format HH:MM:SS erforderlich. Wenn dieser Wert nicht festgelegt wird, wird der Standardwert von einer Minute (00:01:00) verwendet. Wenn Sie diese benutzerdefinierte Eigenschaft nicht angeben, schlägt die Bereitstellung möglicherweise fehl, falls die virtuelle Maschine neu gestartet wird, bevor Arbeitselemente des Gast-Agents abgeschlossen sind.
<code>VirtualMachine.Customize.WaitComplete</code>	Legen Sie diese Eigenschaft auf „True“ fest, um zu verhindern, dass der Bereitstellungsworkflow Arbeitselemente an den Gast-Agent sendet, bevor alle Anpassungen abgeschlossen sind. Legen Sie diese Eigenschaft auf „False“ fest, um die Erstellung von Arbeitselementen zu ermöglichen, bevor die Anpassung abgeschlossen ist.
<code>VirtualMachine.SoftwareN.Name</code>	Gibt den beschreibenden Namen der Softwareanwendung <i>N</i> oder eines Skripts an, die bzw. das während der Bereitstellung installiert oder ausgeführt werden soll. Dies ist eine optionale und rein informative Eigenschaft. Sie hat keine echte Funktion für den erweiterten Klon-Workflow oder den Gast-Agent, ist aber hilfreich für die benutzerdefinierte Softwareauswahl in einer Benutzeroberfläche oder für Berichte zur Softwarenutzung.

Tabelle 1-24. Benutzerdefinierte Eigenschaften zum Anpassen bereitgestellter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.SoftwareN.ScriptPath	<p>Gibt den vollständigen Pfad zum Installationsskript einer Anwendung an. Bei dem Pfad muss es sich um einen gültigen absoluten Pfad wie er im Gastbetriebssystem angezeigt wird handeln und er muss den Namen der Skriptdatei enthalten.</p> <p>Sie können benutzerdefinierte Eigenschaftswerte als Parameter an das Skript übergeben, indem Sie <code>{CustomPropertyName}</code> in der Pfadzeichenfolge einfügen. Angenommen, Sie haben eine benutzerdefinierte Eigenschaft <code>ActivationKey</code> mit dem Wert 1234. In diesem Fall lautet der Skriptpfad <code>D:\InstallApp.bat -key {ActivationKey}</code>. Der Gast-Agent führt den Befehl <code>D:\InstallApp.bat -key 1234</code> aus. Ihre Skriptdatei kann dann so programmiert werden, dass dieser Wert akzeptiert und verwendet wird.</p> <p>Fügen Sie <code>{Owner}</code> ein, um den Namen des Maschinenbesitzers an das Skript zu übergeben.</p> <p>Sie können auch benutzerdefinierte Eigenschaftswerte als Parameter an das Skript weitergeben, indem Sie <code>{YourCustomProperty}</code> in die Pfadzeichenfolge einfügen. Wenn Sie beispielsweise den Wert <code>\\vra-scripts.mycompany.com\scripts\changeIP.bat</code> eingeben, wird das Skript <code>changeIP.bat</code> von einem gemeinsam genutzten Speicherort ausgeführt. Wenn Sie jedoch den Wert <code>\\vra-scripts.mycompany.com\scripts\changeIP.bat {VirtualMachine.Network0.Address}</code> eingeben, wird das Skript für die Änderung der IP-Adresse ausgeführt, aber auch der Wert der Eigenschaft <code>VirtualMachine.Network0.Address</code> als Parameter an das Skript weitergegeben.</p>

Tabelle 1-24. Benutzerdefinierte Eigenschaften zum Anpassen bereitgestellter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.ScriptPath.Decrypt	<p>Ermöglicht vRealize Automation das Abrufen einer verschlüsselten Zeichenfolge, die als ordnungsgemäß formatierte benutzerdefinierte Eigenschaftsanweisung <code>VirtualMachine.SoftwareN.ScriptPath</code> an die gugen-Befehlszeile übergeben wird.</p> <p>Sie können eine verschlüsselte Zeichenfolge wie beispielsweise Ihr Kennwort als benutzerdefinierte Eigenschaft in einem Befehlszeilenargument bereitstellen. Auf diese Weise können Sie verschlüsselte Informationen speichern, die der Gast-Agent entschlüsseln und als gültiges Befehlszeilenargument interpretieren kann. Beispielsweise ist die benutzerdefinierte Eigenschaftszeichenfolge</p> <pre>VirtualMachine.Software0.ScriptPath = c:\dosomething.bat</pre> <p><i>Kennwort</i> nicht sicher, da sie ein tatsächliches Kennwort enthält.</p> <p>Zum Entschlüsseln des Kennworts können Sie eine benutzerdefinierte vRealize Automation-Eigenschaft erstellen, wie beispielsweise <code>MyPassword = password</code>, und die Verschlüsselung durch Aktivieren des verfügbaren Kontrollkästchens aktivieren. Der Gast-Agent entschlüsselt den Eintrag [MyPassword] in den Wert in der benutzerdefinierten Eigenschaft <code>MyPassword</code> und führt das Skript als <code>c:\dosomething.bat password</code> aus.</p> <ul style="list-style-type: none"> ■ Erstellen Sie die benutzerdefinierte Eigenschaft MyPassword = Kennwort, wobei <i>Kennwort</i> der Wert Ihres tatsächlichen Kennworts ist. Aktivieren Sie die Verschlüsselung durch Aktivieren des verfügbaren Kontrollkästchens. ■ Legen Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.ScriptPath.Decrypt</code> als VirtualMachine.ScriptPath.Decrypt = true fest. ■ Legen Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.Software0.ScriptPath</code> als VirtualMachine.Software0.ScriptPath = c:\dosomething.bat [MyPassword] fest. <p>Wenn Sie <code>VirtualMachine.ScriptPath.Decrypt</code> auf „False“ festlegen oder die benutzerdefinierte Eigenschaft <code>VirtualMachine.ScriptPath.Decrypt</code> nicht erstellen, wird die Zeichenfolge in den eckigen Klammern ([und]) nicht entschlüsselt.</p>

Tabelle 1-24. Benutzerdefinierte Eigenschaften zum Anpassen bereitgestellter Maschinen mit einem Gast-Agent (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.SoftwareN.ISOName	Gibt den Pfad und den Dateinamen der ISO-Datei in Bezug auf das Stammverzeichnis des Datenspeichers an. Das Format lautet <i>/Ordnername/Unterordnername/Dateiname.iso</i> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.
VirtualMachine.SoftwareN.ISOLocation	Gibt den Speicherpfad an, der die ISO-Imagedatei enthält, die von der Anwendung oder dem Skript verwendet werden soll. Formatieren Sie den in der Hostreservierung angezeigten Pfad, wie beispielsweise <i>netapp-1:it_nfs_1</i> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.

Benutzerdefinierte Eigenschaften für die Integration von BMC BladeLogic Configuration Manager

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für die Integration von BMC BladeLogic Configuration Manager bereitstellen können.

Tabelle 1-25. Für die Integration von BMC BladeLogic Configuration Manager erforderliche benutzerdefinierte Eigenschaften

Benutzerdefinierte Eigenschaft	Beschreibung
VirtualMachine.EPI.Type	Gibt den External Provisioning Infrastructure-Typ an.
VirtualMachine.Admin.Owner	Gibt den Benutzernamen des Maschinenbesitzers an.
BMC.Software.Install	Legen Sie diese Eigenschaft auf „True“ fest, um die Integration von BMC BladeLogic Configuration Manager zu aktivieren.
EPI.Server.Name	Gibt den Namen des External Provisioning Infrastructure-Servers an, wie beispielsweise den Namen des Servers, der BMC BladeLogic hostet. Wenn mindestens ein allgemeiner BMC EPI-Agent installiert wurde, ohne einen BMC BladeLogic Configuration Manager-Host anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen. Wenn nur dedizierte BMC EPI-Agents für bestimmte BMC BladeLogic Configuration Manager-Hosts installiert wurden, muss dieser Wert genau mit dem für einen dieser Agents konfigurierten Servernamen übereinstimmen.
BMC.Service.Profile	Gibt den Namen des Standardauthentifizierungsprofils auf dem BMC BladeLogic-Server an.

Tabelle 1-25. Für die Integration von BMC BladeLogic Configuration Manager erforderliche benutzerdefinierte Eigenschaften (Fortsetzung)

Benutzerdefinierte Eigenschaft	Beschreibung
BMC.Software.BatchLocation	Gibt den Speicherort bei der Konfiguration von BMC BladeLogic an, in dem Softwareaufträge bereitgestellt werden. Dieser Wert muss mit dem entsprechenden Wert von Vrm.Software.IdNNNN übereinstimmen. Ein gültiger Wert wäre beispielsweise /Application Deployment.
VMware.VirtualCenter.OperatingSystem	Gibt die Version des vCenter Server-Gastbetriebssystems (VirtualMachineGuestOsIdentifier) an, mit der vCenter Server die Maschine erstellt. Diese Betriebssystemversion muss mit der Betriebssystemversion übereinstimmen, die auf der bereitgestellten Maschine installiert werden soll. Administratoren können Eigenschaftsgruppen mithilfe mehrerer Eigenschaftensätze erstellen, wie beispielsweise VMware[OS_Version]Properties. Diese Eigenschaftensätze sind vordefiniert und enthalten die korrekten Werte für VMware.VirtualCenter.OperatingSystem. Diese Eigenschaft dient für die virtuelle Bereitstellung. Weitere Informationen hierzu finden Sie im Abschnitt zum Enumerationstyp VirtualMachineGuestOsIdentifier in der vSphere API/SDK-Dokumentation. Eine Liste der aktuell akzeptierten Werte finden Sie in der vCenter Server-Dokumentation.

Benutzerdefinierte Eigenschaften, um BMC BladeLogic Configuration Manager-Softwareaufträge verfügbar zu machen

Konfigurieren Sie BMC BladeLogic Configuration Manager-Aufträge für vRealize Automation-Integrationen. Stellen Sie alle Softwareaufträge den Benutzern, die Maschinen anfordern, zur Auswahl zur Verfügung, oder geben Sie einen Softwareauftrag an, der auf alle Maschinen angewendet werden soll, die über den Blueprint bereitgestellt werden.

Tabelle 1-26. Benutzerdefinierte Eigenschaften, um Softwareaufträge verfügbar zu machen

Benutzerdefinierte Eigenschaft	Beschreibung
LoadSoftware	Legen Sie diese Eigenschaft auf „True“ fest, um Softwareinstallationsoptionen zu aktivieren.
Vrm.Software.IdNNNN	Gibt einen Softwareauftrag oder eine Softwarerichtlinie an, die auf alle Maschinen, die über den Blueprint bereitgestellt werden, angewendet werden soll. Legen Sie diesen Wert auf job_type=job_path fest, wobei job_type für den BMC BladeLogic-Auftragstyp und job_path für den Speicherort des Auftrags in BMC BladeLogic steht, wie beispielsweise 4=/Utility/putty. NNNN ist eine Zahl zwischen 1000 und 1999. Die erste Eigenschaft muss mit 1000 beginnen und für jede zusätzliche Eigenschaft muss eine Erhöhung in numerischer Reihenfolge durchgeführt werden. <div> 1 – AuditJob 2 – BatchJob 3 – ComplianceJob 4 – DeployJob 5 – FileDeployJob 6 – NSHScriptJob 7 – PatchAnalysisJob 8 – SnapshotJob </div>

Optionale benutzerdefinierte Eigenschaften für die Integration von BMC BladeLogic Configuration Manager

Sie können auch optionale benutzerdefinierte Eigenschaften verwenden, die häufig zusammen mit BMC BladeLogic Configuration Manager-Blueprints verwendet werden.

Tabelle 1-27. Optionale benutzerdefinierte Eigenschaften für die Integration von BMC BladeLogic Configuration Manager

Eigenschaft	Definition
BMC.AddServer.Delay	Gibt an, wie viele Sekunden gewartet werden soll, bevor die Maschine zu BMC BladeLogic Configuration Manager hinzugefügt wird. Die Standardeinstellung ist 30.
BMC.AddServer.Retry	Gibt an, wie viele Sekunden gewartet werden soll, bevor der Vorgang erneut versucht wird, falls der erste Versuch, die Maschine zu BMC BladeLogic Configuration Manager hinzuzufügen, fehlgeschlagen ist. Die Standardeinstellung ist 100.

Benutzerdefinierte Eigenschaften für die Integration von HP Server Automation

vRealize Automation enthält benutzerdefinierte Eigenschaften, mit denen Sie zusätzliche Steuerelemente für die Integration von HP Server Automation bereitstellen können. Einige benutzerdefinierte Eigenschaften sind für die Integration von HP Server Automation erforderlich. Andere benutzerdefinierte Eigenschaften sind optional.

Erforderliche benutzerdefinierte Eigenschaften für die Integration von HP Server Automation

Bestimmte benutzerdefinierte Eigenschaften sind erforderlich, damit ein Blueprint mit HP Server Automation verwendet werden kann.

Tabelle 1-28. Erforderliche benutzerdefinierte Eigenschaften für die Integration von HP Server Automation

Eigenschaft	Definition
<code>VMware.VirtualCenter.OperatingSystem</code>	Gibt die Version des vCenter Server-Gastbetriebssystems (<code>VirtualMachineGuestOsIdentifier</code>) an, mit der vCenter Server die Maschine erstellt. Diese Betriebssystemversion muss mit der Betriebssystemversion übereinstimmen, die auf der bereitgestellten Maschine installiert werden soll. Administratoren können Eigenschaftsgruppen mithilfe mehrerer Eigenschaftensätze erstellen, wie beispielsweise <code>VMware[OS_Version]Properties</code> . Diese Eigenschaftensätze sind vordefiniert und enthalten die korrekten Werte für <code>VMware.VirtualCenter.OperatingSystem</code> . Diese Eigenschaft dient für die virtuelle Bereitstellung.
<code>VirtualMachine.EPI.Type</code>	Gibt den External Provisioning Infrastructure-Typ an.
<code>EPI.Server.Name</code>	Gibt den Namen des External Provisioning Infrastructure-Servers an, wie beispielsweise den Namen des Servers, der BMC BladeLogic hostet. Wenn mindestens ein allgemeiner BMC EPI-Agent installiert wurde, ohne einen BMC BladeLogic Configuration Manager-Host anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen.
<code>Opware.Software.Install</code>	Legen Sie diese Eigenschaft auf „True“ fest, um HP Server Automation das Installieren von Software zu erlauben.
<code>Opware.Server.Name</code>	Gibt den vollqualifizierten Namen des HP Server Automation-Servers an.
<code>Opware.Server.Username</code>	Gibt den Benutzernamen an, der bereitgestellt wird, wenn eine Kennwortdatei im Agent-Verzeichnis erstellt wurde, wie beispielsweise „opwareadmin“. Für diesen Benutzernamen ist Administratorzugriff auf die HP Server Automation-Instanz erforderlich.
<code>Opware.BootImage.Name</code>	Gibt den Wert für das Start-Image gemäß der Definition in HP Server Automation für das 32-Bit-WinPE-Image an, beispielsweise „winpe32“. Diese Eigenschaft ist bei der Bereitstellung mittels Klonen nicht erforderlich.
<code>Opware.Customer.Name</code>	Gibt den Wert für den Kundennamen gemäß der Definition in HP Server Automation an, wie beispielsweise „MyCompanyName“.
<code>Opware.Facility.Name</code>	Gibt den Wert für den Facility-Namen gemäß der Definition in HP Server Automation an, wie beispielsweise „Cambridge“.

Tabelle 1-28. Erforderliche benutzerdefinierte Eigenschaften für die Integration von HP Server Automation (Fortsetzung)

Eigenschaft	Definition
<code>Opware.Machine.Password</code>	Gibt das standardmäßige lokale Administratorkennwort für ein Betriebssystemsequenz-WIM-Image wie etwa <code>Opware.OSSequence.Name</code> gemäß der Definition in HP Server Automation an, wie beispielsweise „P@ssword1“.
<code>Opware.OSSequence.Name</code>	Gibt den Wert für den Namen der Betriebssystemsequenz gemäß der Definition in HP Server Automation an, wie beispielsweise „Windows 2008 WIM“.
<code>Opware.Realm.Name</code>	Gibt den Wert für den Bereichsnamen gemäß der Definition in HP Server Automation an, wie beispielsweise „Production“.
<code>Opware.Register.Timeout</code>	Gibt an, wie lange in Sekunden gewartet werden soll, bis die Erstellung eines Bereitstellungsauftrags abgeschlossen ist.
<code>VirtualMachine.CDRom.Attach</code>	Legen Sie diese Eigenschaft auf „False“ fest, um die Maschine ohne CD-ROM-Gerät bereitzustellen. Der Standardwert ist „True“.
<code>Linux.ExternalScript.Name</code>	Gibt den Namen eines optionalen Anpassungsskripts an, wie beispielsweise <code>config.sh</code> , das der Linux-Gast-Agent nach der Installation des Betriebssystems ausführt. Diese Eigenschaft ist für über Vorlagen geklonte Linux-Maschinen verfügbar, auf denen der Linux-Agent installiert ist.
<code>Linux.ExternalScript.LocationType</code>	Gibt den Speicherorttyp des in der Eigenschaft <code>Linux.ExternalScript.Name</code> benannten Anpassungsskripts an. Mögliche Werte sind „local“ oder „nfs“.
<code>Linux.ExternalScript.Path</code>	Gibt den lokalen Pfad zum Linux-Anpassungsskript oder den Exportpfad zur Linux-Anpassung auf dem NFS-Server an. Dieser Wert muss mit einem Schrägstrich beginnen und darf den Dateinamen nicht enthalten, wie beispielsweise <code>/scripts/linux/config.sh</code> .

Optionale benutzerdefinierte Eigenschaften für die Integration von HP Server Automation

Bestimmte benutzerdefinierte Eigenschaften sind optional, damit ein Blueprint mit HP Server Automation verwendet werden kann.

Tabelle 1-29. Optionale benutzerdefinierte Eigenschaften für die Integration von HP Server Automation

Eigenschaft	Definition
Opware.ProvFail.Notify	(Optional) Gibt die Benachrichtigungs-E-Mail-Adresse für HP Server Automation für den Fall eines Bereitstellungsfehlers an, wie beispielsweise „provisionfail@lab.local“.
Opware.ProvFail.Notify	(Optional) Gibt den HP Server Automation-Benutzer an, an den das Besitzrecht übertragen wird, falls die Bereitstellung fehlschlägt.
Opware.ProvSuccess.Notify	(Optional) Gibt die Benachrichtigungs-E-Mail-Adresse für HP Server Automation im Falle einer erfolgreichen Bereitstellung an.
Opware.ProvSuccess.Owner	(Optional) Gibt den HP Server Automation-Benutzer an, an den das Besitzrecht im Falle einer erfolgreichen Bereitstellung übertragen wird.

Benutzerdefinierte Eigenschaften, mit denen HP Server Automation-Softwareaufträge verfügbar gemacht werden

Je nachdem, wie Ihr Fabric-Administrator HP Server Automation-Aufträge für die vRealize Automation-Integration konfiguriert, können Sie möglicherweise wahlweise alle Softwareaufträge den Benutzern, die Maschinen anfordern, zur Auswahl zur Verfügung stellen, oder Sie können Aufträge angeben, die auf alle Maschinen angewendet werden sollen, die über Ihren Blueprint bereitgestellt werden.

Tabelle 1-30. Benutzerdefinierte Eigenschaften, um Softwareaufträge verfügbar zu machen

Eigenschaft	Definition
LoadSoftware	Legen Sie diese Eigenschaft auf „True“ fest, um Softwareinstallationsoptionen zu aktivieren.
Vrm.Software.Id	(Optional) Gibt eine HP Server Automation-Richtlinie an, die auf alle Maschinen, die über den Blueprint bereitgestellt werden, angewendet werden soll. <i>NNNN</i> ist eine Zahl zwischen 1000 und 1999. Die erste Eigenschaft muss mit 1000 beginnen und für jede zusätzliche Eigenschaft muss eine Erhöhung in numerischer Reihenfolge durchgeführt werden.

Benutzerdefinierte Eigenschaften gruppiert nach dem Namen

Mithilfe von benutzerdefinierten Eigenschaften können Sie zusätzliche vRealize Automation-Steuerelemente bereitstellen.

Benutzerdefinierte Eigenschaften wurden hier nach dem Namen gruppiert. Nach der Funktion gruppierte benutzerdefinierte Eigenschaften finden Sie unter [Benutzerdefinierte Eigenschaften gruppiert nach der Funktion](#).

Benutzerdefinierte Eigenschaften mit Unterstrich (_)

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit einem Unterstrich (_) beginnen.

Tabelle 1-31. Tabelle der benutzerdefinierten Eigenschaften mit Unterstrichen (_)

Eigenschaft	Beschreibung
_debug_deployment	<p>Mit Ausnahme von Skalierungsvorgängen, die teilweise erfolgreiche Bereitstellungen ermöglichen, wird standardmäßig die gesamte Bereitstellung gelöscht, wenn einzelne Ressourcen nicht bereitgestellt werden können. Sie können das Standardverhalten überschreiben, indem Sie den benutzerdefinierten Eigenschaftswert <code>_debug_deployment</code> auf „true“ festlegen. Wenn die Bereitstellung fehlschlägt, verhindert die benutzerdefinierte Debugging-Eigenschaft, dass ein Rollback der Ressourcen durchgeführt wird. Auf diese Weise können Sie ermitteln, welche Komponenten bei der Bereitstellung fehlgeschlagen sind.</p> <p>Mit anderen Worten: Wenn Sie <code>_debug_deployment</code> auf „true“ festlegen, können Sie die Anpassung und den ersten Start (z. B. Agent) einfacher debuggen, da durch die Einstellung sichergestellt ist, dass Maschinen nach einem Bereitstellungsfehler nicht gelöscht werden. Andernfalls ändert die Einstellung nichts direkt am Bereitstellungsvorgang und wirkt sich nicht direkt auf den Gast-Agent oder die Anpassung aus (z. B. Festlegen unserer Ergebnisse relativ zu einer vCenter-Anpassungsspezifikation).</p> <p>Hinweis: Auf ein fehlgeschlagenes Katalogelement kann normalerweise nicht zugegriffen werden, da bei einem Fehler sofort ein Rollback durchgeführt wird. Wenn <code>_debug_deployment</code> jedoch auf „true“ festgelegt ist, behandelt vRealize Automation die andernfalls fehlgeschlagene Bereitstellung als teilweise erfolgreich und ermöglicht somit den Zugriff.</p> <p>Um die benutzerdefinierte Eigenschaft auf einen Blueprint anzuwenden, fügen Sie <code>_debug_deployment</code> beim Erstellen oder Bearbeiten eines Blueprints über die Registerkarte Eigenschaften zur Seite Blueprint-Eigenschaften hinzu. Die <code>_debug_deployment</code>-Eigenschaft wird auf der Softwarebereitstellungsebene verwendet, und nicht auf der Ebene der Gast-Agent- oder Maschinenbereitstellung.</p> <p>Sie können vRealize Automation auch so konfigurieren, dass virtuelle Maschinen nach dem Fehlschlagen der Bereitstellung nicht gelöscht werden, indem Sie Einstellungen in der Datei <code>VRMAgent.exe.config</code> verwenden.</p>
_deploymentName	<p>Wenn diese Eigenschaft einem Blueprint hinzugefügt wird, können Sie damit einen benutzerdefinierten Namen für die Bereitstellung angeben, indem Sie den Wert von <code>_deploymentName</code> auf Ihre benutzerdefinierte Zeichenfolge festlegen. Wenn mehrere Instanzen dieser Bereitstellung in einer Einzelanforderung bereitgestellt werden, wird der benutzerdefinierte Name als Präfix verwendet. Wenn Sie</p>

Tabelle 1-31. Tabelle der benutzerdefinierten Eigenschaften mit Unterstrichen (_) (Fortsetzung)

Eigenschaft	Beschreibung
	<p>möchten, dass Benutzer eigene Bereitstellungsamen angeben, legen Sie diese Eigenschaft so fest, dass Überschreibungen zulässig sind. Die folgenden zwei Vorbehalte sind für die Nutzung erforderlich:</p> <ul style="list-style-type: none"> ■ Sie müssen diese Eigenschaft auf Blueprint-Ebene hinzufügen, nicht auf Komponentenebene. Klicken Sie beispielsweise beim Erstellen oder Bearbeiten eines Blueprints auf die Registerkarte Eigenschaften und wählen Sie anschließend Benutzerdefinierte Eigenschaften > Neu aus, um die <code>_deploymentName</code>-Eigenschaft dem Blueprint hinzuzufügen. Fügen Sie die Eigenschaft keiner Maschine oder anderen Komponente im Blueprint hinzu. ■ Sie müssen diese Eigenschaft als separate Eigenschaft und nicht als Mitglied einer Eigenschaftsgruppe hinzufügen.

Benutzerdefinierte Eigenschaften A

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben A beginnen.

Tabelle 1-32. Tabelle mit benutzerdefinierten Eigenschaften – A

Eigenschaft	Beschreibung
<code>AD.Lookup.Department</code>	Gibt den Kostenstellenwert an, der in einer an Genehmiger gesendeten Benachrichtigungs-E-Mail enthalten ist. Dieser Eigenschaftswert muss im Blueprint angegeben werden.
<code>agent.download.url</code>	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für Ihre Software-Agent-Datei angegeben. Beispielsweise <code>https://Private_IP:1443/software-service/resources/noble-agent.jar</code>.</p> <p>Gibt die URL für den VPN-Agent in Ihrer Bereitstellung an. Das URL-Format lautet <code>https:// Private_IP:1443/software-service/resources/noble-agent.jar</code></p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.agent.service.url</code> und <code>software.ebs.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie die PAT- oder NAT-Übersetzung und die Portweiterleitung verwenden.</p>
<code>amazon.AmazonEC2Config.ServiceURL</code>	Gibt die URL für den Amazon-Konfigurationsdienst für Amazon GovCloud an, beispielsweise <code>amazon.AmazonEC2Config.ServiceURL=https://ec2.us-gov-west-1.amazonaws.com</code> .
<code>amazon.ElasticLoadBalancingConfig.ServiceURL</code>	Gibt die URL für den Amazon-Lastausgleichskonfigurationsdienst für Amazon GovCloud an, beispielsweise <code>amazon.ElasticLoadBalancingConfig.ServiceURL=https://elasticloadbalancing.us-gov-west-1.amazonaws.com</code> .

Tabelle 1-32. Tabelle mit benutzerdefinierten Eigenschaften – A (Fortsetzung)

Eigenschaft	Beschreibung
<code>Amazon.ElasticLoadBalancer.Names</code>	Weist über einen Blueprint bereitgestellte Maschinen den elastischen Lastausgleichsmodulen zu, die mit den angegebenen Werten übereinstimmen. Diese Eigenschaft kann für vSphere-, Amazon- und Hyper-V-Konfigurationen verwendet werden.
<code>Amazon.Extensions.UserData</code>	<p>Gibt den Namen eines Amazon-Benutzerdatenskripts an, das beim Starten einer Instanz während des ersten Startzyklus ausgeführt wird. Die Eigenschaft unterstützt den Austausch von Zeichenfolgen aus anderen benutzerdefinierten Eigenschaften, um dynamischen Anforderungen zuzulassen. Sie können die Eigenschaft entweder zum gesamten vRealize Automation-Blueprint oder zu einer AWS-Maschinenkomponente im Blueprint hinzufügen.</p> <p>Informationen zu Amazon-Benutzerdatenskripts finden Sie im Thema Ausführen von Befehlen auf Ihrer Linux-Instanz beim Start in der Produktdokumentation von <i>Amazon Elastic Compute Cloud</i>.</p> <p>Sie können eine Reihe von benutzerdefinierten Eigenschaften an die Eigenschaft <code>Amazon.Extensions.UserData</code> übergeben, indem Sie diese in eine Datei aufnehmen, deren Name mit <code>Amazon.CustomProperty.Shell</code> beginnt.</p>
<code>amazon.IAMInstanceProfile.ARN</code>	Gibt die ARNs (Amazon Resource Names) des AWS-Instanzprofils „Identity and Access Management (IAM)“ bei Anforderung einer AWS-Instanz an. Wenn Sie diese Eigenschaft, z. B. <code>amazon.IAMInstanceProfile.ARN</code> = <i>ARN-Wert des IAM-Instanzprofils</i> , zu einem Blueprint hinzufügen und dann eine Bereitstellung aus dem Katalog anfordern, enthält die bereitgestellte virtuelle Amazon-Maschine oder Instanz die angegebene IAM-Rolle. Der DEM liest die Eigenschaftsspezifikation (z. B. <code>amazon.IAMInstanceProfile.ARN</code> = <i>ARN-Wert des IAM-Instanzprofils</i>) und schließt sie in den <code>RunInstanceRequest-Amazon-Workflow</code> ein.
<code>Amazon.Instance.Id</code>	Gibt die Amazon-Instanz-ID einer Maschine an, die auf einem Amazon EC2-Endpoint bereitgestellt wird. Diese Eigenschaft kann für vSphere- und Amazon-Konfigurationen verwendet werden.
<code>Amazon.Instance.GroupName</code>	<p>Gibt den Namen der vorhandenen AWS-Platzierungsgruppe für den verknüpften Amazon-Endpoint an. Die Platzierungsgruppe muss vor der vRealize Automation-Datensammlung des Endpoints in der Verfügbarkeitszone des Ziels vorhanden sein.</p> <p>Fügen Sie die benutzerdefinierte Eigenschaft <code>Amazon.Instance.GroupName</code> zu einem Blueprint hinzu, um die AWS-Platzierungsgruppe anzugeben, die während der Maschinenbereitstellung verwendet wird.</p>
<code>Amazon.elasticIpAddress.ipAddress</code>	Gibt die Amazon-IP-Adresse an, wobei <i>ipAddress</i> die spezifische IP-Adresse ist, die der Instanz zugewiesen werden soll.
<code>Amazon.Placement.Tenancy</code>	Legen Sie diese Eigenschaft auf „= dedicated“ fest, um anzugeben, dass die AWS-Verbindung speziell für einen dedizierten Mandanten ist. Diese Eigenschaft kann für VPC-Subnetze verwendet werden.

Tabelle 1-32. Tabelle mit benutzerdefinierten Eigenschaften – A (Fortsetzung)

Eigenschaft	Beschreibung
<code>Amazon.Storage.Encrypt</code>	<p>Wenn auf „Wahr“ gesetzt, wird angegeben, ob die an den EC2 angeschlossenen Amazon EBS-Speicherfestplatten verschlüsselt werden sollen oder nicht. Der Standardwert lautet „Falsch“.</p> <p>Die Eigenschaft gilt nur für neue EBS-Volume-Verschlüsselungen. Volumes, die Teil einer Amazon Machine Image (AMI)-Definition sind, behalten ihre AMI-Einstellungen unabhängig von dieser Eigenschaft bei.</p>
<code>Amazon.Storage.iops</code>	<p>Gibt die Eingabe-/Ausgabevorgänge pro Sekunde (IOPS) für das zugeordnete Speichergerät an.</p> <p>Aktuell wird diese Eigenschaft nur unterstützt, wenn für den Eigenschaftswert <code>Amazon.Storage.Type</code> „io1“ angegeben wird. Weitere Informationen finden Sie in der Dokumentation zu Amazon EBS-Volume-Typen.</p> <p>Fügen Sie die benutzerdefinierte Eigenschaft <code>Amazon.Storage.iops</code> zu einem Blueprint hinzu, um die IOPS anzugeben. Der Speichertyp „io1“ ist der einzige AWS-Speichertyp, in dem Sie IOPS angeben können.</p>
<code>Amazon.Storage.Type</code>	<p>Gibt den Amazon EBS-Volume-Typ an, der für Datenspeicher in Bezug auf den zugeordneten Amazon-Endpoint verwendet wird. Alle Festplatten werden mit dem angegebenen Typ bereitgestellt. Es ist nicht möglich, für jede Festplatte einen anderen Volume-Typ anzugeben.</p> <p>Legen Sie den Eigenschaftswert auf einen der in der Dokumentation zu Amazon EBS-Volume-Typen angegebenen Werte für API-Namen fest, z. B. „io1“ oder „gp2“.</p> <p>Fügen Sie einem Blueprint die benutzerdefinierte Eigenschaft <code>Amazon.Storage.Type</code> zu, um den während der Maschinenbereitstellung zu verwendenden EBS-Volume-Typ anzugeben.</p>
<code>Azure.Windows.ScriptPath</code>	<p>Gibt den Pfad zum heruntergeladenen Skript an, mit die Verwendung von Tunneln für Windows-basierte Systeme konfiguriert wird. Aktualisieren Sie den Pfad entsprechend den Anforderungen Ihrer Bereitstellung.</p>
<code>Azure.Linux.ScriptPath</code>	<p>Gibt den Pfad zum heruntergeladenen Skript an, mit dem die Verwendung von Tunneln für Linux-basierte Systeme konfiguriert wird. Aktualisieren Sie den Pfad entsprechend den Anforderungen Ihrer Bereitstellung.</p>

Benutzerdefinierte Eigenschaften B

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben B beginnen.

Tabelle 1-33. Tabelle mit benutzerdefinierten Eigenschaften – B

Eigenschaft	Definition
BMC.AddServer.Delay	Gibt an, wie viele Sekunden gewartet werden soll, bevor die Maschine zu BMC BladeLogic Configuration Manager hinzugefügt wird. Die Standardeinstellung ist 30.
BMC.AddServer.Retry	Gibt an, wie viele Sekunden gewartet werden soll, bevor der Vorgang erneut versucht wird, falls der erste Versuch, die Maschine zu BMC BladeLogic Configuration Manager hinzuzufügen, fehlgeschlagen ist. Die Standardeinstellung ist 100.
BMC.Service.Profile	Gibt den Namen des Standardauthentifizierungsprofils auf dem BMC BladeLogic-Server an.
BMC.Software.BatchLocation	Gibt den Speicherort bei der Konfiguration von BMC BladeLogic an, in dem Softwareaufträge bereitgestellt werden. Dieser Wert muss mit dem entsprechenden Wert von Vrm.Software.IdNNNN übereinstimmen. Ein gültiger Wert wäre beispielsweise /Application Deployment.
BMC.Software.Install	Legen Sie diese Eigenschaft auf „True“ fest, um die Integration von BMC BladeLogic Configuration Manager zu aktivieren.

Benutzerdefinierte Eigenschaften C

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben C beginnen.

Tabelle 1-34. Tabelle mit benutzerdefinierten Eigenschaften – C

Eigenschaft	Definition
Cisco.Organization.Dn	Gibt den Distinguished Name (DN) der Cisco UCS Manager-Organisation an, in der über die Business-Gruppe bereitgestellte Cisco UCS-Maschinen platziert werden. Beispielsweise „org-root/org-Engineering“. Wenn die angegebene Organisation nicht in der Cisco UCS Manager-Instanz vorhanden ist, von der die Maschine verwaltet wird, schlägt die Bereitstellung fehl. Diese Eigenschaft ist nur für Business-Gruppen verfügbar.
CloneFrom	Gibt den Namen einer vorhandenen Maschine oder eines vorhandenen Virtualisierungsplattformobjekts an, die bzw. das geklont werden soll. Beispielsweise eine Vorlage in vCenter Server wie etwa „Win2k8tmpl“.
CloneSpec	Gibt den Namen einer Anpassungsspezifikation auf einer geklonten Maschine an. Beispielsweise ein vordefiniertes SysPrep-Objekt in vCenter Server wie etwa „Win2k Customization Spec“. Der Standardwert ist im Blueprint angegeben.
Command.DiskPart.Options	Wenn Sie die WIM-basierte virtuelle Bereitstellung auf ESX Server-Hosts verwenden, legen Sie diese Eigenschaft auf „Align=64“ fest, um beim Formatieren und Partitionieren der Festplatte der Maschine die empfohlenen Ausrichtungparameter zu verwenden. Diese Eigenschaft ist für die physische Bereitstellung nicht verfügbar.

Tabelle 1-34. Tabelle mit benutzerdefinierten Eigenschaften – C (Fortsetzung)

Eigenschaft	Definition
<code>Command.FormatDisk.Options</code>	Wenn Sie die WIM-basierte virtuelle Bereitstellung auf ESX Server-Hosts verwenden, legen Sie diese Eigenschaft auf „/A:32K“ fest, um beim Formatieren und Partitionieren der Festplatte der Maschine die empfohlenen Ausrichtungsparameter zu verwenden. Diese Eigenschaft ist für die physische Bereitstellung nicht verfügbar.
<code>containers.ipam.driver</code>	<p>Nur für die Verwendung mit Containern. Gibt den IPAM-Treiber an, der zu verwenden ist, wenn eine Container-Netzwerkkomponente zu einem Blueprint hinzugefügt wird. Welche Werte unterstützt werden, hängt von den Treibern ab, die in der Container-Hostumgebung installiert sind, in der sie verwendet werden. Ein unterstützter Wert wäre z. B. <code>infoblox</code> oder <code>calico</code>, je nachdem, welche IPAM-Plug-Ins auf dem Container-Host installiert sind.</p> <p>Bei diesem Eigenschaftsnamen und dem Wert wird die Groß-/Kleinschreibung beachtet. Beim Hinzufügen des Eigenschaftswerts wird dieser nicht überprüft. Wenn zur Bereitstellungszeit der angegebene Treiber nicht auf dem Container-Host vorhanden ist, wird eine Fehlermeldung ausgegeben und die Bereitstellung schlägt fehl.</p>
<code>containers.network.driver</code>	<p>Nur für die Verwendung mit Containern. Gibt den Netzwerktreiber an, der zu verwenden ist, wenn eine Container-Netzwerkkomponente zu einem Blueprint hinzugefügt wird. Welche Werte unterstützt werden, hängt von den Treibern ab, die in der Container-Hostumgebung installiert sind, in der sie verwendet werden. Standardmäßig gehören zu den von Docker bereitgestellten Netzwerktreibern Bridge-, Overlay- und Macvlan-Treiber, wobei bei den von Virtual Container Host (VCH) bereitgestellten Netzwerktreibern der Bridge-Treiber enthalten ist. Netzwerktreiber von Drittanbietern, wie z. B. <code>weave</code> und <code>calico</code>, stehen möglicherweise ebenfalls zur Verfügung, je nachdem, welche Netzwerk-Plug-Ins auf dem Container-Host installiert sind.</p> <p>Bei diesem Eigenschaftsnamen und dem Wert wird die Groß-/Kleinschreibung beachtet. Beim Hinzufügen des Eigenschaftswerts wird dieser nicht überprüft. Wenn zur Bereitstellungszeit der angegebene Treiber nicht auf dem Container-Host vorhanden ist, wird eine Fehlermeldung ausgegeben und die Bereitstellung schlägt fehl.</p>
<code>Container</code>	Nur für die Verwendung mit Containern. Der Standardwert ist <code>App.Docker</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.
<code>Container.Auth.User</code>	Nur für die Verwendung mit Containern. Gibt den Benutzernamen für das Herstellen einer Verbindung zum Container-Host an.
<code>Container.Auth.Password</code>	Nur für die Verwendung mit Containern. Gibt entweder das Kennwort für den Benutzernamen oder für den zu verwendenden öffentlichen bzw. privaten Schlüssel an. Verschlüsselter Eigenschaftswert wird unterstützt.
<code>Container.Auth.PublicKey</code>	Nur für die Verwendung mit Containern. Gibt den öffentlichen Schlüssel für das Herstellen einer Verbindung zum Container-Host an.

Tabelle 1-34. Tabelle mit benutzerdefinierten Eigenschaften – C (Fortsetzung)

Eigenschaft	Definition
<code>Container.Auth.PrivateKey</code>	Nur für die Verwendung mit Containern. Gibt den privaten Schlüssel für das Herstellen einer Verbindung zum Container-Host an. Verschlüsselter Eigenschaftswert wird unterstützt.
<code>Container.Connection.Protocol</code>	Nur für die Verwendung mit Containern. Gibt das Kommunikationsprotokoll an. Der Standardwert ist <code>API</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.
<code>Container.Connection.Scheme</code>	Nur für die Verwendung mit Containern. Gibt das Kommunikationsschema an. Die Standardeinstellung ist <code>https</code> .
<code>Container.Connection.Port</code>	Nur für die Verwendung mit Containern. Gibt den Container-Verbindungs-Port an. Die Standardeinstellung ist <code>2376</code> .
<code>Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineActivated</code>	Nur für die Verwendung mit Containern. Gibt die Event-Broker-Eigenschaft an, um alle Container-Eigenschaften freizulegen, und wird für das Registrieren eines bereitgestellten Hosts verwendet. Der Standardwert ist <code>Container*</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.
<code>Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.Disposing</code>	Nur für die Verwendung mit Containern. Gibt die Event-Broker-Eigenschaft an, um alle der oben genannten Container-Eigenschaften freizulegen, und wird für das Aufheben der Registrierung eines bereitgestellten Hosts verwendet. Der Standardwert ist <code>Container*</code> und erforderlich. Ändern Sie diese Eigenschaft nicht.

Benutzerdefinierte Eigenschaften E

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben E beginnen.

Tabelle 1-35. Tabelle mit benutzerdefinierten Eigenschaften – E

Eigenschaft	Definition
EPI.Server.Collection	Gibt den Namen der Citrix Provisioning-Kollektion an, für die die Maschine registriert werden soll.
EPI.Server.Name	<p>Gibt den Namen des External Provisioning Infrastructure-Servers an, wie beispielsweise den Namen des Servers, der BMC BladeLogic hostet. Wenn mindestens ein allgemeiner BMC EPI-Agent installiert wurde, ohne einen BMC BladeLogic Configuration Manager-Host anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen.</p> <p>Wenn nur dedizierte BMC EPI-Agents für bestimmte BMC BladeLogic Configuration Manager-Hosts installiert wurden, muss dieser Wert genau mit dem für einen dieser Agents konfigurierten Servernamen übereinstimmen.</p> <p>Gibt den Namen des Servers an, der HP Server Automation hostet. Wenn mindestens ein allgemeiner Opware EPI-Agent installiert wurde, ohne einen Server Automation-Server anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen.</p> <p>Wenn nur dedizierte EPI-Agents für bestimmte HP Server Automation-Server installiert wurden, muss dieser Wert genau mit dem für einen dieser Agents konfigurierten Servernamen übereinstimmen.</p> <p>Wenn mindestens ein allgemeiner EPI-Agent des entsprechenden Typs (VirtualMachine.EPI.Type) installiert wurde, ohne einen Server anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen. Wenn nur dedizierte EPI-Agents für bestimmte Server des entsprechenden Typs installiert wurden, muss dieser Wert genau mit dem für einen dieser Agents konfigurierten Servernamen übereinstimmen.</p>
EPI.Server.Port	Gibt den Port an, über den der Provisioning Server kontaktiert werden soll. Wenn Sie einen Citrix Provisioning Server verwenden, geben Sie diese Eigenschaft nicht an, um den Standardportwert 54321 zu verwenden.
EPI.Server.Site	Gibt den Namen der Citrix Provisioning-Site an, die die Kollektion und den Store enthält, die mit den Eigenschaften EPI.Server.Collection und EPI.Server.Store identifiziert werden, wie beispielsweise „site1“.
EPI.Server.Store	Gibt den Namen des Citrix Provisioning-Stores an, der die vDisk enthält, die mit der Eigenschaft EPI.Server.VDiskName identifiziert wird, wie beispielsweise „store1“.
EPI.Server.VDiskName	Gibt den Namen der Citrix Provisioning vDisk an, über die bereitgestellt werden soll, wie beispielsweise „disk1“.
ext.policy.activedirectory.customizationWorkflowTag	Das Tag, das Sie einem benutzerdefinierten vRealize Orchestrator-Workflow hinzugefügt haben. Die Active Directory-Richtlinie sucht nach benutzerdefinierten Workflows mit dem angegebenen Tag und verwendet, wenn welche gefunden werden, den Workflow mit dem Tag beim Erstellen eines Active Directory-Datensatzes.
ext.policy.activedirectory.customizationDeleteWorkflowTag	Das Tag, das Sie einem benutzerdefinierten vRealize Orchestrator-Workflow hinzugefügt haben. Die Active Directory-Richtlinie sucht nach benutzerdefinierten Workflows mit dem angegebenen Tag und verwendet, wenn welche gefunden werden, den Workflow mit dem Tag beim Löschen eines Active Directory-Datensatzes.

Tabelle 1-35. Tabelle mit benutzerdefinierten Eigenschaften – E (Fortsetzung)

Eigenschaft	Definition
<code>ext.policy.activedirectory.domain</code>	Die Domäne, die Sie verwenden möchten, und nicht die Domäne in der aktuellen Active Directory-Richtlinie. Überschreibt den Wert von <code>ext.policy.activedirectory.system.domain</code> , der in der Active Directory-Richtlinie angegeben ist.
<code>ext.policy.activedirectory.endpoint.id</code>	Der Richtlinienbezeichner, der zum Angeben oder Überschreiben einer Richtlinie verwendet werden soll. Sie müssen die ID einer vorhandenen Active Directory-Richtlinie angeben. Überschreibt den Wert von <code>ext.policy.activedirectory.system.endpoint.id</code> , der in der Active Directory-Richtlinie angegeben ist.
<code>ext.policy.activedirectory.id</code>	Die für den Benutzer lesbare ID für die Active Directory-Richtlinie. Überschreibt den Wert von <code>ext.policy.activedirectory.system.id</code> , der in der Active Directory-Richtlinie angegeben ist.
<code>ext.policy.activedirectory.ignore</code>	Gibt an, dass die Maschine nicht von einer Richtlinie zu Active Directory hinzugefügt wird. Die Active Directory-Richtlinie wird übersprungen.
<code>ext.policy.activedirectory.machineName</code>	Der Name der Maschine in Active Directory, den Sie verwenden möchten, und nicht der Name in der aktuellen Active Directory-Richtlinie. Überschreibt den Wert von <code>ext.policy.activedirectory.system.machineName</code> , der in der Active Directory-Richtlinie angegeben ist.
<code>ext.policy.activedirectory.orgunit</code>	Die Organisationseinheit, die Sie verwenden möchten, und nicht die Organisationseinheit in der aktuellen Active Directory-Richtlinie. Überschreibt den Wert von <code>ext.policy.activedirectory.system.orgunit</code> , der in der Active Directory-Richtlinie angegeben ist.
<code>ext.policy.activedirectory.system.domain</code>	Systemeigenschaft für die Domäne der Maschine in Active Directory. Wenn Sie diese Eigenschaft ändern, die von den definierten Richtlinien verwendet wird, können Sie die Richtlinie deaktivieren. Mithilfe von <code>ext.policy.activedirectory.domain</code> können Sie den Richtlinienwert überschreiben.
<code>ext.policy.activedirectory.system.endpoint.id</code>	Systemeigenschaft für den Namen des Active Directory vRealize Orchestrator-Endpoints. Wenn Sie diese Eigenschaft ändern, die von den definierten Richtlinien verwendet wird, können Sie die Richtlinie deaktivieren. Mithilfe von <code>ext.policy.activedirectory.endpoint.id</code> können Sie den Richtlinienwert überschreiben.
<code>ext.policy.activedirectory.system.id</code>	Systemeigenschaft für die für den Benutzer lesbare ID für die Active Directory-Richtlinie. Wenn Sie diese Eigenschaft ändern, die von den definierten Richtlinien verwendet wird, können Sie die Richtlinie deaktivieren. Mithilfe von <code>ext.policy.activedirectory.id</code> können Sie den Richtlinienwert überschreiben.

Tabelle 1-35. Tabelle mit benutzerdefinierten Eigenschaften – E (Fortsetzung)

Eigenschaft	Definition
<code>ext.policy.activedirectory.system.machineName</code>	Systemeigenschaft für den Namen der Maschine in Active Directory. Wenn Sie diese Eigenschaft ändern, die von den definierten Richtlinien verwendet wird, können Sie die Richtlinie deaktivieren. Mithilfe von <code>ext.policy.activedirectory.machineName</code> können Sie den Richtlinienwert überschreiben.
<code>ext.policy.activedirectory.system.orgunit</code>	Systemeigenschaft für den Distinguished Name der Active Directory-Organisationseinheit. Wenn Sie diese Eigenschaft ändern, die von den definierten Richtlinien verwendet wird, können Sie die Richtlinie deaktivieren. Mithilfe von <code>ext.policy.activedirectory.orgunit</code> können Sie den Richtlinienwert überschreiben.

Benutzerdefinierte Eigenschaften H

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben H beginnen.

Tabelle 1-36. Tabelle mit benutzerdefinierten Eigenschaften – H

Eigenschaft	Definition
<code>Hostname</code>	Gibt den Namen der Hostmaschine an und überschreibt den generierten Maschinennamen in der Eigenschaft <code>VirtualMachine.Admin.Name</code> . Wenn <code>Hostname</code> nicht verwendet wird, wird der Wert <code>VirtualMachine.Admin.Name</code> als Maschinename verwendet. Die maximal zulässige Anzahl an Zeichen für den Wert <code>Hostname</code> beträgt 15.
<code>Hyperv.Network.Type</code>	Gibt den Netzwerkadaptertyp der virtuellen Maschine an. Diese Eigenschaft ist nur für die Verwendung mit Hyper-V (SCVMM) gültig. Wenn der Wert auf synthetisch festgelegt ist, gibt dies an, dass der Blueprint eine Generation-2-Maschine auf einer Hyper-V (SCVMM) 2012 R2-Ressource bereitstellen darf. Die Generation-2-Bereitstellung erfordert zudem, dass der Blueprint die Eigenschaftseinstellung <code>Scvmm.Generation2 = true</code> enthält. Der Legacy-Wert ist nicht mit WinXP- oder Server 2003 x64-Gastbetriebssystemen kompatibel. Der Standardwert ist „Synthetisch“.

Benutzerdefinierte Eigenschaften I

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben I beginnen.

Tabelle 1-37. Tabelle mit benutzerdefinierten Eigenschaften – I

Eigenschaft	Definition
<code>Image.ISO.Location</code>	<p>Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Speicherort des ISO-Images an, von dem aus gestartet werden soll. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.</p> <p>Für die virtuelle Bereitstellung mit vCenter Server ist dies der Name eines Datenspeichers in der Instanz, auf den die Bereitstellungs-Computing-Ressource zugreifen kann. Für die virtuelle Bereitstellung mit XenServer ist dies der Name eines Speicher-Repositorys.</p> <p>Für die physische Bereitstellung ist dies die HTTP-URL des Image-Speicherorts mit Webzugriff.</p>
<code>Image.ISO.Name</code>	<p>Bei den Werten für diese Eigenschaft wird die Groß-/Kleinschreibung beachtet. Gibt den Namen des ISO-Images an, von dem aus gestartet werden soll, wie beispielsweise „/ISO/Microsoft/WinPE.iso“. Das Format dieses Werts hängt von der jeweiligen Plattform ab. Ausführliche Informationen finden Sie in der Dokumentation für Ihre Plattform. Diese Eigenschaft ist für die WIM-basierte Bereitstellung, die Linux Kickstart- und die autoYaST-Bereitstellung sowie für die SCCM-basierte Bereitstellung erforderlich.</p> <p>Für die virtuelle Bereitstellung mit vCenter Server bezeichnet dieser Wert den Pfad des Images in dem durch <code>Image.ISO.Location</code> angegebenen Datenspeicher. Für diesen Wert müssen Schrägstriche verwendet werden, und außerdem muss dieser Wert mit einem Schrägstrich beginnen. Für die virtuelle Bereitstellung mit XenServer bezeichnet dieser Wert den Namen des Images in dem durch <code>Image.ISO.Location</code> angegebenen Speicher-Repository. Für die virtuelle Bereitstellung mit Hyper-V gibt dieser Wert den vollständigen lokalen Pfad zum Image an.</p> <p>Für die physische Bereitstellung gibt dieser Wert den Dateinamen des Images an.</p>
<code>Image.ISO.UserName</code>	Gibt den Benutzernamen für den Zugriff auf die CIFS-Freigabe im Format <i>username@domain</i> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
<code>Image.ISO.Password</code>	Gibt das Kennwort für die Eigenschaft <code>Image.ISO.UserName</code> an. Für Dell iDRAC-Integrationen, bei denen sich das Image auf einer CIFS-Freigabe befindet, die für den Zugriff die Authentifizierung erfordert.
<code>Image.WIM.Path</code>	Gibt den UNC-Pfad zur WIM-Datei an, aus der bei der WIM-basierten Bereitstellung ein Image extrahiert wird. Das Pfadformat ist <i>\\server\share\$</i> , beispielsweise <i>\\lab-ad\dfs\$</i> .
<code>Image.WIM.Name</code>	Gibt den Namen der WIM-Datei an, deren Speicherort mit der Eigenschaft <code>Image.WIM.Path</code> festgelegt wird.
<code>Image.WIM.Index</code>	Gibt den verwendeten Index zum Extrahieren des korrekten Images aus der WIM-Datei an.
<code>Image.Network.User</code>	Gibt den Benutzernamen an, mit dem der WIM-Image-Pfad (<code>Image.WIM.Path</code>) einem Netzlaufwerk auf der bereitgestellten Maschine zugeordnet werden soll. Dies ist in der Regel ein Domänenkonto mit Zugriff auf die Netzwerkfreigabe.

Tabelle 1-37. Tabelle mit benutzerdefinierten Eigenschaften – I (Fortsetzung)

Eigenschaft	Definition
Image.Network.Password	Gibt das Kennwort für die Eigenschaft Image.Network.User an.
Image.Network.Letter	Gibt den Laufwerkbuchstaben an, dem der WIM-Image-Pfad auf der bereitgestellten Maschine zugeordnet wird. Der Standardwert ist „K“.
Infrastructure.Admin.MachineObjectOU	Gibt die Organisationseinheit (Organizational Unit, OU) der Maschine an. Wenn Maschinen durch die Business-Gruppen-OU in der erforderlichen OU platziert werden, wird diese Eigenschaft nicht benötigt.
Infrastructure.Admin.ADUser	Gibt die Benutzer-ID des Domänenadministrators an. Mithilfe dieses Bezeichners werden Active Directory-Benutzer und -Gruppen abgefragt, wenn keine anonyme Bindung verwendet werden kann.
Infrastructure.Admin.ADPASSWORD	Gibt das Kennwort für die Domänenadministrator-Benutzer-ID Infrastructure.Admin.ADUser an.
Infrastructure.Admin.DefaultDomain	Gibt die Standarddomäne auf der Maschine an.
Infrastructure.ResourcePool.Name	Gibt ggf. den Ressourcenpool an, zu der die Maschine gehört. Der Standardwert ist der Wert, der in der Reservierung angegeben ist, über die die Maschine bereitgestellt wurde.

Benutzerdefinierte Eigenschaften L

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben L beginnen.

Tabelle 1-38. Tabelle mit benutzerdefinierten Eigenschaften – L

Eigenschaft	Beschreibung
Linux.ExternalScript.LocationType	Gibt den Speicherorttyp des in der Eigenschaft Linux.ExternalScript.Name benannten Anpassungsskripts an. Mögliche Werte sind „local“ oder „nfs“. Darüber hinaus müssen Sie mit der Eigenschaft Linux.ExternalScript.Path den Skriptspeicherort angeben. Wenn der Speicherorttyp „nfs“ lautet, sollten Sie auch die Eigenschaft Linux.ExternalScript.Server verwenden.
Linux.ExternalScript.Name	Gibt den Namen eines optionalen Anpassungsskripts an, wie beispielsweise config.sh, das der Linux-Gast-Agent nach der Installation des Betriebssystems ausführt. Diese Eigenschaft ist für über Vorlagen geklonte Linux-Maschinen verfügbar, auf denen der Linux-Agent installiert ist. Wenn Sie ein externes Skript angeben, müssen Sie auch mithilfe der Eigenschaften Linux.ExternalScript.LocationType und Linux.ExternalScript.Path dessen Speicherort definieren.
Linux.ExternalScript.Path	Gibt den lokalen Pfad zum Linux-Anpassungsskript oder den Exportpfad zur Linux-Anpassung auf dem NFS-Server an. Dieser Wert muss mit einem Schrägstrich beginnen und darf den Dateinamen nicht enthalten, wie beispielsweise /scripts/linux/config.sh.

Tabelle 1-38. Tabelle mit benutzerdefinierten Eigenschaften – L (Fortsetzung)

Eigenschaft	Beschreibung
Linux.ExternalScript.Server	Gibt den Namen des NFS-Servers an, wie beispielsweise „lab-ad.lab.local“, auf dem das in Linux.ExternalScript.Name angegebene externe Linux-Anpassungsskript gespeichert ist.
LoadSoftware	Legen Sie diese Eigenschaft auf „True“ fest, um Softwareinstallationsoptionen zu aktivieren.

Benutzerdefinierte Eigenschaften M

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben M beginnen.

Tabelle 1-39. Tabelle mit benutzerdefinierten Eigenschaften – M

Eigenschaft	Beschreibung
MaximumProvisionedMachines	Gibt die maximale Anzahl von verknüpften Klonen für einen Snapshot der virtuellen Maschine an. Der Standardwert ist „unlimited“.
Machine.SSH	<p>Legen Sie diese Eigenschaft auf „True“ fest, um die Option Verbindungsherstellung mithilfe von SSH auf der vRealize Automation-Seite „Elemente“ für Linux-Maschinen zu aktivieren, die über diesen Blueprint bereitgestellt wurden. Wenn diese Eigenschaft auf „True“ festgelegt ist und die Option Verbinden via RDP oder SSH der Maschine im Blueprint aktiviert ist, wird für alle Linux-Maschinen, die über diesen Blueprint bereitgestellt werden, die Option Verbindungsherstellung mithilfe von SSH für berechtigte Benutzer angezeigt.</p> <p>Gemäß der Option Verbindungsherstellung mithilfe von SSH muss der Browser über ein Plug-In verfügen, das SSH unterstützt, z. B. der SSH-Terminalclient „FireSSH“ für Mozilla Firefox und Google Chrome. Wenn das Plug-In vorhanden ist, wird bei Auswahl von Verbindungsherstellung mithilfe von SSH eine SSH-Konsole mit der Aufforderung angezeigt, Administratoranmeldedaten einzugeben.</p>

Benutzerdefinierte Eigenschaften N

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben N beginnen.

Tabelle 1-40. Tabelle mit benutzerdefinierten Eigenschaften – N

Eigenschaft	Beschreibung
NSX.Edge.ApplianceSize	<p>Gibt die zulässigen Größentypen der NSX-Edge-Appliance für die bereitgestellte Maschine oder die Bereitstellung an. Folgende Optionen stehen zur Verfügung:</p> <ul style="list-style-type: none"> ■ compact <p>Für kleine Bereitstellungen, POCs und Verwendung in einzelnen Diensten.</p> <ul style="list-style-type: none"> ■ CPU = 1 ■ RAM = 512 MB ■ Festplatte = 512 MB ■ large <p>Für kleine bis mittlere Bereitstellungen oder Bereitstellungen mit mehreren Mandanten.</p> <ul style="list-style-type: none"> ■ CPU = 2 ■ RAM = 1 GB ■ Festplatte = 512 MB ■ quadlarge <p>Für Bereitstellungen mit Equal-Cost Multi-Path Routing (ECMP) mit hohem Durchsatz oder mit hoher Leistung und Firewall.</p> <ul style="list-style-type: none"> ■ CPU = 4 ■ RAM = 1 GB ■ Festplatte = 512 MB ■ xlarge <p>Für Bereitstellungen mit L7-Lastausgleich und dedizierte Core-Bereitstellungen.</p> <ul style="list-style-type: none"> ■ CPU = 6 ■ RAM = 8 GB ■ Festplatte = 4,5 GB (4 GB Swap) <p>Weitere Informationen finden Sie unter den Systemanforderungen für NSX.</p>
NSX.Edge.HighAvailability	<p>Wenn der Wert auf „True“ (NSX.Edge.HighAvailability=true) festgelegt ist, wird der Hochverfügbarkeitsmodus auf der NSX Edge-Maschine aktiviert, die über den Blueprint bereitgestellt wird.</p> <p>Bei Verwendung mit NSX.Edge.HighAvailability.PortGroup=port_group_name können Sie mit dieser Eigenschaft einen NSX Edge während der Blueprint-Erstellung konfigurieren.</p> <p>Sie können diese Eigenschaft zu einer NSX-Komponente für den Lastausgleichsdienst im vRealize Automation-Blueprint oder direkt zum vRealize Automation-Blueprint hinzufügen.</p> <p>Muss in Verbindung mit NSX.Edge.HighAvailability.PortGroup=port_group_name verwendet werden.</p>

Tabelle 1-40. Tabelle mit benutzerdefinierten Eigenschaften – N (Fortsetzung)

Eigenschaft	Beschreibung
NSX.Edge.HighAvailability.PortGroup	<p>Erstellt eine mit dem angegebenen Namen der Portgruppe verbundene interne Schnittstelle oder interne vNIC, z. B. NSX.Edge.HighAvailability.PortGroup=VM Network, wobei VM Network eine Portgruppe mit einem Distributed Switch mit Hochverfügbarkeit (vLAN-gestützt) oder eine NSX-Portgruppe mit einem logischen Switch ist. NSX im Hochverfügbarkeitsmodus erfordert mindestens eine interne Schnittstelle oder vNIC.</p> <p>Bei Verwendung mit NSX.Edge.HighAvailability=true können Sie mit dieser Eigenschaft Hochverfügbarkeit für einen NSX Edge während der Blueprint-Erstellung konfigurieren.</p> <p>Wenn Sie bei aktivierter Hochverfügbarkeit One-Arm-Lastausgleich verwenden, müssen Sie für die Hochverfügbarkeit eine getrennte Portgruppe angeben.</p> <hr/> <p>Hinweis Das angegebene Portgruppennetzwerk kann nicht Mitglied des Reservierungspools sein, da die Verwendung der Portgruppe durch die Eigenschaft mit der regulären Nutzung der Portgruppe durch die Bereitstellung konfliktiert, was den folgenden Fehler hervorruft:</p> <div data-bbox="826 961 1233 1014" style="background-color: #f0f0f0; padding: 5px; border: 1px solid #ccc;"> Portgroup must be unique within an Edge... </div> <hr/> <p>Muss in Verbindung mit NSX.Edge.HighAvailability=true verwendet werden.</p>

Tabelle 1-40. Tabelle mit benutzerdefinierten Eigenschaften – N (Fortsetzung)

Eigenschaft	Beschreibung
NSX.Validation.Disable.Single.Edge.Uplink	<p>Wenn der Wert auf „true“ festgelegt ist, ist die NSX-Validierung deaktiviert, die folgende Bedingungen überprüft:</p> <ul style="list-style-type: none"> ■ Alle bedarfsgesteuerten NAT-Netzwerke auf der Blueprint-Quelle im gleichen externen Netzwerk. ■ Alle weitergeleiteten bedarfsgesteuerten Netzwerke auf dem Blueprint, die das gleiche externe Netzwerk wie die VIP-Quelle des Lastausgleichs verwenden. ■ Alle bedarfsgesteuerten Lastausgleichskomponenten auf dem Blueprint haben VIPs auf dem gleichen externen Netzwerk oder bedarfsgesteuerte Netzwerke, die vom gleichen externen Netzwerk gesichert werden. <p>Wenn diese Validierungsprüfung deaktiviert ist, kann eine Bereitstellung zwar erfolgreich sein, der Zugriff auf einige Netzwerkkomponenten ist u. U. jedoch nicht möglich.</p> <p>Falls nicht vorhanden oder auf „false“ gesetzt, ist die Validierung aktiviert (Standard).</p> <p>Ein einzelner NSX Edge unterstützt nur ein externes Netzwerk als dessen Uplink-Netzwerk. Mehrere IP-Adressen von demselben externen Netzwerk werden unterstützt. Ein Blueprint kann eine beliebige Anzahl von externen oder bedarfsgesteuerten Netzwerkkomponenten enthalten; NSX unterstützt jedoch nur ein externes Netzwerk als Uplink-Netzwerk.</p> <p>Diese Eigenschaft kann nur auf der Blueprint-Ebene angegeben werden. Sie kann nicht für eine Komponente auf der Blueprint-Arbeitsfläche angegeben werden.</p>
NSX.Validation.Disable.Blueprint.NSXT	<p>Wenn der Wert auf „true“ festgelegt ist, wird die gesamte NSX-T-Validierung für die Blueprint-Aktion Fertig stellen deaktiviert.</p> <p>Wenn nicht vorhanden oder wenn der Wert auf „false“ festgelegt ist, ist die NSX-T-Validierungsprüfung aktiviert (Standard).</p> <p>Wenn beispielsweise im Blueprint überlappende Subnetze vorhanden sind, wird eine Fehlermeldung angezeigt, wenn Sie im Blueprint auf Fertig stellen klicken, und die Überlappung verhindert, dass der Blueprint fertig gestellt wird, obwohl Sie ihn speichern können. Wenn Sie den Blueprint fertig stellen möchten, können Sie unter Verwendung der Seite Blueprint-Eigenschaften <code>NSX.Validation.Disable.Blueprint.NSXT</code> hinzufügen und anschließend den Blueprint fertig stellen.</p> <p>Die Eigenschaft deaktiviert nur NSX-T-Validierungen für die Blueprint-Aktion Fertig stellen.</p>

Benutzerdefinierte Eigenschaften O

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben O beginnen.

Tabelle 1-41. Tabelle mit benutzerdefinierten Eigenschaften – O

Eigenschaft	Beschreibung
Opware.BootImage.Name	Gibt den Wert für das Start-Image gemäß der Definition in HP Server Automation für das 32-Bit-WinPE-Image an, beispielsweise „winpe32“. Diese Eigenschaft ist bei der Bereitstellung mittels Klonen nicht erforderlich.
Opware.Customer.Name	Gibt den Wert für den Kundennamen gemäß der Definition in HP Server Automation an, wie beispielsweise „MyCompanyName“.
Opware.Facility.Name	Gibt den Wert für den Facility-Namen gemäß der Definition in HP Server Automation an, wie beispielsweise „Cambridge“.
Opware.Machine.Password	Gibt das standardmäßige lokale Administratorkennwort für ein Betriebssystemsequenz-WIM-Image wie etwa Opware.OSSequence.Name gemäß der Definition in HP Server Automation an, wie beispielsweise „P@ssword1“.
Opware.OSSequence.Name	Gibt den Wert für den Namen der Betriebssystemsequenz gemäß der Definition in HP Server Automation an, wie beispielsweise „Windows 2008 WIM“.
Opware.ProvFail.Notify	(Optional) Gibt die Benachrichtigungs-E-Mail-Adresse für HP Server Automation für den Fall eines Bereitstellungsfehlers an, wie beispielsweise „provisionfail@lab.local“.
Opware.ProvFail.Owner	(Optional) Gibt den HP Server Automation-Benutzer an, an den das Besitzrecht übertragen wird, falls die Bereitstellung fehlschlägt.
Opware.ProvSuccess.Notify	(Optional) Gibt die Benachrichtigungs-E-Mail-Adresse für HP Server Automation im Falle einer erfolgreichen Bereitstellung an.
Opware.ProvSuccess.Owner	(Optional) Gibt den HP Server Automation-Benutzer an, an den das Besitzrecht im Falle einer erfolgreichen Bereitstellung übertragen wird.
Opware.Realm.Name	Gibt den Wert für den Bereichsnamen gemäß der Definition in HP Server Automation an, wie beispielsweise „Production“.
Opware.Register.Timeout	Gibt an, wie lange in Sekunden gewartet werden soll, bis die Erstellung eines Bereitstellungsauftrags abgeschlossen ist.
Opware.Server.Name	Gibt den vollqualifizierten Namen des HP Server Automation-Servers an.
Opware.Server.Username	Gibt den Benutzernamen an, der bereitgestellt wird, wenn eine Kennwortdatei im Agent-Verzeichnis erstellt wurde, wie beispielsweise „opswareadmin“. Für diesen Benutzernamen ist Administratorzugriff auf die HP Server Automation-Instanz erforderlich.
Opware.Software.Install	Legen Sie diese Eigenschaft auf „True“ fest, um HP Server Automation das Installieren von Software zu erlauben.

Benutzerdefinierte Eigenschaften P

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben P beginnen.

Tabelle 1-42. Tabelle mit benutzerdefinierten Eigenschaften – P

Eigenschaft	Beschreibung
Plugin.AdMachineCleanup.Delete	Legen Sie diese Eigenschaft auf „True“ fest, um die Konten gelöschter Maschinen zu entfernen, anstatt sie zu deaktivieren.
Plugin.AdMachineCleanup.Execute	Legen Sie diese Eigenschaft auf „True“ fest, um das Active Directory-Bereinigungs-Plug-In zu aktivieren. Standardmäßig wird das Konto jeder gelöschten Maschine deaktiviert.
Plugin.AdMachineCleanup.MoveToOu	Verschiebt das Konto von gelöschten Maschinen in eine neue Active Directory-Organisationseinheit. Der Wert ist die Organisationseinheit, in die Sie das Konto verschieben. Für diesen Wert ist das Format <i>ou=OU, dc=dc</i> erforderlich, wie beispielsweise „ou=trash,cn=computers,dc=lab,dc=local“.
Plugin.AdMachineCleanup.UserName	Gibt den Benutzernamen eines Active Directory-Kontos mit ausreichenden Rechten zum Ausführen von Active Directory-Aktionen wie beispielsweise das Löschen, Deaktivieren, Umbenennen oder Verschieben von Active Directory-Konten an. Für diesen Wert ist das Format <i>Domäne\Benutzername</i> erforderlich, wie beispielsweise „lab \administrator“. Diese Eigenschaft ist erforderlich, wenn der vRealize Automation Manager-Dienst nicht über diese Rechte in einer Domäne verfügt, was passieren kann, wenn Sie Maschinen in mehreren Domänen bereitstellen.
Plugin.AdMachineCleanup.Password	Gibt das Kennwort für die Eigenschaft <code>Plugin.AdMachineCleanup.UserName</code> an.
Plugin.AdMachineCleanup.Domain	Gibt den Active Directory-Domännennamen an, der das zu löschende Maschinenkonto enthält.
Plugin.AdMachineCleanup.RenamePrefix	Benennt die Konten von gelöschten Maschinen durch Hinzufügen eines Präfixes um. Dieser Wert ist die voranzustellende Präfixzeichenfolge, wie beispielsweise „destroyed_“.
Pxe.Clean.ScriptName	Gibt den Namen eines im vRealize Automation Model Manager installierten EPI PowerShell-Skripts an, das auf der Maschine nach deren Bereitstellung ausgeführt werden soll. Dieser Wert ist der Name, der dem Skript beim Hochladen in den Model Manager zugewiesen ist, wie beispielsweise <code>clean.ps1</code> .
Pxe.Setup.ScriptName	Gibt ein benutzerdefiniertes EPI PowerShell-Skript an, das auf der Maschine ausgeführt werden soll, bevor sie vom PXE Network Bootstrap-Programm gestartet wird. Dieser Wert ist der Name, der dem Skript beim Hochladen in den Model Manager zugewiesen ist, wie beispielsweise <code>setup.ps1</code> .

Benutzerdefinierte Eigenschaften R

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben R beginnen.

Tabelle 1-43. Tabelle mit benutzerdefinierten Eigenschaften – R

Eigenschaft	Beschreibung
ReservationPolicyID	Gibt nicht den Reservierungsrichtliniennamen, sondern die Reservierungsrichtlinien-ID an. Beispielsweise handelt es sich bei dem von der vRealize Orchestrator-Eigenschaft <code>getApplicableReservationPolicies</code> zurückgegebenen Namen um den Namen der Reservierungsrichtlinie und nicht um die ID der Reservierungsrichtlinie.

Benutzerdefinierte Eigenschaften S

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben S beginnen.

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S

Eigenschaft	Beschreibung
<p><i>SysPrep.Section.Key</i></p> <ul style="list-style-type: none"> ■ <i>SysPrep.GuiUnattended.AdminPassword</i> ■ <i>SysPrep.GuiUnattended.EncryptedAdminPassword</i> ■ <i>SysPrep.GuiUnattended.TimeZone</i> 	<p>Gibt Informationen an, die der SysPrep-Antwortdatei auf Maschinen während der WinPE-Phase der Bereitstellung hinzugefügt werden sollen. Bereits in der SysPrep-Antwortdatei vorhandene Informationen werden von diesen benutzerdefinierten Eigenschaften überschrieben. <i>Section</i> steht für den Namen des Abschnitts der SysPrep-Antwortdatei, wie beispielsweise „GuiUnattended“ oder „UserData“. <i>Key</i> steht für einen Schlüsselnamen im Abschnitt. Um beispielsweise als Zeitzone einer bereitgestellten Maschine Westpazifik (Normalzeit) festzulegen, definieren Sie die benutzerdefinierte Eigenschaft <i>GuiUnattended.UserData.TimeZone</i> und legen Sie „275“ als Wert fest.</p> <p>Eine vollständige Aufstellung der Abschnitte, Schlüssel und akzeptierten Werte finden Sie in der Dokumentation zum Systemvorbereitungsdienstprogramm (SysPrep) für Windows.</p> <p>Die folgenden <i>Section.Key</i>-Kombinationen können für die WIM-basierte Bereitstellung angegeben werden:</p> <ul style="list-style-type: none"> ■ <i>GuiUnattended</i> <ul style="list-style-type: none"> ■ <i>AdminPassword</i> ■ <i>EncryptedAdminPassword</i> ■ <i>TimeZone</i> ■ <i>UserData</i> <ul style="list-style-type: none"> ■ <i>ProductKey</i> ■ <i>FullName</i> ■ <i>ComputerName</i> ■ <i>OrgName</i> ■ <i>Identification</i> <ul style="list-style-type: none"> ■ <i>DomainAdmin</i> ■ <i>DomainAdminPassword</i> ■ <i>JoinDomain</i> ■ <i>JoinWorkgroup</i>
<i>Sysprep.Identification.DomainAdmin</i>	Gibt einen Benutzernamen mit Administratorzugriff auf die Zieldomäne in Active Directory an. Schließen Sie die Benutzerdomäne nicht in die Anmeldedaten ein, die Sie an vCloud Director oder vCloud Air senden.
<i>Sysprep.Identification.DomainAdminPassword</i>	Gibt das Kennwort für die Eigenschaft <i>Sysprep.Identification.DomainAdmin</i> an.
<i>Sysprep.Identification.JoinDomain</i>	Gibt den Namen der Domäne an, der in Active Directory beigetreten werden soll.

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
<code>Sysprep.Identification.JoinWorkgroup</code>	Gibt den Namen der Arbeitsgruppe an, der beigetreten werden soll, wenn keine Domäne verwendet wird.
<code>Sysprep.UserData.ComputerName</code>	Gibt einen Maschinennamen an, wie beispielsweise „lab-client005“.
<code>Sysprep.UserData.FullName</code>	Gibt den vollständigen Namen eines Benutzers an.
<code>Sysprep.UserData.OrgName</code>	Gibt den Organisationsnamen des Benutzers an.
<code>Sysprep.UserData.ProductKey</code>	Gibt den Windows-Product Key an.
<code>SCCM.Collection.Name</code>	Gibt den Namen der SCCM-Kollektion an, die die Aufgabenabfolge für die Betriebssystembereitstellung enthält.
<code>SCCM.CustomVariable.Name</code>	Gibt den Wert einer benutzerdefinierten Variablen an, wobei <i>Name</i> der Name einer benutzerdefinierten Variable ist, die für die SCCM-Aufgabenabfolge verfügbar gemacht werden soll, nachdem die bereitgestellte Maschine bei der SCCM-Kollektion registriert wurde. Dieser Wert wird durch die von Ihnen ausgewählte benutzerdefinierte Variable bestimmt. Wenn Ihre Integration dies erfordert, können Sie mithilfe von <code>SCCM.RemoveCustomVariablePrefix</code> das Präfix <code>SCCM.CustomVariable.</code> aus Ihrer benutzerdefinierten Variablen entfernen.
<code>SCCM.Server.Name</code>	Gibt den vollqualifizierten Domännennamen des SCCM-Servers an, auf dem die Kollektion gespeichert ist, wie beispielsweise „lab-sccm.lab.local“.
<code>SCCM.Server.SiteCode</code>	Gibt den Standortcode des SCCM-Servers an.
<code>SCCM.Server.UserName</code>	Gibt einen Benutzernamen mit Administratorzugriff auf den SCCM-Server an.
<code>SCCM.Server.Password</code>	Gibt das Kennwort für die Eigenschaft <code>SCCM.Server.UserName</code> an.
<code>SCCM.RemoveCustomVariablePrefix</code>	Legen Sie diese Eigenschaft auf <i>true</i> fest, um das Präfix <code>SCCM.CustomVariable.</code> aus benutzerdefinierten SCCM-Variablen zu entfernen, die Sie mithilfe der benutzerdefinierten Eigenschaft <code>SCCM.CustomVariable.Name</code> erstellt haben.
<code>Scvmm.Generation2</code>	Wenn wahr, gibt dies an, dass der Blueprint eine Generation-2-Maschine auf einer Hyper-V (SCVMM) 2012 R2-Ressource bereitstellen darf. Die Generation-2-Bereitstellung erfordert zudem, dass der Blueprint die Eigenschaftseinstellung <code>Hyperv.Network.Type = synthetic</code> enthält.

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
Snapshot.Policy.AgeLimit	<p>Legt den Altersgrenzwert in Tagen für Snapshots, die auf Maschinen angewendet werden können, fest. Diese Eigenschaft betrifft die vSphere-Bereitstellung.</p> <p>Wenn ein Snapshot den Altersgrenzwert überschreitet, ist die Option „Übernehmen“ nicht mehr verfügbar.</p> <p>Wenn der Grenzwert für das Snapshot-Alter erreicht wird, ist der Snapshot zwar weiterhin vorhanden, aber Sie können ihn nicht mehr wiederherstellen. Den Snapshot können Sie mit dem vSphere-Client löschen.</p>
Snapshot.Policy.Limit	<p>Legt die zulässige Anzahl von Snapshots pro Maschine fest. Die Standardeinstellung ist ein Snapshot pro Maschine. Diese Eigenschaft betrifft die vSphere-Bereitstellung. Wenn Sie diese Eigenschaft auf „0“ festlegen, wird die Blueprint-Option zum Erstellen eines Snapshots für alle Benutzer mit Ausnahme der Support- und Managerrollen ausgeblendet.</p> <p>Snapshots werden in einer hierarchischen Struktur dargestellt.</p> <ul style="list-style-type: none"> ■ Tiefe – der Maximalwert ist 31. ■ Breite – es gibt keinen Grenzwert.
software.agent.service.url	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für die vRealize Automation-Softwaredienst-API angegeben. Beispielsweise</p> <p>https://Private_IP:1443/software-service/api.</p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.ebs.url</code> und <code>agent.download.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie PAT oder NAT und die Portweiterleitung verwenden.</p>
software.agent.task.timeout.seconds	<p>Gibt das Zeitüberschreitungslimit für Softwareskripte, die auf Agents ausgeführt werden, in Sekunden an. Standardmäßig beträgt das Zeitüberschreitungslimit für Softwareskripte, die auf Agents ausgeführt werden, 6 Stunden.</p>

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
<code>software.ebs.url</code>	<p>Bei Verwendung der Portweiterleitung werden hiermit die private IP-Adresse Ihrer Amazon AWS-Tunnelmaschine sowie der Port für den vRealize Automation-Ereignisbrokerdienst angegeben. Beispielsweise <code>https://Private_IP:1443/event-broker-service/api</code>.</p> <p>Sie können diese Eigenschaft in Kombination mit <code>software.agent.service.url</code> und <code>agent.download.url</code> zu einer Reservierung oder zum Computing-Ressourcen-Endpoint hinzufügen. Mithilfe dieser Eigenschaft können Sie auch eine private Adresse und einen Port angeben, wenn Sie PAT oder NAT und die Portweiterleitung verwenden.</p>
<code>software.http.proxyHost</code>	<p>Gibt den Hostnamen oder die Adresse des Proxy-Servers an.</p> <p>Sie müssen sowohl <code>software.http.proxyHost</code> als auch <code>software.http.proxyPort</code> angeben, damit Softwareinhalteigenschaften den Proxy-Server verwenden.</p> <hr/> <p>Hinweis Mithilfe der Software-Proxy-Einstellungen können Sie für eine Softwarekomponente den Wert eines Inhaltseigenschaftstyps definieren. Inhaltseigenschaften sind URLs, die vom Agent heruntergeladen werden. Der Agent verwendet die Variable als Dateipfad zur lokal heruntergeladenen Datei. Mithilfe der Software-Proxy-Einstellungen können Sie jedoch über den Proxy-Host anstatt über die URL herunterladen.</p>
<code>software.http.proxyPassword</code>	<p>Gibt das Kennwort für den Benutzernamen an, mit dem die Authentifizierung beim Proxy-Server durchgeführt werden soll. Verwenden Sie dies in Kombination mit <code>software.http.proxyUser</code>.</p> <p>Die Einstellung <code>software.http.proxyPassword</code> ist erforderlich, wenn Sie die Einstellung <code>software.http.proxyUser</code> verwenden.</p> <hr/> <p>Hinweis Mithilfe der Software-Proxy-Einstellungen können Sie für eine Softwarekomponente den Wert eines Inhaltseigenschaftstyps definieren. Inhaltseigenschaften sind URLs, die vom Agent heruntergeladen werden. Der Agent verwendet die Variable als Dateipfad zur lokal heruntergeladenen Datei. Mithilfe der Software-Proxy-Einstellungen können Sie jedoch über den Proxy-Host anstatt über die URL herunterladen.</p>

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
software.http.proxyPort	<p>Gibt die Portnummer des Proxy-Servers an. Sie müssen sowohl <code>software.http.proxyHost</code> als auch <code>software.http.proxyPort</code> angeben, damit Softwareinhaltseigenschaften den Proxy-Server verwenden. Es gibt keinen Standardwert für <code>software.http.proxyPort</code>.</p> <p>Hinweis Mithilfe der Software-Proxy-Einstellungen können Sie für eine Softwarekomponente den Wert eines Inhaltseigenschaftstyps definieren. Inhaltseigenschaften sind URLs, die vom Agent heruntergeladen werden. Der Agent verwendet die Variable als Dateipfad zur lokal heruntergeladenen Datei. Mithilfe der Software-Proxy-Einstellungen können Sie jedoch über den Proxy-Host anstatt über die URL herunterladen.</p>

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
software.http.proxyUser	<p>Gibt den Benutzernamen an, mit dem die Authentifizierung beim Proxy-Server durchgeführt werden soll. Verwenden Sie dies in Kombination mit software.http.proxyPassword.</p> <p>Die Einstellung software.http.proxyUser ist optional. Die Einstellung software.http.proxyPassword ist erforderlich, wenn Sie die Einstellung software.http.proxyUser verwenden.</p> <p>Hinweis Mithilfe der Software-Proxy-Einstellungen können Sie für eine Softwarekomponente den Wert eines Inhaltseigenschaftstyps definieren. Inhaltseigenschaften sind URLs, die vom Agent heruntergeladen werden. Der Agent verwendet die Variable als Dateipfad zur lokal heruntergeladenen Datei. Mithilfe der Software-Proxy-Einstellungen können Sie jedoch über den Proxy-Host anstatt über die URL herunterladen.</p>
software.http.noProxyList	<p>Gibt eine Liste der Hosts und optionalen Ports an, die den proxyHost nicht verwenden dürfen. Mit der ursprünglichen Inhaltseigenschaft erfolgt der Download direkt über URLs, die mit den in der Liste aufgeführten Elementen übereinstimmen. Die Einstellung software.http.noProxyList ist nur anwendbar, wenn der Proxy-Server konfiguriert ist. Beispielsweise für die folgende kommagetrennte Liste:</p> <pre>"buildweb.eng.vmware.com,confluence.eng.vmware.com:443,*.eng.vmware.com:80"</pre> <p>Es gelten die folgenden Anweisungen:</p> <ul style="list-style-type: none"> ■ Keine URL, deren HOST „buildweb.eng.vmware.com“ lautet, darf den Proxy-Server verwenden. ■ Keine URL, deren HOST „confluence.eng.vmware.com“ und deren PORT 443 lautet, darf den Proxy-Server verwenden.

Tabelle 1-44. Tabelle mit benutzerdefinierten Eigenschaften – S (Fortsetzung)

Eigenschaft	Beschreibung
	<ul style="list-style-type: none"> Keine URL, deren HOST aus dem Namespace „eng.vmware.com“ stammt und deren PORT 80 lautet, darf den Proxy-Server verwenden. <p>Hinweis Mithilfe der Software-Proxy-Einstellungen können Sie für eine Softwarekomponente den Wert eines Inhaltseigenschaftstyps definieren. Inhaltseigenschaften sind URLs, die vom Agent heruntergeladen werden. Der Agent verwendet die Variable als Dateipfad zur lokal heruntergeladenen Datei. Mithilfe der Software-Proxy-Einstellungen können Sie jedoch über den Proxy-Host anstatt über die URL herunterladen.</p>

Benutzerdefinierte Eigenschaften V

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben V beginnen.

Obwohl die allgemeine Unterstützung für vCloud Networking and Security beendet wurde, sind die benutzerdefinierten VCNS-Eigenschaften nach wie vor zu NSX-Zwecken gültig. Weitere Informationen hierzu finden Sie im [Knowledgebase-Artikel 2144733](#).

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V

Eigenschaft	Beschreibung
VbScript.PreProvisioning.Name	Gibt den vollständigen Pfad eines Visual Basic-Skripts an, das vor der Bereitstellung einer Maschine ausgeführt werden soll. Beispielsweise %System-Drive%\Programme (x86)\VMware\VCAC Agents\EPI_Agent\Scripts\SendEmail.vbs. Die Skriptdatei muss in dem System vorhanden sein, auf dem der EPI-Agent für Visual Basic-Skripts installiert ist.
VbScript.PostProvisioning.Name	Gibt den vollständigen Pfad eines Visual Basic-Skripts an, das nach der Bereitstellung einer Maschine ausgeführt werden soll. Beispielsweise %System-Drive%\Programme (x86)\VMware\VCAC Agents\EPI_Agent\Scripts\SendEmail.vbs. Die Skriptdatei muss in dem System vorhanden sein, auf dem der EPI-Agent für Visual Basic-Skripts installiert ist.
VbScript.UnProvisioning.Name	Gibt den vollständigen Pfad eines Visual Basic-Skripts an, das beim Löschen einer Maschine ausgeführt werden soll. Beispielsweise %System-Drive%\Programme (x86)\VMware\VCAC Agents\EPI_Agent\Scripts\SendEmail.vb. Die Skriptdatei muss in dem System vorhanden sein, auf dem der EPI-Agent für Visual Basic-Skripts installiert ist.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VCloud.Lease.Sync.TimeBufferMins	<p>Gibt einen ganzzahligen Schwellenwert für eine Computing-Ressource an, sodass die Lease-Synchronisierung zwischen vCloud Director und vRealize Automation nur für mit vCloud Director oder vCloud Air bereitgestellte Maschinen ausgeführt wird, die in vCloud Director oder vCloud Air in diesem Zeitraum ablaufen sollen. Falls ein Konflikt festgestellt wird, wird der Leasewert synchronisiert und an die in vRealize Automation definierte Leasedauer angepasst. Der Standardwert für VCloud.Lease.Sync.TimeBufferMins beträgt 720 Minuten (12 Stunden). Wenn VCloud.Lease.Sync.TimeBufferMins nicht vorhanden ist, wird der Standardwert verwendet. Wenn beispielsweise die Standardwerte verwendet werden, führt vRealize Automation den Workflow zur Lease-Synchronisierungsprüfung alle 45 Minuten aus, was dem Workflow-Standardwert entspricht. Nur die Leases von Maschinen, die innerhalb von 12 Stunden ablaufen sollen, werden geändert und an die in vRealize Automation definierte Leasedauer angepasst.</p>
VCloud.Owner.UseEndpointAccount	<p>Legen Sie diese Eigenschaft auf „True“ fest, um das Endpoint-Konto als vCloud Air- oder vCloud Director-Maschinenbesitzer für Bereitstellungs- und Importvorgänge zuzuweisen. Für Vorgänge zum Ändern des Besitzers wird der Besitzer nicht auf dem Endpoint geändert. Wenn diese Option nicht angegeben ist oder auf „False“ festgelegt ist, ist der vRealize Automation-Besitzer der Maschinenbesitzer.</p>
VCloud.Template.MakeIdenticalCopy	<p>Legen Sie diese Eigenschaft auf „True“ fest, um eine identische Kopie der vCloud Air- oder vCloud Director-Vorlage für die Maschinenbereitstellung zu klonen. Die Maschine wird als identische Kopie der Vorlage bereitgestellt. In der Vorlage angegebene Einstellungen, einschließlich des Speicherpfads, ersetzen im Blueprint angegebene Einstellungen. Die einzigen Änderungen gegenüber der Vorlage sind die Namen der geklonten Maschinen, die anhand des im Blueprint angegebenen Maschinenpräfixes generiert werden.</p> <p>vCloud Air- oder vCloud Director-Maschinen, die als identische Kopien bereitgestellt werden, können Netzwerke und Speicherprofile verwenden, die in der vRealize Automation-Reservierung nicht verfügbar sind. Um nicht berücksichtigte Reservierungszuteilungen zu vermeiden, sollten Sie sicherstellen, dass das in der Vorlage angegebene Speicherprofil oder Netzwerk in der Reservierung verfügbar ist.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VCNS.LoadBalancerEdgePool.Names.name</code>	<p>Gibt die NSX-Lastausgleichspools an, denen die virtuelle Maschine während der Bereitstellung zugewiesen wird. Die virtuelle Maschine wird allen Dienstports von allen angegebenen Pools zugewiesen. Bei dem Wert handelt es sich um einen <i>Edge/Pool</i>-Namen oder eine durch Kommas getrennte Liste von <i>Edge/Pool</i>-Namen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt.</p> <hr/> <p>Hinweis Sie können die IP-Adresse einer Maschine einem vorhandenen Lastausgleichsdienst mithilfe der benutzerdefinierten Eigenschaft <code>VCNS.LoadBalancerEdgePool.Names</code> hinzufügen. vRealize Automation und NSX verwenden das erste Element des angegebenen Edge-Lastausgleichsdienst-Pools, um den Port des neuen Elements zu ermitteln und die Porteinstellungen zu überwachen. Für NSX 6.2 muss jedoch die Porteinstellung für das Element nicht angegeben werden. Um Bereitstellungsfehler zu vermeiden, wenn <code>VCNS.LoadBalancerEdgePool.Names</code> mit NSX 6.2 zum Hinzufügen einer Maschine zu einem bestehenden Pool verwendet wird, müssen Sie einen Portwert für das erste Element des Lastausgleichsdienst-Pools in NSX angeben.</p> <hr/> <p>Durch Anfügen eines Namens können Sie mehrere Versionen einer benutzerdefinierten Eigenschaft erstellen. Beispielsweise werden mit den folgenden Eigenschaften Lastausgleichspools aufgelistet, die zur allgemeinen Verwendung und für Maschinen mit hohen, mäßigen und niedrigen Leistungsanforderungen eingerichtet sind:</p> <ul style="list-style-type: none"> ■ <code>VCNS.LoadBalancerEdgePool.Names</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.moderate</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.high</code> ■ <code>VCNS.LoadBalancerEdgePool.Names.low</code>
<code>VCNS.SecurityGroup.Names.name</code>	<p>Gibt die NSX-Sicherheitsgruppe(n) an, der bzw. denen die virtuelle Maschine während der Bereitstellung zugewiesen wird. Bei dem Wert handelt es sich um einen Sicherheitsgruppennamen oder eine durch Kommas getrennte Liste von Sicherheitsgruppennamen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt.</p> <p>Durch Anfügen eines Namens können Sie mehrere Versionen der Eigenschaft erstellen, die separat oder kombiniert verwendet werden können. Beispielsweise können mit den folgenden Eigenschaften Sicherheitsgruppen aufgelistet werden, die zur allgemeinen Verwendung, für die Vertriebsmitarbeiter und für den Support gedacht sind:</p> <ul style="list-style-type: none"> ■ <code>VCNS.SecurityGroup.Names</code> ■ <code>VCNS.SecurityGroup.Names.sales</code> ■ <code>VCNS.SecurityGroup.Names.support</code>
<code>VCNS.SecurityGroup.Names.blueprint_name</code>	<p>Bei Verwendung von NSX wird hiermit der Edge-Pool angegeben, dem der Blueprint zugeordnet werden soll.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VCNS.SecurityTag.Names.name</code>	<p>Gibt das NSX-Sicherheitstag bzw. die -Sicherheitstags an, dem bzw. denen die virtuelle Maschine während der Bereitstellung zugeordnet wird. Bei dem Wert handelt es sich um einen Sicherheits-Tag-Namen oder eine durch Kommas getrennte Liste von Sicherheits-Tag-Namen. Bei Namen wird die Groß- und Kleinschreibung berücksichtigt.</p> <p>Durch Anfügen eines Namens können Sie mehrere Versionen der Eigenschaft erstellen, die separat oder kombiniert verwendet werden können. Beispielsweise können mit den folgenden Eigenschaften Sicherheits-Tags aufgelistet werden, die zur allgemeinen Verwendung, für die Vertriebsmitarbeiter und für den Support gedacht sind:</p> <ul style="list-style-type: none"> ■ <code>VCNS.SecurityTag.Names</code> ■ <code>VCNS.SecurityTag.Names.sales</code> ■ <code>VCNS.SecurityTag.Names.support</code>
<code>VirtualMachine.Admin.UseGuestAgent</code>	<p>Wenn der Gast-Agent als Dienst in einer Vorlage für das Klonen installiert ist, legen Sie diese Eigenschaft im Maschinen-Blueprint auf „True“ fest, um den Gast-Agent-Dienst auf Maschinen, die anhand dieser Vorlage geklont werden, zu aktivieren. Beim Starten der Maschine wird der Gast-Agent-Dienst gestartet. Legen Sie diese Eigenschaft auf „False“ fest, um den Gast-Agent zu deaktivieren. Mit der Einstellung „False“ verwendet der erweiterte Klon-Workflow den Gast-Agent nicht für Aufgaben des Gastbetriebssystems, wodurch dessen Funktionalität auf <code>VMwareCloneWorkflow</code> reduziert wird. Wenn diese Option nicht angegeben ist oder auf einen anderen Wert als „False“ festgelegt ist, sendet der erweiterte Klon-Workflow Arbeitselemente an den Gast-Agent.</p> <p>Diese Eigenschaft gilt nicht für die Bereitstellung von Amazon Web Services.</p>
<code>VirtualMachine.Admin.NameCompletion</code>	<p>Gibt den Domänennamen an, der in den vollqualifizierten Domänennamen der Maschine einbezogen werden soll, den die RDP- oder SSH-Dateien für die Benutzeroberflächenoptionen Verbindungsherstellung mithilfe von RDP oder Verbindungsherstellung mithilfe von SSH generieren. Legen Sie beispielsweise „myCompany.com“ als Wert fest, um den vollqualifizierten Domänennamen <code>my-machine-name.myCompany.com</code> in der RDP- oder SSH-Datei zu generieren.</p>
<code>VirtualMachine.Admin.ConnectAddress</code>	<p>Gibt die Adresse der RDP-Verbindung der Maschine an, auf die eine RDP-Datei heruntergeladen wird, wenn die Benutzeroberflächenoption Verbindungsherstellung mithilfe von RDP verwendet wird oder an automatische E-Mails angehängt wird. Verwenden Sie diese Option nur in einem Blueprint oder einer Eigenschaftsgruppe, wenn der Benutzer zu einer Eingabe aufgefordert werden soll und Sie keinen Standardwert angegeben haben.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.ConnectAddress.Regex</code>	<p>Wird von einem vRealize Automation-Administrator für die Definition eines regulären Ausdrucks verwendet, um eine passende IP-Adresse für Terminalverbindungen wie etwa eine RDP-Verbindung zuzuordnen. Wenn eine passende IP-Adresse gefunden wird, wird sie unter der benutzerdefinierten Eigenschaft <code>VirtualMachine.Admin.ConnectAddress</code> gespeichert. Andernfalls wird die erste verfügbare IP-Adresse zugewiesen.</p> <p>Beispielsweise kann durch Festlegen des Eigenschaftswerts auf <code>10.10.0.</code> eine IP-Adresse in einem <code>10.10.0.*</code>-Subnetz ausgewählt werden, das der virtuellen Maschine zugewiesen ist. Wenn das Subnetz nicht zugewiesen wurde, wird die Eigenschaft ignoriert. Diese Eigenschaft ist für die Verwendung mit OpenStack verfügbar.</p>
<code>VirtualMachine.Admin.ThinProvision</code>	<p>Legt fest, ob Thin Provisioning auf ESX-Computing-Ressourcen verwendet wird. Festplatten-bereitstellung wird vom zugrunde liegenden Speicher abstrahiert. Legen Sie diese Eigenschaft auf „True“ fest, um Thin Provisioning zu verwenden. Legen Sie diese Eigenschaft auf „False“ fest, um die Standardbereitstellung zu verwenden. Diese Eigenschaft dient für die virtuelle Bereitstellung.</p>
<code>VirtualMachine.Admin.CustomizeGuestOSDelay</code>	<p>Gibt an, wie lange nach Abschluss der Anpassung gewartet werden soll, bevor die Anpassung des Gastbetriebssystems gestartet wird. Für diesen Wert ist das Format HH:MM:SS erforderlich. Wenn dieser Wert nicht festgelegt wird, wird der Standardwert von einer Minute (00:01:00) verwendet. Wenn Sie diese benutzerdefinierte Eigenschaft nicht angeben, schlägt die Bereitstellung möglicherweise fehl, falls die virtuelle Maschine neu gestartet wird, bevor Arbeitselemente des Gast-Agents abgeschlossen sind.</p> <p>Diese Eigenschaft gilt nicht für die Bereitstellung von Amazon Web Services.</p>
<code>VirtualMachine.Admin.Datastore.Cluster.ResourceLeaseDurationSec</code>	<p>Gibt bei Bereitstellung auf mehreren VMs und Verwendung von SDRS einen Wert in Sekunden (im Bereich von 30 bis 3600) für die Reservierung von Speicherressourcen während des API-Aufrufs <code>RecommendDataStore</code> an. Sie können diese Eigenschaft einer Business-Gruppe oder einem Blueprint oder beim Anfordern einer Bereitstellung hinzufügen. Die Lease-Sperre wird nicht auf alle Datenspeicher im Speicher-Cluster angewendet, sondern nur auf den Datenspeicher, der von der Bereitstellung verwendet wird. Die Lease-Sperre wird bei vollständiger oder fehlgeschlagener Bereitstellung freigegeben.</p> <p>Wenn nicht angegeben, wird keine Sperre auf die Speicherressourcen zum Bereitstellungszeitpunkt angewendet. Im Hinblick auf die Größe des Arbeitsspeichers kann das gleichzeitige Anfordern von mehr als 10 VMs zu Bereitstellungsfehlern führen.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.NetworkInterfaceType</code>	<p>Gibt den Netzwerkadaptertyp an, der vom Gastbetriebssystem unterstützt und emuliert wird. Hiermit erstellen Sie eine neue virtuelle Maschine und weisen einen bestimmten Adaptertyp für einen Vorlagenklonvorgang zu. Ändern Sie hiermit die Netzwerkeinstellungen einer neu bereitgestellten virtuellen Maschine. Die folgenden Optionen sind verfügbar:</p> <ul style="list-style-type: none"> ■ E1000 (Standard) ■ VirtIO ■ RTL8139 ■ RTL8139 VirtIO
<code>VirtualMachine.Admin.Name</code>	<p>Gibt den generierten Maschinennamen für vSphere an, wie beispielsweise „CodyVM01“. Legen Sie beim Erstellen von benutzerdefinierten Workflows oder Plug-Ins für die Anpassung des Namens einer virtuellen Maschine diese Eigenschaft in Übereinstimmung mit dem Namen der virtuellen Maschine fest. Hierbei handelt es sich um eine interne Eingabeeigenschaft für den Agent zur Benennung der virtuellen Maschine.</p> <p>Hinweis Diese Eigenschaft ist nur für vSphere gedacht.</p> <p>Der im Blueprint angegebene Wert hat keine Auswirkungen auf diese Eigenschaft. Diese Eigenschaft ist nicht dazu gedacht, den Benutzer zu einer Eingabe aufzufordern. Verwenden Sie zu diesem Zweck die Eigenschaft <code>HostName</code>. Wenn die Eigenschaft zur Laufzeit festgelegt wird, stimmt der im Hypervisor erstellte Containername möglicherweise nicht mit dem Datensatznamen des Elements überein.</p>
<code>VirtualMachine.Admin.UUID</code>	Gibt die UUID der Maschine an. Der Gast-Agent zeichnet den Wert auf, wenn die Maschine erstellt wird. Dieser Wert ist schreibgeschützt. Der Wert im Blueprint oder in der Eigenschaftsgruppe hat keine Auswirkungen auf diese Eigenschaft.
<code>VirtualMachine.Admin.AgentID</code>	Gibt die UUID des Gast-Agents an. Der Gast-Agent zeichnet den Wert auf, wenn die Maschine erstellt wird. Dieser Wert ist schreibgeschützt. Der Wert im Blueprint oder in der Eigenschaftsgruppe hat keine Auswirkungen auf diese Eigenschaft.
<code>VirtualMachine.Admin.Owner</code>	Gibt den Benutzernamen des Maschinenbesitzers an.
<code>VirtualMachine.Admin.Approver</code>	Gibt den Benutzernamen des Gruppenmanagers an, der die Maschinenanforderung genehmigt hat.
<code>VirtualMachine.Admin.Description</code>	Gibt die Beschreibung der Maschine an, die vom Besitzer oder einem Administrator eingegeben oder geändert wurde.
<code>VirtualMachine.Admin.EncryptPasswords</code>	Wenn Sie diese Eigenschaft auf „True“ festlegen, werden die Administrator Kennwörter verschlüsselt.
<code>VirtualMachine.Admin.AdministratorEmail</code>	Gibt die Manager-E-Mail-Adressen oder Active Directory-Konten für die Business-Gruppe des Bereitstellungs-Blueprints an. Mehrere E-Mail-Adressen werden durch ein Komma getrennt, wie beispielsweise <code>AlbertAdmin@VMware.com,WeiLeeMgr@VMware.com</code> .

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.TotalDiskUsage</code>	Gibt den von der Maschine insgesamt verwendeten Festplattenspeicherplatz in GB an, einschließlich aller Festplatten gemäß den Eigenschaften <code>VirtualMachine.DiskN.Size</code> und der Auslagerungsdatei gemäß der Eigenschaft <code>VMware.Memory.Reservation</code> . Sie geben den Wert in GB an, der Festplattenspeicher wird jedoch von vRealize Automation in MB gespeichert.
<code>VirtualMachine.Admin.Hostname</code>	<p>Informiert den Administrator, welcher Host für die Bereitstellung der Maschine auf dem Endpoint verwendet wird. Der angegebene Wert wird auf der Maschine implementiert und während der Datenerfassung aufgefüllt. Wenn beispielsweise die Computing-Ressource einer Maschine geändert wird, aktualisiert ein Proxy-Agent den Wert für die Eigenschaft <code>VirtualMachine.Admin.Hostname</code> der Maschine.</p> <p>Hinweis Dies ist eine interne Ausgabeeigenschaft des Agents, die während des Datenerfassungsvorgangs aufgefüllt wird und den Host, auf dem sich eine Maschine befindet, identifiziert.</p>
<code>VirtualMachine.Admin.ClusterName</code>	<p>Informiert den Administrator, welcher Cluster die Computing-Ressource für die zu verwendende Maschine enthält.</p> <p>Hinweis Dies ist eine interne Ausgabeeigenschaft des Agents, die während des Datenerfassungsvorgangs aufgefüllt wird und den Cluster, in dem sich eine Maschine befindet, identifiziert.</p>
<code>VirtualMachine.Admin.ApplicationID</code>	Listet die Anwendungs-IDs auf, die einer Maschine zugewiesen werden können.
<code>VirtualMachine.Admin.AddOwnerToAdmins</code>	<p>Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um den Besitzer der Maschine, gemäß der Eigenschaft <code>VirtualMachine.Admin.Owner</code>, zur lokalen Administratorengruppe auf der Maschine hinzuzufügen.</p> <p>Diese Eigenschaft ist für die Bereitstellung mittels Klonen nicht verfügbar.</p>
<code>VirtualMachine.Admin.AllowLogin</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um den Maschinenbesitzer zur Gruppe der lokalen Remotedesktopbenutzer, gemäß der Eigenschaft <code>VirtualMachine.Admin.Owner</code> , hinzuzufügen.
<code>VirtualMachine.Admin.DiskInterfaceType</code>	<p>Gibt den Festplattentreibertyp an. Die folgenden Festplattentreiber werden unterstützt:</p> <ul style="list-style-type: none"> ■ IDE (Standard) ■ VirtIO <p>Diese Eigenschaft dient für die virtuelle Bereitstellung.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Admin.EagerZero</code>	<p>Durch Festlegen dieser Eigenschaft auf „True“ werden die Festplatten der Maschine mithilfe des VMware-Bereitstellungsformats Eager-Zero bereitgestellt.</p> <p>Thick-Provision Eager-Zero erstellt einen Typ einer virtuellen Festplatte im Thick-Format, der Clusterfunktionen, wie z. B. Fault Tolerance, unterstützt. Der Speicherplatz, den die virtuelle Festplatte benötigt, wird beim Erstellen zugewiesen. Im Gegensatz zum Flat-Format werden die auf dem physischen Gerät verbleibenden Daten durch Nullbyte ersetzt („zeroed out“), wenn die virtuelle Festplatte erstellt wird. Das Erstellen von Festplatten in diesem Format kann wesentlich länger dauern als das Erstellen anderer Festplattentypen.</p>
<code>VirtualMachine.Admin.ForceHost</code>	<p>Gibt den Namen des ESX-Hosts an. Diese Eigenschaft wird nur berücksichtigt, wenn <code>VirtualMachine.Admin.HostSelectionPolicy</code> auf <code>EXACT_MATCH</code> festgelegt ist.</p> <p>Hinweis Diese Eigenschaft ist nur für vSphere gedacht.</p> <p>Bei der Bereitstellung für einen vSphere-Cluster können Sie mithilfe der Eigenschaft <code>VirtualMachine.Admin.ForceHost</code> den Host angeben, auf dem eine Maschine bereitgestellt werden soll. Diese Eigenschaft wird nur verwendet, wenn DRS für den Cluster nicht auf „Automatisch“ festgelegt ist. Wenn für den Cluster DRS aktiviert ist und „Automatisch“ festgelegt ist, verlagert vSphere die bereitgestellte Maschine beim Neustart.</p>
<code>VirtualMachine.Admin.HostSelectionPolicy</code>	<p>Legen Sie diese Eigenschaft optional auf <code>EXACT_MATCH</code> fest, um das Platzieren der Maschine auf dem mit der Eigenschaft <code>VirtualMachine.Admin.ForceHost</code> angegebenen Host zu erzwingen. Wenn der Host nicht verfügbar ist, führt die Anforderung zu einem Fehler. Wenn kein Host angegeben wird, wird der nächste geeignete und verfügbare Host ausgewählt. Wenn diese Eigenschaft auf <code>EXACT_MATCH</code> festgelegt wird, tritt ein Fehler auf, wenn der angegebene Host nicht über ausreichend Arbeitsspeicher verfügt oder er sich im Wartungsmodus befindet.</p> <p>Hinweis Diese Eigenschaft betrifft nur vSphere.</p>
<code>VirtualMachine.Agent.CopyToDisk</code>	<p>Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um die ausführbare Datei für den Gast-Agent in das Verzeichnis <code>%System-Drive%\VRM\Build\Bin</code> auf der Festplatte der Maschine zu kopieren.</p>
<code>VirtualMachine.Agent.GuiRunOnce</code>	<p>Legen Sie diese Eigenschaft auf „True“ fest, um die Ausführung des Gast-Agents in den Abschnitt für einmalige Ausführung in <code>SysPrep.inf</code> einzubeziehen. Legen Sie diese Eigenschaft auf „False“ fest, damit der Linux-Agent den Bereitstellungsworkflow beendet.</p>
<code>VirtualMachine.Agent.Reboot</code>	<p>Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um anzugeben, dass der Gast-Agent die Maschine nach der Installation des Gastbetriebssystems neu startet.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.CDRom.Attach</code>	Legen Sie diese Eigenschaft auf „False“ fest, um die Maschine ohne CD-ROM-Gerät bereitzustellen. Der Standardwert ist „True“.
<code>VirtualMachine.CPU.Count</code>	<p>Gibt die Anzahl der einer Maschine zugeteilten CPUs an (z. B. 2). Der Standardwert ist der von der CPU-Einstellung im Blueprint angegebene Wert.</p> <p>Hinweis Dieser benutzerdefinierte Eigenschaftswert wird durch den CPU-Wert im Blueprint überschrieben, wenn die Maschine erstmalig bereitgestellt wird.</p>
<code>VirtualMachine.Customize.WaitComplete</code>	<p>Legen Sie diese Eigenschaft auf „True“ fest, um zu verhindern, dass der Bereitstellungsworkflow Arbeitselemente an den Gast-Agent sendet, bevor alle Anpassungen abgeschlossen sind. Legen Sie diese Eigenschaft auf „False“ fest, um die Erstellung von Arbeitselementen zu ermöglichen, bevor die Anpassung abgeschlossen ist.</p> <p>Diese Eigenschaft gilt nicht für die Bereitstellung von Amazon Web Services.</p>
<code>VirtualMachine.Core.Count</code>	<p>Wenn Sie einen Wert größer als 0 festlegen, wird die Anzahl der Kerne pro Socket beim Bereitstellen der virtuellen Maschine angegeben.</p> <p>Mithilfe dieser Eigenschaften können Sie in einem Blueprint die Kerne pro virtuellem Socket oder die Gesamtanzahl der Sockets angeben. Beispielsweise können Ihre Lizenzbestimmungen die pro Socket lizenzierte Software beschränken, oder aber verfügbare Betriebssysteme erkennen nur eine bestimmte Anzahl von Sockets und zusätzliche CPUs müssen als zusätzliche Kerne bereitgestellt werden.</p>
<code>VirtualMachine.DiskN.Letter</code>	<p>Gibt den Laufwerkbuchstaben oder Mount-Punkt der Festplatte <i>N</i> einer Maschine an. Der Standardwert ist C. Um beispielsweise den Buchstaben D für die Festplatte 1 anzugeben, definieren Sie die benutzerdefinierte Eigenschaft als <code>VirtualMachine.Disk1.Letter</code> und geben Sie den Wert „D“ ein. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent gibt dieser Wert den Laufwerkbuchstaben oder Einhängepunkt an, unter dem die zusätzliche Festplatte <i>N</i> vom Gast-Agent im Gastbetriebssystem gemountet wird.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.IsFixed</code>	<p>Deaktiviert die Bearbeitung einer bestimmten Festplatte bei der Neukonfiguration einer Maschine. Legen Sie diese Eigenschaft auf „True“ fest, um die Anzeige der Kapazitätsbearbeitungsoption für ein bestimmtes Volume zu deaktivieren. Für den Wert „True“ wird die Groß-/Kleinschreibung berücksichtigt. Der Wert <i>N</i> ist der 0-basierte Index der Festplatte.</p> <p>Alternativ können Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.DiskN.IsFixed</code> in der <code>VirtualMachineProperties</code>-Tabelle der Datenbank auf „True“ festlegen oder mithilfe der Repository-API einen URI-Wert angeben, wie beispielsweise <code>.../Repository/Data/ManagementModelEntities.svc/VirtualMachines(guid'60D93A8A-F541-4CE0-A6C6-78973AC0F1D2')/VirtualMachineProperties</code>.</p>
<code>VirtualMachine.DiskN.Label</code>	Gibt die Bezeichnung für die Festplatte <i>N</i> einer Maschine an. Für die Festplattenbezeichnung sind maximal 32 Zeichen zulässig. Festplatten müssen sequenziell nummeriert werden. Bei Verwendung in Verbindung mit einem Gast-Agent wird hiermit die Bezeichnung der Festplatte <i>N</i> einer Maschine im Gastbetriebssystem angegeben.
<code>VirtualMachine.DiskN.Active</code>	Legen Sie diese Eigenschaft auf „True“ (Standardwert) fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine aktiv ist. Legen Sie diese Eigenschaft auf „False“ fest, um anzugeben, dass die Festplatte <i>N</i> der Maschine nicht aktiv ist.
<code>VirtualMachine.DiskN.FS</code>	Für die Verwendung mit Windows-Gast-Agent (gugent). Gibt das Dateisystem der Festplatte <i>N</i> der Maschine an. Mögliche Optionen sind NTFS (Standardwert), FAT und FAT32. Ein Beispiel für die Verwendung finden Sie im Windows-Agent-Skript <code>10_setupdisks.bat</code> .
<code>VirtualMachine.DiskN.FileSystem</code>	Für die Verwendung mit Linux-Gast-Agent (gugent). Gibt das Dateisystem der Festplatte <i>N</i> der Maschine an. Die Optionen sind ext3, ext4 und XFS. Ein Beispiel für die Verwendung finden Sie im Linux-Agent-Skript <code>30_DiskSetup.sh</code> .
<code>VirtualMachine.DiskN.Percent</code>	Gibt an, welcher Prozentsatz der Festplatte <i>N</i> von einem Gast-Agent für die Verwendung durch die Maschine formatiert werden soll. Diese Maschine kann den restlichen Bereich der Festplatte nicht verwenden.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.DiskN.StorageReservationPolicy</code>	Gibt die Speicherreservierungsrichtlinie an, die für die Suche nach Speicher für die Festplatte <i>N</i> verwendet werden soll. Hiermit wird außerdem die benannte Speicherreservierungsrichtlinie einem Volume zugewiesen. Um diese Eigenschaft zu verwenden, ersetzen Sie die Volume-Nummer im Eigenschaftsnamen durch <i>N</i> und geben Sie den Namen einer Speicherreservierungsrichtlinie als Wert an. Diese Eigenschaft entspricht dem Namen der im Blueprint angegebenen Speicherreservierungsrichtlinie. Festplatten müssen sequenziell nummeriert werden. Diese Eigenschaft kann für alle virtuellen Reservierungen und vCloud-Reservierungen verwendet werden. Für physische, Amazon- oder OpenStack-Reservierungen kann diese Eigenschaft nicht verwendet werden. Mithilfe von <code>VirtualMachine.DiskN.StorageReservationPolicyMode</code> können Sie verhindern, dass die Bereitstellung im Fall von unzureichendem Speicherplatz in den Datenspeichern in einer Speicherreservierungsrichtlinie fehlschlägt. Verwenden Sie diese benutzerdefinierte Eigenschaft, sodass vRealize Automation einen Datenspeicher außerhalb der angegebenen Speicherreservierungsrichtlinie auswählen kann, wenn in den Datenspeichern in der Richtlinie nicht ausreichend Speicherplatz verbleibt.
<code>VirtualMachine.DiskN.StorageReservationPolicyMode</code>	Teilt die Festplatte <i>N</i> der geeignetsten verfügbaren Speicherreservierungsrichtlinie zu.
<code>VirtualMachine.DiskN.Storage</code>	Gibt den Datenspeicher an, auf dem die Festplatte <i>N</i> der Maschine platziert werden soll, wie beispielsweise <code>DATASTORE01</code> . Mithilfe dieser Eigenschaft wird auch ein einzelner Datenspeicher zu einem verknüpften Klon-Blueprint hinzugefügt. <i>N</i> ist der Index (beginnend mit 0) des zuzuweisenden Volumes. Geben Sie den Namen des Datenspeichers ein, der dem Volume zugewiesen werden soll. Dies ist der Datenspeichername, der auf der Seite „Computing-Ressource bearbeiten“ im Feld „Speicherpfad“ angezeigt wird. Festplatten müssen sequenziell nummeriert werden.
<code>VirtualMachine.EPI.Type</code>	Gibt den External Provisioning Infrastructure-Typ an. Legen Sie diese Eigenschaft für die BMC BladeLogic-Integration auf „BMC“ fest. Legen Sie diese Eigenschaft für die Citrix Provisioning Server-Integration auf „CitrixProvisioning“ fest.
<code>VirtualMachine.EULA.AcceptAll</code>	Legen Sie diese Eigenschaft auf „True“ fest, um anzugeben, dass alle Lizenzvereinbarungen (EULAs) für die VM-Vorlagen der vCloud Air- oder vCloud Director-Endpoints während der Bereitstellung akzeptiert werden.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Host.TpmEnabled</code>	<p>Beschränkt die Bereitstellung virtueller Maschinen auf Hosts, auf denen ein TPM-Gerät (Trust Protection Module) installiert ist und das von ESX und vSphere erkannt wird. Der Standardwert lautet „False“.</p> <p>Für alle Hosts in einem Cluster muss ein Trust Protection Module-Gerät installiert sein. Wenn keine akzeptablen Hosts oder Cluster gefunden werden, kann die Maschine erst bereitgestellt werden, nachdem diese Eigenschaft entfernt wurde.</p>
<code>VirtualMachine.Memory.Size</code>	<p>Gibt für die Maschine die Größe des Arbeitsspeichers in MB an, wie beispielsweise 1024. Der Standardwert ist der von der Arbeitsspeicher-Einstellung im Blueprint angegebene Wert.</p> <p>Hinweis Diese benutzerdefinierte Eigenschaftseinstellung wird durch die Arbeitsspeicher-Einstellung im Blueprint überschrieben, wenn die Maschine erstmalig bereitgestellt wird.</p>
<code>VirtualMachine.NetworkN.Address</code>	<p>Gibt die IP-Adresse des Netzwerkgeräts <i>N</i> in einer mit einer statischen IP-Adresse bereitgestellten Maschine an.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>
<code>VirtualMachine.NetworkN.AdditionAddressM</code>	<p>Definiert eine zusätzliche IP-Adresse <i>M</i>, die einer OpenStack-Instanz für das Netzwerk <i>N</i> zugeteilt wird, mit Ausnahme des IP-Adresssatzes, der durch die Eigenschaft <code>VirtualMachine.NetworkN</code> angegeben wird. <code>Address</code> angegeben wird. Weitere Adressen werden auf der Registerkarte „Netzwerk“ in der Spalte „Zusätzliche Adressen“ angezeigt.</p> <p>Diese Eigenschaft wird von der Zustandsdatenerfassung der OpenStack-Maschine verwendet. Für diese Eigenschaft werden nur vom OpenStack-Endpoint Daten erfasst, aber sie gilt nicht speziell für OpenStack und kann von anderen Endpoint-Typen für die Lebenszyklus-Erweiterbarkeit verwendet werden.</p> <p>Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VirtualMachine.NetworkN.AddressType	<p>Gibt an, wie die Zuteilung der IP-Adresse an den Netzwerkanbieter übermittelt wird, wobei „NetworkN“ die Netzwerknummer (beginnend bei 0) ist. Die folgenden Werte sind verfügbar:</p> <ul style="list-style-type: none"> ■ DHCP ■ Statisch ■ MANUAL (nur verfügbar für vCloud Air und vCloud Director) <p>Für den Wert MANUAL ist auch die Angabe einer IP-Adresse erforderlich.</p> <p>Mit dieser Eigenschaft können Sie vCloud Air-, vCloud Director- und vSphere-Maschinenkomponenten im Blueprint konfigurieren. Weitere Informationen finden Sie auch unter VirtualMachine.NetworkN.Name. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>
VirtualMachine.NetworkN.MacAddressType	<p>Gibt an, ob die MAC-Adresse des Netzwerkgeräts <i>N</i> generiert wird („generated“) oder benutzerdefiniert („static“) ist. Diese Eigenschaft ist für das Klonen verfügbar.</p> <p>Der Standardwert lautet „generated“. Mit dem Wert „static“ müssen Sie auch VirtualMachine.NetworkN.MacAddress verwenden, um die MAC-Adresse anzugeben.</p> <p>Die benutzerdefinierten Eigenschaften VirtualMachine.NetworkN gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VirtualMachine.NetworkN.MacAddress	<p>Gibt die MAC-Adresse des Netzwerkgeräts <i>N</i> an. Diese Eigenschaft ist für das Klonen verfügbar.</p> <p>Wenn VirtualMachine.NetworkN.MacAddressType den Wert „generated“ hat, enthält diese Eigenschaft die generierte Adresse.</p> <p>Wenn VirtualMachine.NetworkN.MacAddressType den Wert „static“ hat, enthält diese Eigenschaft die MAC-Adresse. Für virtuelle Maschinen, die auf ESX Server-Hosts bereitgestellt werden, muss die Adresse innerhalb des von VMware angegebenen Bereichs liegen. Weitere Informationen finden Sie in der vSphere-Dokumentation.</p> <p>Die benutzerdefinierten Eigenschaften VirtualMachine.Network<i>N</i> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VirtualMachine.NetworkN.Name	<p>Gibt den Namen des Netzwerks an, mit dem eine Verbindung hergestellt werden soll. Beispielsweise das Netzwerkgerät <i>N</i>, mit dem eine Maschine verbunden wird. Dies entspricht einer Netzwerkkarte (Network Interface Card, NIC).</p> <p>Standardmäßig wird ein Netzwerk aus den in der Reservierung verfügbaren Netzwerkpfaden zugewiesen, in denen die Maschine bereitgestellt wird. Siehe auch <code>VirtualMachine.NetworkN.AddressType</code>.</p> <p>Sie können sicherstellen, dass ein Netzwerkgerät mit einem bestimmten Netzwerk verbunden wird, indem Sie für diese Eigenschaft den Namen eines Netzwerks in einer verfügbaren Reservierung festlegen. Wenn Sie beispielsweise als Eigenschaften <i>N</i>= 0 und 1 festlegen, erhalten Sie zwei NICs und deren zugewiesenen Wert, vorausgesetzt das Netzwerk ist in der zugeordneten Reservierung ausgewählt.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p> <p>Ein Beispiel dafür, wie Sie diese benutzerdefinierte Eigenschaft verwenden können, um <code>VirtualMachine.Network0.Name</code> basierend auf der Auswahl des Konsumenten aus einer Liste der vordefinierten verfügbaren Netzwerke dynamisch festzulegen, finden Sie im Blog Adding a Network Selection Drop-Down in vRA 7 (Hinzufügen eines Dropdown-Menüs für die Netzwerkauswahl in vRA 7).</p>
VirtualMachine.NetworkN.PortID	<p>Gibt die für das Netzwerkgerät <i>N</i> zu verwendende Port-ID an, wenn eine dvPort-Gruppe mit einem vSphere Distributed Switch verwendet wird.</p> <p>Die benutzerdefinierten Eigenschaften <code>VirtualMachine.NetworkN</code> gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke wird diese Eigenschaft nicht unterstützt.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VirtualMachine.NetworkN.NetworkProfileName	<p>Gibt den Namen eines Netzwerkprofils an, aus dem dem Netzwerkgerät <i>N</i> eine statische IP-Adresse zugewiesen werden soll oder aus dem der statische IP-Adressbereich bezogen werden soll, der dem Netzwerkgerät <i>N</i> einer geklonten Maschine zugewiesen werden kann. Dabei steht <i>N</i>=0 für das erste Gerät, 1 für das zweite Gerät usw.</p> <p>Das Netzwerkprofil, auf das die Eigenschaft verweist, wird zum Zuweisen einer IP-Adresse verwendet. Die Eigenschaft bestimmt das Netzwerk, an das die Maschine angeschlossen ist, basierend auf der Reservierung.</p> <p>Die Änderung dieses Eigenschaftswerts hat nach der Zuweisung des Netzwerks keine Auswirkung auf die erwarteten IP-Adresswerte für die angegebenen Maschinen.</p> <p>Mit der WIM-basierten Bereitstellung für virtuelle Maschinen können Sie mithilfe dieser Eigenschaft ein Netzwerkprofil und eine Netzwerkschnittstelle angeben. Sie können aber auch den Abschnitt „Netzwerk“ der Seite „Virtuelle Reservierung“ verwenden.</p> <p>Die folgenden Attribute des Netzwerkprofils sind für die Zuweisung von statischen IP-Adressen in einem Klon-Blueprint verfügbar:</p> <ul style="list-style-type: none"> ■ VirtualMachine.NetworkN.SubnetMask ■ VirtualMachine.NetworkN.Gateway ■ VirtualMachine.NetworkN.PrimaryDns ■ VirtualMachine.NetworkN.SecondaryDns ■ VirtualMachine.NetworkN.PrimaryWins ■ VirtualMachine.NetworkN.SecondaryWins ■ VirtualMachine.NetworkN.DnsSuffix ■ VirtualMachine.NetworkN.DnsSearchSuffixes <p>Die benutzerdefinierten Eigenschaften VirtualMachine.NetworkN gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p> <p>Sie können diese benutzerdefinierte Eigenschaft nicht verwenden, um einen Profilnamen für eine bedarfsgesteuerte NAT oder ein bedarfsgesteuertes geroutetes Netzwerk zu definieren. Da Profilnamen für bedarfsgesteuerte Netzwerke bei Zuteilungszeit generiert werden (bei der Bereitstellung), sind deren Namen beim Erstellen oder Bearbeiten des Blueprints unbekannt. Um Informationen von bedarfsgesteuerten Netzwerken von NSX anzugeben, verwenden Sie die entsprechende Netzwerkkomponente in der Blueprint-Design-Arbeitsfläche für Ihre vSphere-Maschinenkomponenten.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<ul style="list-style-type: none"> ■ VirtualMachine.NetworkN.SubnetMask ■ VirtualMachine.NetworkN.Gateway ■ VirtualMachine.NetworkN.PrimaryDns ■ VirtualMachine.NetworkN.SecondaryDns ■ VirtualMachine.NetworkN.PrimaryWins ■ VirtualMachine.NetworkN.SecondaryWins ■ VirtualMachine.NetworkN.DnsSuffix ■ VirtualMachine.NetworkN.DnsSearchSuffixes 	<p>Konfiguriert Attribute des in VirtualMachine.NetworkN.NetworkProfileName angegebenen Netzwerkprofils.</p> <p>Die benutzerdefinierten Eigenschaften VirtualMachine.NetworkN gelten speziell für einzelne Blueprints und Maschinen. Wenn eine Maschine angefordert wird, erfolgt die Zuteilung der Netzwerk- und IP-Adresse vor der Zuweisung der Maschine zu einer Reservierung. Es ist nicht garantiert, dass Blueprints einer bestimmten Reservierung zugeteilt werden, weshalb Sie diese Eigenschaft nicht für eine Reservierung verwenden sollten.</p> <p>Wenn Sie mithilfe von VirtualMachine.NetworkN.DnsSearchSuffixes Werte für mehrere DNS-Suchsuffixe angeben, können Sie die Werte für eine Windows-Bereitstellung durch Kommas voneinander trennen. Für bedarfsgesteuerte NAT-Netzwerke oder bedarfsgesteuerte geroutete Netzwerke werden diese Eigenschaften nicht unterstützt.</p>
VirtualMachine.Rdp.File	<p>Gibt die Datei an, die die RDP-Einstellungen enthält, die beim Öffnen eines RDP-Links zur Maschine verwendet werden sollen. Diese Eigenschaft kann zusammen mit bzw. als Alternative für VirtualMachine.Rdp.SettingN verwendet werden. Die Datei muss sich im Ordner <code>vRA_installation_dir\Server\Website\Rdp</code> befinden. Sie müssen das Verzeichnis Rdp erstellen.</p> <p>Informationen hierzu finden Sie unter VirtualMachine.Rdp.SettingN.</p>
VirtualMachine.Rdp.SettingN	<p>Gibt die RDP-Einstellungen ein, die beim Öffnen eines RDP-Links zu einer Maschine verwendet werden sollen. <i>N</i> ist eine eindeutige Zahl zur Unterscheidung der RDP-Einstellungen. Beispiel: Um die RDP-Authentifizierungsebene so festzulegen, dass keine Authentifizierungsanforderung festgelegt wird, definieren Sie die benutzerdefinierte Eigenschaft VirtualMachine.Rdp.Setting1 und legen den Wert für die Authentifizierungsebene auf „i:3“ fest. Weitere Informationen zu RDP-Einstellungen und ihrer korrekten Syntax finden Sie in der Dokumentation zu Microsoft Windows RDP wie beispielsweise unter RDP Settings for Remote Desktop Services in Windows Server.</p> <p>Informationen hierzu finden Sie unter VirtualMachine.Rdp.File.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Reconfigure.DisableHotCpu</code>	<p>Legen Sie diese Eigenschaft auf „True“ fest, um anzugeben, dass mit der Option für die Maschinenneukonfiguration die angegebene Maschine neu gestartet wird. Standardmäßig wird mit der Option für die Maschinenneukonfiguration die Maschine nicht neu gestartet.</p> <p>Beim Hinzufügen von CPU, Arbeitsspeicher oder Speicher bei laufendem Betrieb schlägt die Option für die Maschinenneukonfiguration fehl und die Maschine wird nicht neu gestartet, außer die Einstellung <code>Hot Add</code> ist in vSphere für die Maschine oder Vorlage aktiviert. Sie können <code>VirtualMachine.Reconfigure.DisableHotCpu=true</code> einer Maschinenkomponente in einem vRealize Automation-Blueprint hinzufügen, um die Einstellung <code>Hot Add</code> zu deaktivieren und den Neustart der Maschine unabhängig von der vSphere-Einstellung <code>Hot Add</code> zu erzwingen. Diese benutzerdefinierte Eigenschaft ist nur für Maschinentypen verfügbar, die die Hardwareneukonfiguration unterstützen, nämlich vSphere, vCloud Air und vCloud Director.</p>
<code>VirtualMachine.Request.Layout</code>	Gibt das Eigenschaftslayout an, das auf der Seite für VM-Anforderungen verwendet werden soll. Dieser Wert muss mit dem Namen des zu verwendenden Layouts übereinstimmen.
<code>VirtualMachine.Software.Execute</code>	<p>Wenn diese Eigenschaft auf „True“ festgelegt ist, werden Gast-Agents für die Bereitstellung von Amazon Web Services unterstützt.</p> <p>Verwenden Sie diese Eigenschaft mit den benutzerdefinierten Eigenschaften <code>VirtualMachine.SoftwareN.Name</code> und <code>VirtualMachine.SoftwareN.ScriptPath</code>, um Gast-Agents in der Amazon Web Services-Bereitstellung zu konfigurieren und zu verwenden.</p>
<code>VirtualMachine.SoftwareN.Name</code>	Gibt den beschreibenden Namen der Softwareanwendung <i>N</i> oder eines Skripts an, die bzw. das während der Bereitstellung installiert oder ausgeführt werden soll. Dies ist eine optionale und rein informative Eigenschaft. Sie hat keine echte Funktion für den erweiterten Klon-Workflow oder den Gast-Agent, ist aber hilfreich für die benutzerdefinierte Softwareauswahl in einer Benutzeroberfläche oder für Berichte zur Softwarenutzung.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VirtualMachine.SoftwareN.ScriptPath	<p data-bbox="751 268 1385 386">Gibt den vollständigen Pfad zum Installationsskript einer Anwendung an. Bei dem Pfad muss es sich um einen gültigen absoluten Pfad wie er im Gastbetriebssystem angezeigt wird handeln und er muss den Namen der Skriptdatei enthalten.</p> <p data-bbox="751 401 1410 680">Sie können benutzerdefinierte Eigenschaftswerte als Parameter an das Skript übergeben, indem Sie <code>{CustomPropertyName}</code> in der Pfadzeichenfolge einfügen. Angenommen, Sie haben eine benutzerdefinierte Eigenschaft <code>ActivationKey</code> mit dem Wert <code>1234</code>. In diesem Fall lautet der Skriptpfad <code>D:\InstallApp.bat -key {ActivationKey}</code>. Der Gast-Agent führt den Befehl <code>D:\InstallApp.bat -key 1234</code> aus. Ihre Skriptdatei kann dann so programmiert werden, dass dieser Wert akzeptiert und verwendet wird.</p> <p data-bbox="751 695 1406 812">Sie können auch benutzerdefinierte Eigenschaftswerte als Parameter an das Skript weitergeben, indem Sie <code>{YourCustomProperty}</code> in die Pfadzeichenfolge einfügen. Wenn Sie beispielsweise den Wert <code>\\vra-scripts.mycompany.com\scripts\changeIP.bat</code> eingeben, wird das Skript <code>changeIP.bat</code> von einem gemeinsam genutzten Speicherort ausgeführt. Wenn Sie jedoch den Wert <code>\\vra-scripts.mycompany.com\scripts\changeIP.bat {VirtualMachine.Network0.Address}</code> eingeben, wird das Skript für die Änderung der IP-Adresse ausgeführt, aber auch der Wert der Eigenschaft <code>VirtualMachine.Network0.Address</code> als Parameter an das Skript weitergegeben.</p> <p data-bbox="751 1083 1406 1136">Fügen Sie <code>{Owner}</code> ein, um den Namen des Maschinenbesitzers an das Skript zu übergeben.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.ScriptPath.Decrypt</code>	<p>Ermöglicht vRealize Automation das Abrufen einer verschlüsselten Zeichenfolge, die als ordnungsgemäß formatierte benutzerdefinierte Eigenschaftsanweisung <code>VirtualMachine.SoftwareN.ScriptPath</code> an die gугent-Befehlszeile übergeben wird.</p> <p>Sie können eine verschlüsselte Zeichenfolge wie beispielsweise Ihr Kennwort als benutzerdefinierte Eigenschaft in einem Befehlszeilenargument bereitstellen. Auf diese Weise können Sie verschlüsselte Informationen speichern, die der Gast-Agent entschlüsseln und als gültiges Befehlszeilenargument interpretieren kann. Beispielsweise ist die benutzerdefinierte Eigenschaftszeichenfolge <code>VirtualMachine.Software0.ScriptPath = c:\dosomething.bat Kennwort</code> nicht sicher, da sie ein tatsächliches Kennwort enthält.</p> <p>Zum Entschlüsseln des Kennworts können Sie eine benutzerdefinierte vRealize Automation-Eigenschaft erstellen, wie beispielsweise <code>MyPassword = password</code>, und die Verschlüsselung durch Aktivieren des verfügbaren Kontrollkästchens aktivieren. Der Gast-Agent entschlüsselt den Eintrag [MyPassword] in den Wert in der benutzerdefinierten Eigenschaft <code>MyPassword</code> und führt das Skript als <code>c:\dosomething.bat password</code> aus.</p> <ul style="list-style-type: none"> ■ Erstellen Sie die benutzerdefinierte Eigenschaft MyPassword = Kennwort, wobei <i>Kennwort</i> der Wert Ihres tatsächlichen Kennworts ist. Aktivieren Sie die Verschlüsselung durch Aktivieren des verfügbaren Kontrollkästchens. ■ Legen Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.ScriptPath.Decrypt</code> als VirtualMachine.ScriptPath.Decrypt = true fest. ■ Legen Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.Software0.ScriptPath</code> als VirtualMachine.Software0.ScriptPath = c:\dosomething.bat [MyPassword] fest. <p>Wenn Sie <code>VirtualMachine.ScriptPath.Decrypt</code> auf „False“ festlegen oder die benutzerdefinierte Eigenschaft <code>VirtualMachine.ScriptPath.Decrypt</code> nicht erstellen, wird die Zeichenfolge in den eckigen Klammern ([und]) nicht entschlüsselt.</p>
<code>VirtualMachine.SoftwareN.ISOName</code>	Gibt den Pfad und den Dateinamen der ISO-Datei in Bezug auf das Stammverzeichnis des Datenspeichers an. Das Format lautet <code>/Ordnername/Unterordnername/Dateiname.iso</code> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.
<code>VirtualMachine.SoftwareN.ISOLocation</code>	Gibt den Speicherpfad an, der die ISO-Imagedatei enthält, die von der Anwendung oder dem Skript verwendet werden soll. Formatieren Sie den in der Hostreservierung angezeigten Pfad, wie beispielsweise <code>netapp-1:it_nfs_1</code> . Wenn kein Wert angegeben wird, wird das ISO-Image nicht gemountet.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Storage.Name</code>	Identifiziert den Speicherpfad, in dem sich die Maschine befindet. Der Standardwert ist der in der Reservierung angegebene Wert, der zum Bereitstellen der Maschine verwendet wurde.
<code>VirtualMachine.Storage.AllocationType</code>	<p>Speichert erfasste Gruppen in einem einzigen Datenspeicher. Eine verteilte Umgebung speichert Festplatten im Round Robin-Verfahren. Geben Sie einen der folgenden Werte an:</p> <ul style="list-style-type: none"> ■ Erfasst <p>Alle Festplatten zusammenhalten.</p> ■ Verteilt <p>Ermöglicht die Platzierung der Festplatten in jedem Datenspeicher oder Datenspeicher-Cluster, der in der Reservierung verfügbar ist.</p> <p>Ein Beispiel dafür, wie die <code>VirtualMachine.Storage.AllocationType</code>-Eigenschaft zum Erstellen von Datenspeicher-Clustern verwendet wird, finden Sie im Blog Keeping Multiple Disks Together (Mehrere Festplatten zusammenhalten).</p>
<code>VirtualMachine.Storage.Cluster.Automation.Enabled</code>	Wenn Sie diese Eigenschaft auf „True“ festlegen, ist die Speicherclusterautomatisierung auf der Maschine aktiviert. Wenn diese Eigenschaft auf „False“ festgelegt ist, ist die Speicherclusterautomatisierung auf der Maschine deaktiviert. Der Typ der Speicherclusterautomatisierung wird durch die benutzerdefinierte Eigenschaft <code>VirtualMachine.Storage.Cluster.Automation.Behavior</code> bestimmt.
<code>VirtualMachine.Storage.Cluster.Automation.Behavior</code>	<p>Gibt den Typ des SDRS-Verhaltens an, wenn <code>VirtualMachine.Storage.Cluster.Automation.Enabled</code> auf „True“ festgelegt ist.</p> <p>Die verfügbaren Werte für den Verhaltenstyp sind „automated“ und „manual“.</p> <p>Die Eigenschaften <code>VirtualMachine.Storage.Cluster.Automation.Enabled</code> und <code>VirtualMachine.Storage.Cluster.Automation.Behavior</code> werden nach der Bereitstellung der Maschine und nach Abschluss der Erfassung von Bestandslistendaten festgelegt. Wenn die Automatisierung deaktiviert wird, ist <code>VirtualMachine.Storage.Cluster.Automation.Behavior</code> nicht auf der Maschine vorhanden.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VirtualMachine.Storage.ReserveMemory</code>	<p>Legen Sie diese Eigenschaft für die Verwaltung der vSwap-Speicherzuteilung auf „True“ fest, um die Verfügbarkeit sicherzustellen und die Zuteilung in der Reservierung festzulegen. Die vSwap-Zuteilung wird beim Erstellen oder Neukonfigurieren einer virtuellen Maschine berücksichtigt. Die vSwap-Zuteilungsprüfung ist nur für vSphere-Endpoints verfügbar.</p> <hr/> <p>Hinweis Wenn Sie die benutzerdefinierte Eigenschaft <code>VirtualMachine.Storage.ReserveMemory</code> beim Erstellen oder Neukonfigurieren der Maschine in vRealize Automation nicht angeben, ist die Verfügbarkeit von Auslagerungsspeicherplatz nicht sichergestellt. Wenn Sie diese Eigenschaft für eine bereits bereitgestellte Maschine hinzufügen und die zugeteilte Reservierung voll ist, überschreitet der in der Reservierung zugeteilte Speicher möglicherweise den tatsächlich zugeteilten Speicher.</p>
<code>VirtualMachine.VDI.Type</code>	<p>Gibt den Virtual Desktop Infrastructure-Typ an. Für die XenDesktop-Bereitstellung legen Sie hierfür „XenDesktop“ fest.</p>
<code>VMware.AttributeN.Name</code>	<p>Gibt den Namen eines Attributs in vRealize Orchestrator an. Beispielsweise wird hiermit der Wert des in der Eigenschaft <code>VMware.AttributeN.Name</code> verwendeten Attributs angegeben. Ersetzen Sie den Buchstaben <i>N</i> durch eine Zahl, die bei 0 beginnt und für jedes festzulegende Attribut erhöht wird.</p>
<code>VMware.AttributeN.Value</code>	<p>Gibt den Wert des in der Eigenschaft <code>VMware.AttributeN.Name</code> verwendeten Attributs an. Ersetzen Sie den Buchstaben <i>N</i> durch eine Zahl, die bei 0 beginnt und für jedes festzulegende Attribut erhöht wird.</p>
<code>VMware.Endpoint.Openstack.IdentityProvider.Domain.Name</code>	<p>Ermöglicht es, dass vRealize Automation die erforderliche Authentifizierung des KeystoneV3-Domänennamens unterstützt. Wenn Keystone V3 aktiviert ist, können Sie die Eigenschaft verwenden, um dem OpenStack-Endpoint eine bestimmte Domäne zur Authentifizierung mit einem Keystone V3 OpenStack-Identitätsanbieter zuzuweisen.</p> <ul style="list-style-type: none"> ■ Fügen Sie für neue Endpoints die benutzerdefinierte Eigenschaft hinzu, um eine bestimmte Domäne zuzuweisen. ■ Fügen Sie für aktualisierte oder migrierte Endpoints die benutzerdefinierte Eigenschaft nur dann hinzu, wenn die Datenerfassung nach einem Upgrade oder einer Migration fehlschlägt.
<code>VMware.Endpoint.Openstack.IdentityProvider.Version</code>	<p>Gibt die Version des OpenStack-Identitätsanbieters (Keystone) an, die beim Authentifizieren eines OpenStack-Endpoints verwendet werden soll. Konfigurieren Sie den Wert 3 für die Authentifizierung mit dem OpenStack-Identitätsanbieter Keystone V3. Wenn Sie einen anderen Wert verwenden oder diese benutzerdefinierte Eigenschaft nicht verwenden, wird für die Authentifizierung standardmäßig Keystone V2 verwendet.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VMware.Endpoint.Openstack.Release	Auslaufend. Gibt beim Erstellen eines OpenStack-Endpoints die OpenStack-Version an, wie beispielsweise „Havana“ oder „Icehouse“. Erforderlich für die Bereitstellung von OpenStack 6.2.1, 6.2.2 und 6.2.3.
VMware.Endpoint.NSX.HideDiscoveredSecurityObjects	<p>Legen Sie die Option auf „true“ fest, um neu erkannte Sicherheitsobjekte im aktiven Mandanten für die NSX-Endpoints zu verbergen, mit denen die Sicherheitsobjekte verbunden sind. Anderenfalls stehen alle neuen Sicherheitsobjekte nach der Datenerfassung in allen Mandanten zur Verfügung, vorausgesetzt, dass das Sicherheitsobjekt für einen Endpoint gilt, in dem Sie über eine Reservierung verfügen. Mit dieser Option können Sie vermeiden, dass Benutzer auf Sicherheitsobjekte zugreifen, wenn Sie diese Objekte einem einzelnen Mandanten zuweisen oder für alle Mandanten maskieren möchten. Legen Sie die Option auf „false“ fest, um zu „Global“ zurück zu wechseln, wodurch alle neuen Sicherheitsobjekte nach der Datenerfassung in allen Mandanten zur Verfügung stehen, vorausgesetzt, dass das Sicherheitsobjekt für einen Endpoint gilt, in dem Sie eine Reservierung besitzen.</p> <p>Damit dies wirksam wird, fügt der Fabric-Administrator die benutzerdefinierte Eigenschaft <code>VMware.Endpoint.NSX.HideDiscoveredSecurityObjects</code> dem zugehörigen NSX-Endpoint hinzu, der mit einem vSphere-Endpoint verknüpft ist. Die Einstellung gilt für die nächste Erfassung von Bestandslistendaten. Vorhandene Sicherheitsobjekte bleiben unverändert.</p> <p>Zum Ändern der Mandanteneinstellung eines Sicherheitsobjekts, für das bereits Daten erfasst wurden, wie z. B. vorhandene Sicherheitsobjekte nach dem Upgrade auf die aktuelle Version von vRealize Automation, können Sie die Einstellung „Mandanten-ID“ des Sicherheitsobjekts programmgesteuert mithilfe der REST API von vRealize Automation oder mithilfe von vRealize CloudClient bearbeiten. Die verfügbaren Einstellungen unter „Mandanten-ID“ für den NSX-Endpoint lauten wie folgt:</p> <ul style="list-style-type: none"> ■ "<code><global></code>" – das Sicherheitsobjekt steht allen Mandanten zur Verfügung. Dies ist die Standardeinstellung für vorhandene Sicherheitsobjekte nach dem Upgrade auf diese Version und für alle neuen Sicherheitsobjekte, die Sie erstellen. ■ "<code><unscoped></code>" – das Sicherheitsobjekt steht keinem Mandanten zur Verfügung. Nur der Systemadministrator kann auf das Sicherheitsobjekt zugreifen. Diese Einstellung eignet sich für die Definition von Sicherheitsobjekten, die irgendwann einem bestimmten Mandanten zugewiesen werden sollen. ■ "<code>Tenant_id_name</code>" – das Sicherheitsobjekt steht nur einem einzelnen benannten Mandanten zur Verfügung.

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VMware.Hardware.Version	Gibt die VM-Hardwareversion an, die für vSphere-Einstellungen verwendet werden soll. Unterstützte Werte sind derzeit „vmx-04“, „vmx-07“, „vmx-08“, „vmx-09“ und „vmx-10“. Diese Eigenschaft betrifft VM-Erstellungs- und VM-Aktualisierungsworkflows und ist nur für einfache Workflow-Blueprints verfügbar.
VMware.VirtualCenter.OperatingSystem	<p>Gibt die Version des vCenter Server-Gastbetriebssystems (VirtualMachineGuestOsIdentifier) an, mit der vCenter Server die Maschine erstellt. Diese Betriebssystemversion muss mit der Betriebssystemversion übereinstimmen, die auf der bereitgestellten Maschine installiert werden soll. Administratoren können Eigenschaftsgruppen mithilfe mehrerer Eigenschaftensätze erstellen, wie beispielsweise VMware[OS_Version]Properties. Diese Eigenschaftensätze sind vordefiniert und enthalten die korrekten Werte für VMware.VirtualCenter.OperatingSystem. Diese Eigenschaft dient für die virtuelle Bereitstellung.</p> <p>Wenn diese Eigenschaft einen Nicht-Windows-Wert aufweist, ist die Benutzeroberflächenoption Verbindungsherstellung mithilfe von RDP deaktiviert. Diese Eigenschaft kann in einem virtuellen, physischen oder Cloud-Blueprint verwendet werden.</p> <p>Weitere Informationen hierzu finden Sie im Abschnitt zum Enumerationstyp VirtualMachineGuestOsIdentifier in der vSphere API/SDK-Dokumentation. Eine Liste der aktuell akzeptierten Werte finden Sie in der vCenter Server-Dokumentation.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VMware.SCSI.Type	<p>Für vCloud Air-, vCloud Director - oder vSphere-Maschinenkomponenten in Blueprints wird hiermit der SCSI-Maschinentyp mithilfe eines der folgenden Werte, bei denen die Groß-/Kleinschreibung beachtet wird, angegeben:</p> <ul style="list-style-type: none"> ■ buslogic Verwenden Sie die BusLogic-Emulation für die virtuelle Festplatte. ■ lsilogic Verwenden Sie die LSILogic-Emulation für die virtuelle Festplatte (Standardwert). ■ lsilogicsas Verwenden Sie die LSILogic SAS 1068-Emulation für die virtuelle Festplatte. ■ pvscsi Verwenden Sie die Para-Virtualisierungs-Emulation für die virtuelle Festplatte. ■ Keine Verwenden Sie diese Option, wenn kein SCSI-Controller für diese Maschine vorhanden ist. <p>Die Eigenschaft <code>VMware.SCSI.Type</code> ist für die Verwendung mit dem CloneWorkflow-Bereitstellungsworkflow nicht verfügbar. Wenn Sie den CloneWorkflow-Bereitstellungsworkflow beim Konfigurieren der Maschinenkomponente auf der Design-Arbeitsfläche des Blueprints angeben, können Sie die Eigenschaft <code>VMware.SCSI.Type</code> nicht verwenden.</p>
VMware.SCSI.Sharing	<p>Gibt an, in welchem Modus der VMware-SCSI-Bus der Maschine gemeinsam verwendet wird. Mögliche Werte basieren auf dem ENUM-Wert <code>VirtualSCSISharing</code> und enthalten „noSharing“, „physicalSharing“ und „virtualSharing“.</p> <p>Wenn Sie den CloneWorkflow-Bereitstellungsworkflow beim Konfigurieren der Maschinenkomponente auf der Design-Arbeitsfläche des Blueprints angeben, ist die Eigenschaft <code>VMware.SCSI.Sharing</code> nicht verfügbar.</p> <p>Die Eigenschaft <code>VMware.SCSI.Sharing</code> ist für die Verwendung mit dem CloneWorkflow-Bereitstellungsworkflow nicht verfügbar. Wenn Sie den CloneWorkflow-Bereitstellungsworkflow beim Konfigurieren der Maschinenkomponente auf der Design-Arbeitsfläche des Blueprints angeben, können Sie die Eigenschaft <code>VMware.SCSI.Sharing</code> nicht verwenden.</p>
VMware.Memory.Reservation	<p>Definiert die Menge an reserviertem Arbeitsspeicher für die virtuelle Maschine in MB, beispielsweise 1024. Durch Festlegen dieses Werts wird auch die Größe der VM-Auslagerungsdatei auf der Festplatte um die angegebene Menge verringert.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VMware.Network.Type</code>	<p>Gibt das Netzwerk an, mit dem die VM gemäß der Angabe in der Reservierung verbunden werden soll. Der Netzwerkadapter der Maschine muss mit einem eindeutigen Netzwerk verbunden werden.</p> <p>Die folgenden Adaptertypwerte sind verfügbar:</p> <ul style="list-style-type: none"> ■ Flexible (Standard) ■ VirtualPCNet32 (nicht kompatibel mit vSphere). ■ E1000 oder VirtualE1000 ■ VMXNET oder VirtualVMXNET ■ VMXNET2 ■ VMXNET3 <p>Legen Sie diese Eigenschaft bei der Bereitstellung von virtuellen Windows-32-Bit-Maschinen auf ESX Server-Hosts auf „E1000“ fest, um sicherzustellen, dass die Maschinen mit dem richtigen Netzwerkadapter erstellt werden. Diese Eigenschaft wird für die physische Bereitstellung nicht verwendet.</p>
<code>VMware.Ovf.Thumbprint</code>	<p>Wenn sich die OVF-Datei auf einem HTTPS-Server befindet, der über ein Zertifikat verfügt, speichert diese Eigenschaft den Fingerabdruckwert dieses Zertifikats und wird zum Überprüfen des Zertifikats verwendet. Dies spielt keine Rolle, wenn die OVF-Datei auf einem HTTP-Server gehostet wird. Die Eigenschaft wird automatisch erstellt, wenn Sie eine OVF-Datei mit dem Bereitstellungsworkflow <code>ImportOvfWorkflow</code> in der Benutzeroberfläche der Blueprint-Komponente importieren. Wenn Sie den Blueprint programmatisch mit vRealize Automation-REST APIs oder vRealize CloudClient erstellen, müssen Sie die Eigenschaft manuell erstellen.</p> <p>Hinweis Der Fingerabdruck kann in einem kommagetrennten Format zur Unterstützung einer Zertifikatskette gespeichert werden.</p> <p>Wenn <code>VMware.Ovf.TrustAllCertificates</code> vorhanden ist, wird die Eigenschaft <code>VMware.Ovf.Thumbprint</code> ignoriert.</p>
<code>VMware.Ovf.TrustAllCertificates</code>	<p>Wenn diese Eigenschaft vorhanden und auf „true“ festgelegt ist, wird die Eigenschaft <code>VMware.Ovf.Thumbprint</code> ignoriert, und beim Importieren einer OVF unter Verwendung des Bereitstellungsworkflows <code>ImportOvfWorkflow</code> wird keine Zertifikatsvalidierung durchgeführt.</p>
<code>VMware.Ovf.Configuration.X</code>	<p>Eine OVF-Datei kann benutzerkonfigurierbare Eigenschaften enthalten, zum Beispiel eine Eigenschaft, die das Root-Kennwort einer von der OVF-Datei bereitgestellten VM festlegt. Wenn Sie eine OVF-Datei in einen Blueprint importieren, werden alle benutzerkonfigurierbaren Eigenschaften, die in der OVF-Datei definiert sind, analysiert und in benutzerdefinierte Eigenschaften der Form <code>VMware.Ovf.Configuration.X</code> konvertiert, wobei X der Name der benutzerkonfigurierbaren Eigenschaft aus der OVF-Datei ist.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VMware.VCenterOrchestrator.EndpointName</code>	Diese Eigenschaft überschreibt eine angegebene Endpoint-Einstellung oder gibt an, dass während der IaaS-Bereitstellung von vRealize Automation ein bestimmter Endpoint verwendet wird. Für diese Eigenschaft kann ein entsprechender vRealize Orchestrator-Endpoint festgelegt werden (z. B. eine externe VRO-Komponente), der in der Umgebung verfügbar ist.
<code>VMware.VirtualCenter.Folder</code>	Gibt den Namen des Bestandslistendatenordners im Datacenter an, in dem die virtuelle Maschine platziert werden soll. Der Standardwert ist VRM. Dieser Ordner ist auch der vSphere-Ordner, in dem vRealize Automation bereitgestellte Maschinen platziert, falls die Eigenschaft nicht verwendet wird. Bei diesem Wert kann es sich um einen Pfad mit mehreren Ordnern handeln, wie beispielsweise <code>production\email_servers</code> . Ein Proxy-Agent erstellt den angegebenen Ordner in vSphere, falls der Ordner nicht vorhanden ist. Bei Ordernamen wird die Groß-/Kleinschreibung beachtet. Diese Eigenschaft ist für die virtuelle Bereitstellung verfügbar.
<code>VDI.Server.Website</code>	<p>Gibt den Servernamen der Citrix Web Interface-Site an, die bei der Verbindungsherstellung mit der Maschine verwendet werden soll. Wenn der Wert von <code>VDI.Server.Name</code> eine XenDesktop-Farm ist, muss diese Eigenschaft einen entsprechenden Wert aufweisen, da andernfalls der Maschinenbesitzer mithilfe von XenDesktop keine Verbindung mit der Maschine herstellen kann. Wenn diese Eigenschaft nicht angegeben ist, bestimmt die Eigenschaft <code>VDI.Server.Name</code> den Desktop Delivery Controller, mit dem eine Verbindung hergestellt werden soll. Dies muss der Name eines Servers sein, der einen Desktop Delivery Controller hostet.</p> <p>Hinweis Falls die Citrix Web Interface (WI) durch StoreFront (SF) ersetzt wurde, können Sie diese Eigenschaft anstelle von <code>VDI.Server.Name</code> für die Verbindungsherstellung mit dem XenDesktop-Server verwenden. Ein Beispielwert ist <code>VDI.Server.Website=sqa-xddc-7.sqa.local/Citrix/StoreWeb</code>. Weitere Informationen hierzu finden Sie unter <code>VDI.Server.Name</code>.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
VDI.Server.Name	<p>Gibt den Namen des Servers an, der den Desktop Delivery Controller hostet, mit dem die Registrierung erfolgen soll, oder aber den Namen einer XenDesktop-Farm, die Desktop Delivery Controller enthält, mit denen die Registrierung erfolgen soll.</p> <p>Wenn es sich bei dem Wert um den Namen einer XenDesktop-Farm handelt, muss der Eigenschaftswert VDI.Server.Website die URL einer entsprechenden Citrix Web Interface-Site sein, die bei der Verbindungsherstellung mit der Maschine verwendet werden soll.</p> <p>Wenn der Wert ein Servername ist und mindestens ein allgemeiner XenDesktop VDI-Agent installiert wurde, ohne einen Desktop Delivery Controller-Server anzugeben, wird die Anforderung mit diesem Wert an den gewünschten Server verwiesen. Wenn der Wert ein Servername ist und nur dedizierte XenDesktop VDI-Agents für bestimmte DDC-Server installiert wurden, muss dieser Wert genau mit dem für einen dedizierten Agent konfigurierten Servernamen übereinstimmen.</p> <hr/> <p>Hinweis Weitere Informationen, wie Sie StoreFront als Standardseite in IIS festlegen, finden Sie in der Citrix-Dokumentation. Weitere Informationen finden Sie auch unter VDI.Server.Website.</p> <hr/> <p>Hinweis Änderungen beim Citrix Web Interface-Protokoll haben Auswirkungen darauf, wie der Standardwert VDI.Server.Name erkannt wird. Der Wert der Eigenschaft VDI.Server.Name wird als Standardverbindungszeichenfolge zum Öffnen der Citrix Web Interface verwendet, wenn Benutzer eine Verbindung mit einem virtuellen Desktop herstellen. Dies ist stets der DNS/IP-Wert des XD-Servers. Wenn mit diesem Wert keine Verbindung mit der Citrix-Schnittstelle hergestellt wird, können Sie nicht auf Ihre VMs zugreifen. Sie können jedoch die benutzerdefinierte Eigenschaft VDI.Server.Website verwenden, wenn die Citrix Web Interface auf einem anderen Server als dem XenDesktop-Server gehostet wird. Wenn diese Eigenschaft in der VM vorhanden ist, wird sie anstelle von VDI.Server.Name verwendet.</p> <hr/>
VDI.Server.Group	<p>Für XenDesktop 5 werden mit dieser Eigenschaft der Name der XenDesktop-Gruppe, der Maschinen hinzugefügt werden sollen, sowie der Name des Katalogs, zu der die Gruppe gehört, im Format <i>Gruppenname;Katalogname</i> angegeben.</p> <p>Für XenDesktop 4 wird hiermit der Name der XenDesktop-Gruppe angegeben, der Maschinen hinzugefügt werden sollen. Vorab zugewiesene XenDesktop 4-Gruppen werden unterstützt.</p>
VDI.ActiveDirectory.Interval	<p>Gibt einen optionalen Intervallwert im Zeitspannenformat für die Active Directory-Registrierungsprüfung von Virtual Desktop Infrastructure-Maschinen an. Der Standardwert ist 00:00:15 (15 Sekunden).</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
<code>VDI.ActiveDirectory.Timeout</code>	Gibt einen optionalen Zeitüberschreitungswert an, der angibt, wie lange gewartet werden soll, bevor die Active Directory-Registrierung erneut versucht wird. Der Standardwert ist 00:00:15 (30 Minuten).
<code>VDI.ActiveDirectory.Delay</code>	Gibt einen optionalen Verzögerungswert im Zeitspannenformat zwischen dem erfolgreichen Hinzufügen einer Maschine zu Active Directory und dem Start der XenDesktop-Registrierung an. Der Standardwert ist 00:00:05 (5 Sekunden).
<code>Vrm.DataCenter.Location</code>	<p>Ermöglicht Ihnen die Nutzung eines Blueprints zum Bereitstellen von Maschinen auf mehr als einer Computing-Ressource. Sie können die Eigenschaft <code>Vrm.DataCenter.Location</code> zu einem Blueprint hinzufügen oder die Option Speicherort auf Anforderung anzeigen im Blueprint aktivieren, damit der Benutzer einen Datencenter-Standort angeben muss, wenn er die Maschinenbereitstellung anfordert.</p> <p>Hinweis Wenn Sie die Option Speicherort auf Anforderung anzeigen im Blueprint aktivieren, müssen Sie die benutzerdefinierte Eigenschaft nicht zusätzlich hinzufügen.</p> <p>Datencenter-Standorte sind in der Datei <code>DataCenterLocations.xml</code> konfiguriert. Die dort gespeicherten Standortwerte werden auf die Computing-Ressourcen angewendet.</p> <p>Verwandte Informationen über das Hinzufügen von Datencenter-Standorten finden Sie unter <i>Konfigurieren von vRealize Automation</i>.</p> <p>Da die Eigenschaft <code>Vrm.DataCenter.Location</code> nicht auf den Inhalt der Datei <code>DataCenterLocations.xml</code> zugreifen kann, sind Sie davon abhängig, dass die Benutzer Eigenschaftswerte bereitstellen, die mit den in der Datei <code>DataCenterLocations.xml</code> angegebenen Standorten übereinstimmen.</p> <p>Verwenden Sie diese Eigenschaft, wenn Sie den Wert des Datencenter-Standorts als Input für eine externe Aktion für eine andere benutzerdefinierte Eigenschaft verwenden möchten.</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
Vrm.DataCenter.Policy	<p>Gibt an, ob für die Bereitstellung eine Computing-Ressource im Zusammenhang mit einem bestimmten Standort verwendet werden muss oder ob jeder beliebige Standort zulässig ist. Um diese Funktion zu aktivieren, müssen Sie das Datencenter zu einer Speicherortdatei hinzufügen. Ordnen Sie jeder Computing-Ressource einen Speicherort zu.</p> <p>Legen Sie „Exact“ (Standardwert) fest, um eine angeforderte Maschine auf einer Computing-Ressource bereitzustellen, der der im Blueprint angegebene Speicherort zugeordnet ist. Die Anforderung schlägt fehl, wenn keine Reservierung mit dem angeforderten Speicherort übereinstimmt. Wenn die Eigenschaft nicht vorhanden ist, wird der Standardwert „Exact“ verwendet.</p> <p>Legen Sie „NonExact“ fest, um eine angeforderte Maschine auf einer Computing-Ressource mit ausreichender Kapazität und dem im Blueprint angegebenen Speicherort bereitzustellen. Wenn diese Computing-Ressource nicht verfügbar ist, verwenden Sie die nächste verfügbare Computing-Ressource mit ausreichender Kapazität ohne Berücksichtigung des Speicherorts.</p>
Vrm.ProxyAgent.Uri	<p>Ermöglicht Ihnen, den Vrm.ProxyAgent.Uri-Standardwert außer Kraft zu setzen, der von der VMPS-Endpoint-Adresse in der Konfigurationsdatei für den vRealize Automation-Manager-Dienst abgeleitet wird. Die Einstellung ist häufig auf die lokale Maschine festgelegt, aber möglicherweise möchten Sie sie auf die virtuelle IP (VIP)-Adresse festlegen.</p> <p>Sie können die benutzerdefinierte Eigenschaft Vrm.ProxyAgent.Uri in einem Blueprint angeben. Es folgt ein Beispiel für die Syntax:</p> <p>Vrm.ProxyAgent.Uri=https://loadbalancer-vip/VMPS2Proxy</p>

Tabelle 1-45. Tabelle mit benutzerdefinierten Eigenschaften – V (Fortsetzung)

Eigenschaft	Beschreibung
Vrm.Software.IdNNNN Diese Zeile bezieht sich speziell auf BMC BladeLogic.	Gibt einen Softwareauftrag oder eine Softwarerichtlinie an, die auf alle Maschinen, die über den Blueprint bereitgestellt werden, angewendet werden soll. Legen Sie diesen Wert auf <code>job_type=job_path</code> fest, wobei <code>job_type</code> für den BMC BladeLogic-Auftragstyp und <code>job_path</code> für den Speicherort des Auftrags in BMC BladeLogic steht, wie beispielsweise <code>4=/Utility/putty</code> . <code>NNNN</code> ist eine Zahl zwischen 1000 und 1999. Die erste Eigenschaft muss mit 1000 beginnen und für jede zusätzliche Eigenschaft muss eine Erhöhung in numerischer Reihenfolge durchgeführt werden. <div> 1 – AuditJob 2 – BatchJob 3 – ComplianceJob 4 – DeployJob 5 – FileDeployJob 6 – NSHScriptJob 7 – PatchAnalysisJob 8 – SnapshotJob </div>
Vrm.Software.IdNNNN Diese Zeile bezieht sich speziell auf HP Server Automation.	(Optional) Gibt eine HP Server Automation-Richtlinie an, die auf alle Maschinen, die über den Blueprint bereitgestellt werden, angewendet werden soll. <code>NNNN</code> ist eine Zahl zwischen 1000 und 1999. Die erste Eigenschaft muss mit 1000 beginnen und für jede zusätzliche Eigenschaft muss eine Erhöhung in numerischer Reihenfolge durchgeführt werden.

Benutzerdefinierte Eigenschaften X

Eine Liste der benutzerdefinierten Eigenschaften von vRealize Automation, die mit dem Buchstaben X beginnen.

Tabelle 1-46. Tabelle mit benutzerdefinierten Eigenschaften – X

Eigenschaft	Beschreibung
Xen.Platform.Viridian	Für die virtuelle Bereitstellung legen Sie diese Eigenschaft auf „False“ fest, wenn Sie virtuelle Windows-Maschinen auf einem XenServer-Host oder -Pool bereitstellen. Der Standardwert ist „True“. Diese Eigenschaft wird für die physische Bereitstellung nicht verwendet.

Verwenden des Eigenschaftswörterbuchs

Sie können das Eigenschaftswörterbuch für das Festlegen neuer benutzerdefinierter Eigenschaftsdefinitionen und Eigenschaftsgruppen verwenden.

Eine Eigenschaft wird definiert, um einen bestimmten Datentyp und ein bestimmtes Anzeigeformat für diesen Datentyp zu unterstützen. Sie können auch wiederverwendbare Eigenschaftsgruppen erstellen, um das Hinzufügen mehrerer Eigenschaften zu vereinfachen.

Verwenden von Eigenschaftsdefinitionen

Viele benutzerdefinierte Eigenschaften sind in vRealize Automation enthalten. Sie können auch neue Eigenschaftsdefinitionen definieren, um eindeutige benutzerdefinierte Eigenschaften zu erstellen und um die Bereitstellung von Maschinen besser steuern zu können.

Beim Hinzufügen einer Eigenschaft zu einem Blueprint oder einer Reservierung können Sie festlegen, ob ein Benutzer zur Eingabe eines Eigenschaftswerts aufgefordert und ob der Eigenschaftswert verschlüsselt werden muss.

Sie können festlegen, wie eine Eigenschaft wiedergegeben wird, ob sie z. B. in Form eines Kontrollkästchens oder Dropdown-Menüs mit Werten aus einem benutzerdefinierten vRealize Orchestrator-Workflow angezeigt werden soll.

Sie können Eigenschaften auch zur Steuerung der Funktionen Ihrer benutzerdefinierten Workflows verwenden. Weitere Informationen zur Verwendung von vRealize Automation Designer zum Definieren von und Arbeiten mit benutzerdefinierten Workflows finden Sie unter *Lebenszyklus-Erweiterbarkeit*.

Best Practices für das Benennen von Eigenschaftsdefinitionen

Um Namenskonflikte mit bereitgestellten benutzerdefinierten vRealize Automation-Eigenschaften zu vermeiden, sollten Sie für alle Eigenschaftsnamen, die Sie erstellen, ein standardmäßiges und aussagekräftiges Präfix verwenden. Verwenden Sie für alle neuen Eigenschaftsnamen ein Präfix wie beispielsweise einen Unternehmens- oder Funktionsnamen, gefolgt von einem Punkt. VMware reserviert alle Eigenschaftsnamen, die keinen Punkt (.) enthalten. Eigenschaftsnamen, die sich nicht an diese Empfehlung halten, verursachen möglicherweise einen Konflikt mit benutzerdefinierten vRealize Automation-Eigenschaften. In diesem Fall hat die benutzerdefinierte vRealize Automation-Eigenschaft Vorrang vor von Ihnen erstellten Eigenschaftsdefinitionen.

Allgemeine Verfahren

In den folgenden Schritten wird der allgemeine Vorgang für das Erstellen und Verwenden neuer Eigenschaftsdefinitionen beschrieben:

- 1 Erstellen Sie eine neue Eigenschaftsdefinition und ordnen Sie sie einem Datentyp zu, der einen bestimmten Inhaltstyp ermöglicht, wie z. B. boolesche oder ganzzahlige Inhalte. Verwenden Sie für den neuen Eigenschaftsnamen eine standardmäßige Namenskonvention, wie beispielsweise *my_grouping_prefix.my_property_name*.
- 2 Ordnen Sie einer Eigenschaftsdefinition einen Anzeigetyp zu, wie z. B. ein Kontrollkästchen oder Dropdown-Menü. Verfügbare Anzeigetypen werden dem ausgewählten Datentyp entnommen.
- 3 Fügen Sie die Eigenschaft einem Blueprint entweder individuell oder als Teil einer Eigenschaftsgruppe hinzu.

Fügen Sie die Eigenschaft einem Blueprint hinzu und legen Sie fest, ob der Wert verschlüsselt werden muss.

Fügen Sie die Eigenschaft einem Blueprint hinzu und legen Sie fest, ob der Benutzer zur Eingabe eines Eigenschaftswerts aufgefordert werden soll.

- 4 Geben Sie als Maschinenanforderer erforderliche Werte an, wenn Sie dazu aufgefordert werden.

Verwenden von vRealize Orchestrator-Skriptaktionen

Mit vRealize Orchestrator-Skriptaktionen können Sie den Eigenschaftswert in einem Dropdown-Menü auffüllen. Wenn Sie vRealize Orchestrator-Skriptaktionen verwenden, können Sie ebenfalls einen Dropdown-Menüwert basierend auf den für eine andere Eigenschaft angegebenen Werten auffüllen.

Mit dem Befehl `vra content list --type property-definition` von vRealize CloudClient können Sie alle Eigenschaftsdefinitionen im aktuellen vRealize Automation-Mandanten auflisten. Mit dem Befehl `vra content list --type property-group` von vRealize CloudClient können Sie auch alle Eigenschaftsgruppen auflisten.

Ein Lernprogramm zum Erstellen von dynamischen Eigenschaftsdefinitionen zum Filtern der für Benutzer verfügbaren Optionen finden Sie im Blog [How to use dynamic property definitions](#) (Verwenden von dynamischen Eigenschaftsdefinitionen).

Einschränkungen

Wenn Sie eine Eigenschaftsdefinition erstellen, bei der `Data typeString` entspricht, `Display asDropdown` entspricht und Sie eine vRealize Orchestrator-Aktion verwenden, die Eigenschaften zum Auffüllen der Dropdown-Liste zurückgibt, weist die Liste eine zufällige Reihenfolge auf. Sie können die Reihenfolge nicht bestimmen.

Erstellen und Testen von benutzerdefinierten Eigenschaftsdefinitionen

Sie erstellen eine benutzerdefinierte Eigenschaftsdefinition, die bestimmt, wie die benutzerdefinierte Eigenschaft in vRealize Automation angezeigt wird. Sie können die benutzerdefinierte Eigenschaft einem Blueprint hinzufügen, sodass Sie überprüfen können, ob die Eigenschaft das Kontrollkästchen, das Dropdown-Menü oder andere Steuerelementtypen wie erwartet anzeigt.

Zum Erstellen und Testen der benutzerdefinierten Eigenschaftsdefinitionen benötigen Sie einen Blueprint, der von Ihnen oder einem Test-Benutzerkonto, auf das Sie Zugriff haben, genutzt werden darf. Dieser Test-Blueprint bietet Ihnen die Möglichkeit, die benutzerdefinierte Eigenschaft zu erstellen, sie zu einem Blueprint hinzuzufügen und dann zu überprüfen, ob die benutzerdefinierte Eigenschaft erwartungsgemäß dargestellt wird. Nachdem Sie die benutzerdefinierte Eigenschaft validiert haben, können Sie sie nach Bedarf zu Ihren Produktions-Blueprints hinzufügen.

Voraussetzungen

- Stellen Sie sicher, dass Ihnen ein Blueprint zur Verfügung steht, zu dem Sie die Aktion hinzufügen. Siehe *Konfigurieren von vRealize Automation*.

- Überprüfen Sie, ob Sie für die Nutzung des Blueprints berechtigt sind, sodass Sie die benutzerdefinierten Eigenschaften im Blueprint testen können. Siehe *Konfigurieren von vRealize Automation*.
- Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

1 Erstellen benutzerdefinierter Eigenschaftsdefinitionen

Sie erstellen benutzerdefinierte Eigenschaftsdefinitionen, die bestimmen, wie die benutzerdefinierte Eigenschaft in vRealize Automation angezeigt wird. Die benutzerdefinierte Eigenschaft können Sie in einem Test-Blueprint validieren, bevor Sie sie zu Ihren Produktions-Blueprints hinzufügen.

2 Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint

Sie können vielen Teilen von vRealize Automation benutzerdefinierte Eigenschaften hinzufügen, einschließlich Richtlinien, Business-Gruppen, Endpoints und Reservierungsrichtlinien. Jedoch werden von Ihnen als Eigenschaftsdefinitionen konfigurierte Anzeigeoptionen nur von den Maschinen-Blueprints unterstützt. Das Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint ist eine einfache Möglichkeit zur Sicherstellung, dass die benutzerdefinierte Eigenschaft auf der Benutzeroberfläche so angezeigt wird, wie Sie dies in der Eigenschaftsdefinition vorgesehen haben.

3 Überprüfen der benutzerdefinierten Eigenschaft im Kataloganforderungsformular

Als Ersteller der benutzerdefinierten Eigenschaftsdefinitionen, die vRealize Orchestrator-Aktionen ausführen, sollten Sie Ihre benutzerdefinierten Eigenschaften testen, um sicherzustellen, dass im Anforderungsformular die korrekten Werte angezeigt werden.

Erstellen benutzerdefinierter Eigenschaftsdefinitionen

Sie erstellen benutzerdefinierte Eigenschaftsdefinitionen, die bestimmen, wie die benutzerdefinierte Eigenschaft in vRealize Automation angezeigt wird. Die benutzerdefinierte Eigenschaft können Sie in einem Test-Blueprint validieren, bevor Sie sie zu Ihren Produktions-Blueprints hinzufügen.

■ Erstellen einer Eigenschaftsdefinition

Sie können Eigenschaftsdefinitionen erstellen, um zusätzliche Ebenen der Anpassung von vRealize Automation zu ermöglichen. Wenn Sie eine Eigenschaftsdefinition erstellen, geben Sie einen Datentyp für die Eigenschaft an, wie beispielsweise eine Zeichenfolge oder einen booleschen Typ.

■ Erstellen einer benutzerdefinierten Eigenschaft, die anhand eines regulären Ausdrucks validiert wird

Sie erstellen eine benutzerdefinierte Eigenschaftsdefinition, die einen regulären Ausdruck auswertet, wenn Sie möchten, dass Nutzer des Servicekatalogs validierte Daten im Kataloganforderungsformular eingeben.

- **Erstellen einer benutzerdefinierten Eigenschaftsdefinition, die eine vRealize Orchestrator-Aktion enthält**

Sie erstellen eine benutzerdefinierte Eigenschaftsdefinition, die eine vRealize Orchestrator-Aktion enthält, damit Sie die benutzerdefinierte Eigenschaft einem Blueprint hinzufügen können. Die Aktion wird ausgeführt, wenn der Benutzer des Servicekatalogs die benutzerdefinierte Eigenschaft im Anforderungsformular konfiguriert. Die Aktion ruft die Daten ab, die im Formular angezeigt werden.

- **Bindung von benutzerdefinierten Eigenschaften zum Erstellen einer hierarchischen Beziehung**

Zum Erstellen einer hierarchischen Beziehung zwischen benutzerdefinierten Eigenschaften binden Sie die übergeordnete Eigenschaft an die untergeordnete Eigenschaft. Wenn Sie die übergeordneten und die untergeordneten benutzerdefinierten Eigenschaften einem Blueprint hinzufügen, wählt der anfordernde Benutzer einen Wert für die übergeordnete Eigenschaft aus. Der Wert der übergeordneten Eigenschaft bestimmt die möglichen Werte für die untergeordnete Eigenschaft.

Erstellen einer Eigenschaftsdefinition

Sie können Eigenschaftsdefinitionen erstellen, um zusätzliche Ebenen der Anpassung von vRealize Automation zu ermöglichen. Wenn Sie eine Eigenschaftsdefinition erstellen, geben Sie einen Datentyp für die Eigenschaft an, wie beispielsweise eine Zeichenfolge oder einen booleschen Typ.

Um Konflikte mit bereitgestellten benutzerdefinierten vRealize Automation-Eigenschaften zu vermeiden, sollten Sie das Namensformat *my_prefix.my_property_name1* verwenden. Verwenden Sie beispielsweise ein Standard-Präfix wie etwa einen Unternehmensnamen, gefolgt von einem Punkt (.) und einem beschreibenden Namen.

Von Ihnen erstellte Eigenschaften, die sich nicht an diese Empfehlung halten, verursachen möglicherweise einen Konflikt mit von vRealize Automation bereitgestellten benutzerdefinierten Eigenschaften. In diesem Fall haben die benutzerdefinierten vRealize Automation-Eigenschaften Vorrang vor von Ihnen erstellten Eigenschaften.

Voraussetzungen

Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsdefinitionen** aus.
- 2 Klicken Sie auf **Neu** (+).
- 3 Geben Sie im Textfeld **Name** den neuen Namen der Eigenschaftsdefinition ein.

Verwenden Sie für den neuen Eigenschaftsnamen eine standardmäßige Namenskonvention, wie beispielsweise *my_grouping_prefix.my_property_name*.

Der Wert **Name** wird intern als Eigenschaften-ID gespeichert.

4 Akzeptieren Sie den generierten Wert im Textfeld **Bezeichnung.**

Der Wert **Bezeichnung** wird automatisch mit dem von Ihnen im Textfeld **Name** eingegebenen Wert aufgefüllt. Wenn Sie den Wert **Bezeichnung** zuerst eingeben, wird das Textfeld **Name** mit demselben Wert ausgefüllt.

Der Wert **Bezeichnung** wird auf der Benutzeroberfläche beim Anfordern von Eigenschaften angezeigt, zum Beispiel beim Hinzufügen einer Eigenschaft zu einem Blueprint als Eigenschaftsname.

Der Wert **Bezeichnung** kann mehr Zeichen als der Wert **Name** enthalten.

5 Wählen Sie im Abschnitt **Sichtbarkeit die Option **Alle Mandanten** oder **Dieser Mandant** aus, um zu bestimmen, wo die Eigenschaft verfügbar sein soll.**

Wenn Sie nur mit Mandantenadministratorrechten angemeldet sind, ist nur **Dieser Mandant** verfügbar. Wenn Sie nur mit Fabric-Administratorrechten angemeldet sind, ist nur **Alle Mandanten** verfügbar.

Die Einstellung **Alle Mandanten** oder **Dieser Mandant** kann nach dem Erstellen des Elements nicht geändert werden.

6 (Optional) Geben Sie im Textfeld **Beschreibung eine Eigenschaftenbeschreibung ein.**

Beschreiben Sie den Zweck der Eigenschaftsdefinition und alle anderen hilfreichen Informationen über die Eigenschaft.

7 (Optional) Geben Sie in das Textfeld **Reihenfolge der Anzeige einen Wert ein.**

Die eingegebene Zahl bestimmt, wie der Eigenschaftsname im Anforderungsformular angezeigt wird. Es gelten die folgenden Reihenfolgeregeln:

- Die Reihenfolge der Anzeige betrifft nur Eigenschaften, für die die Einstellungen **Eingabeaufforderung** oder **In Anforderungsformular anzeigen** konfiguriert sind.
- Alle Eigenschaften mit einer Anzeigereihenfolge erscheinen vor Eigenschaften ohne Anzeigereihenfolge.
- Eigenschaften mit einer Anzeigereihenfolge werden vom niedrigsten zum höchsten Wert sortiert. Negative Zahlen sind zulässig.
- Alle Eigenschaften werden alphabetisch angeordnet, wobei alle Eigenschaften mit Anzeigereihenfolge vor Eigenschaften ohne Anzeigereihenfolge erscheinen.
- Wenn zwei Eigenschaften denselben Wert für die Reihenfolge der Anzeige aufweisen, werden sie alphabetisch sortiert.

- 8 Wählen Sie aus dem Dropdown-Menü **Datentyp** einen Datentyp für die Eigenschaftsdefinition aus.

Tabelle 1-47. Datentyp

Datentyp	Anzeigen als
Boolean	Lässt einen booleschen Wert zu. Die Optionen für „Anzeigen als“ sind Kontrollkästchen und Ja/Nein .
Datum/Uhrzeit	Lässt einen im Datum/Uhrzeit-Format eingegebenen Wert zu. Die Option für „Anzeigen als“ ist Datum/Uhrzeit-Auswahl .
Dezimal	Lässt einen Ganzzahlwert oder Dezimalwert zu. Die Optionen für „Anzeigen als“ sind Dropdown , Schieberegler und Textfeld .
Integer	Lässt einen Ganzzahlwert zu. Die Optionen für „Anzeigen als“ sind Dropdown , Schieberegler und Textfeld .
Sichere Zeichenfolge	Lässt einen sicheren oder verschlüsselten Inhalt wie beispielsweise ein Kennwort zu. Als Option angezeigt wird entweder ein Kennwort, das bestätigt werden muss, oder ein Textfeld .
Zeichenfolge	Lässt einen Zeichenfolgenwert zu. Die Optionen für „Anzeigen als“ sind Dropdown , E-Mail , Hyperlink , Textbereich und Textfeld .

- 9 Wenn die Option **Erforderlich** verfügbar ist, wählen Sie **Ja** oder **Nein** aus dem Dropdown-Menü aus, um anzugeben, ob für diese Eigenschaft ein Wert eingegeben werden muss.
- 10 Wählen Sie einen Anzeigesteuerungstyp für diese Eigenschaft aus dem Dropdown-Menü **Anzeigen als** aus. Verfügbare Optionen werden von Ihrer **Datentyp**-Auswahl abgeleitet, wie oben in der Tabelle dargestellt.

Tabelle 1-48. Anzeigen als

Anzeigen als	Beschreibung
CheckBox	Stellt ein einzelnes Kontrollkästchen-Steuerelement bereit.
Datum/Uhrzeit-Auswahl	Stellt ein Datums- und Uhrzeitsteuerelement bereit, das dem Format <i>JJJJ-MM-TT</i> oder <i>MM/TT/JJJJ</i> und einer Uhrzeit im Format <i>HH:MM</i> entspricht (24-Stunden-Format oder gefolgt von AM oder PM).
Dropdown	Stellt ein Dropdown-Menü-Steuerelement bereit.
E-Mail	Stellt ein E-Mail-Steuerelement bereit.
Hyperlink	Zeigt einen Link mit dem Eigenschaftsanzeigenamen als Linktext und den Eigenschaftswert als URL an.
Schieberegler	Stellt ein Schiebereglersteuerelement für einen Wertebereich bereit.

Tabelle 1-48. Anzeigen als (Fortsetzung)

Anzeigen als	Beschreibung
Textbereich	Stellt einen Textbereich bereit, in den Informationen eingegeben werden oder in dem Informationen angezeigt werden.
Textfeld	Stellt ein Textfeld bereit, in das ein Wert eingegeben wird.
Ja/Nein	Gibt einen Ja- oder Nein-Wert an.

- 11 Wählen Sie je nach Ihrem Datentyp weitere Optionen aus, die in der Benutzeroberfläche angezeigt werden. Unten ist eine Beispielseite dargestellt:

The screenshot shows the 'Create Property Definition' form in the Administration tab. The form includes the following fields and options:

- Name:** test-runx
- Label:** MyContest.runx
- Visibility:** ☐ All tenants, ☒ This tenant
- Description:** (Empty text area)
- Display order:** (Dropdown menu)
- Data type:** Integer
- Required:** No
- Minimum value:** (Empty text field)
- Maximum value:** (Empty text field)
- Increment:** 1
- Display as:** Textbox

Buttons for OK and CANCEL are at the bottom.

- 12 Klicken Sie auf **OK**.

Ergebnisse

Die Eigenschaft wird erstellt und ist auf der Seite „Eigenschaftsdefinitionen“ verfügbar.

The screenshot shows the 'Property Definitions' page in the Administration tab. It displays a table with the following data:

Name	Label	Description	Type	Display advice	Visibility
test-runx	MyContest.runx		Integer	Textbox	This tenant

Erstellen einer benutzerdefinierten Eigenschaft, die anhand eines regulären Ausdrucks validiert wird

Sie erstellen eine benutzerdefinierte Eigenschaftsdefinition, die einen regulären Ausdruck auswertet, wenn Sie möchten, dass Nutzer des Servicekatalogs validierte Daten im Kataloganforderungsformular eingeben.

Hierdurch kann beispielsweise ein alphanumerisches Textfeld hinzugefügt werden, in dem der anfordernde Benutzer den Namen einer Anwendung oder Funktion eingibt, der auf fünf bis zehn Zeichen (ohne Sonderzeichen) begrenzt ist. Für dieses Szenario verwenden Sie die benutzerdefinierte Eigenschaft eines regulären Ausdrucks, die in etwa wie folgt konfiguriert ist: **`^[a-zA-Z0-9]{5,10}$`**.

Voraussetzungen

- Stellen Sie sicher, dass Sie über einen regulären Ausdruck verfügen, der die angegebenen Werte erwartungsgemäß validiert.
- Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsdefinitionen** aus.
- 2 Klicken Sie auf das Symbol **Neu** (+).
- 3 Geben Sie die Optionen ein.

Option	Beschreibung
Name	Geben Sie unter Zugrundelegung einer standardmäßigen Namenskonvention einen Wert für den neuen Eigenschaftsnamen ein, beispielsweise <code>my_grouping_prefix.my_property_name</code> .
Bezeichnung	Der Bezeichnung liegt der Name zugrunde. Sie können die Bezeichnung in einen besser lesbaren Namen ändern.
Sichtbarkeit	Die benutzerdefinierten Eigenschaften der Aktion sind nur im aktuellen Mandanten verfügbar. Sie müssen sie konfigurieren, wenn Sie bei dem jeweiligen Mandanten angemeldet sind, um sie in einem anderen Mandanten zur Verfügung zu stellen.
Beschreibung	Beschreiben Sie den Zweck der Eigenschaftsdefinition und alle anderen hilfreichen Informationen über die Eigenschaft.
Reihenfolge der Anzeige	<p>Die eingegebene Zahl bestimmt, wie der Eigenschaftsname im Anforderungsformular angezeigt wird. Es gelten die folgenden Reihenfolgeregeln:</p> <ul style="list-style-type: none"> ■ Die Reihenfolge der Anzeige betrifft nur Eigenschaften, für die die Einstellungen Eingabeaufforderung oder In Anforderungsformular anzeigen konfiguriert sind. ■ Alle Eigenschaften mit einer Anzeigereihenfolge werden vor Eigenschaften ohne Reihenfolgen-Index angezeigt. ■ Eigenschaften mit einer Anzeigereihenfolge werden vom niedrigsten zum höchsten Reihenfolgen-Indexwert sortiert. Sie können auch negative Zahlen verwenden. ■ Alle Eigenschaften werden alphabetisch sortiert, wobei alle Eigenschaften mit Anzeigereihenfolge vor Eigenschaften ohne Anzeigereihenfolge angezeigt werden. ■ Wenn zwei Eigenschaften denselben Wert für die Reihenfolge der Anzeige aufweisen, werden sie alphabetisch sortiert.

Option	Beschreibung
Datentyp	Wählen Sie im Dropdown-Menü Zeichenfolge aus.
Erforderlich	Wählen Sie im Dropdown-Menü Nein aus.
Anzeigen als	Wählen Sie im Dropdown-Menü Textfeld aus.
Benutzereingabe validieren	Geben Sie den regulären Ausdruck ein.

- 4 Geben Sie einen Wert im Textfeld für den Test ein, um zu überprüfen, ob der Ausdruck funktioniert.
- 5 Klicken Sie auf **OK**.

Ergebnisse

Die benutzerdefinierte Eigenschaftsdefinition wird der Liste hinzugefügt und kann nun einem Blueprint hinzugefügt werden.

Nächste Schritte

Fügen Sie die benutzerdefinierte Eigenschaft einem Blueprint für eine Maschine hinzu. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft oder einer Eigenschaftsgruppe mit der Registerkarte „Eigenschaften“ auf einer Maschinenkomponente eines Blueprints](#).

Erstellen einer benutzerdefinierten Eigenschaftsdefinition, die eine vRealize Orchestrator-Aktion enthält

Sie erstellen eine benutzerdefinierte Eigenschaftsdefinition, die eine vRealize Orchestrator-Aktion enthält, damit Sie die benutzerdefinierte Eigenschaft einem Blueprint hinzufügen können. Die Aktion wird ausgeführt, wenn der Benutzer des Servicekatalogs die benutzerdefinierte Eigenschaft im Anforderungsformular konfiguriert. Die Aktion ruft die Daten ab, die im Formular angezeigt werden.

Voraussetzungen

- Überprüfen Sie die Konfigurationsdetails für die benutzerdefinierte Eigenschaft, die Sie erstellen. Siehe [Konfigurationsdetails für die benutzerdefinierten Eigenschaftsdefinitionen für die vRealize Orchestrator-Aktion](#).
- Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsdefinitionen** aus.
- 2 Klicken Sie auf das Symbol **Neu** (+).

3 Geben Sie die Optionen ein.

Option	Beschreibung
Name	Konsultieren Sie die Konfigurationsdetails. Manche der benutzerdefinierten Eigenschaften erfordern spezifische Namen oder Formate. Verwenden Sie, wo immer möglich, eine standardmäßige Benennungskonvention für den Namen der neuen Eigenschaft, wie etwa mein_Gruppierungspräfix.mein_Eigenschaftsname .
Bezeichnung	Der Bezeichnung liegt der Name zugrunde. Sie können die Bezeichnung in einen besser lesbaren Namen ändern.
Sichtbarkeit	Die benutzerdefinierten Eigenschaften der Aktion sind nur im aktuellen Mandanten verfügbar. Sie müssen sie konfigurieren, wenn Sie bei dem jeweiligen Mandanten angemeldet sind, um sie in einem anderen Mandanten zur Verfügung zu stellen.
Beschreibung	Beschreiben Sie den Zweck der Eigenschaftsdefinition und alle anderen hilfreichen Informationen über die Eigenschaft.
Reihenfolge der Anzeige	<p>Die von Ihnen eingegebene Zahl steuert, wo im Anforderungsformular der Eigenschaftsname angezeigt wird. Es gelten die folgenden Reihenfolgeregeln:</p> <ul style="list-style-type: none"> ■ Die Reihenfolge der Anzeige betrifft nur Eigenschaften, für die die Einstellungen Eingabeaufforderung oder In Anforderungsformular anzeigen konfiguriert sind. ■ Alle Eigenschaften mit einer Anzeigereihenfolge werden vor Eigenschaften ohne Reihenfolgen-Index angezeigt. ■ Eigenschaften mit einer Anzeigereihenfolge werden vom niedrigsten zum höchsten Wert sortiert. Sie können auch negative Zahlen verwenden. ■ Alle Eigenschaften werden alphabetisch angeordnet, wobei alle Eigenschaften mit Anzeigereihenfolge vor Eigenschaften ohne Anzeigereihenfolge erscheinen. ■ Wenn zwei Eigenschaften denselben Wert für die Reihenfolge der Anzeige aufweisen, werden sie alphabetisch sortiert.

4 Konsultieren Sie die Konfigurationsdetails, um zu ermitteln, was Sie für die Werte angeben müssen.

Die folgenden Werte sind in den Konfigurationsdetails angegeben:

- Datentyp
- Anzeigen als
- Werte
- Aktionsordner
- Skriptaktion
- Eingabeparameter

5 Klicken Sie auf **OK**.

Ergebnisse

Die benutzerdefinierte Eigenschaftsdefinition wird der Liste hinzugefügt und kann nun einem Blueprint hinzugefügt werden.

Nächste Schritte

Fügen Sie die benutzerdefinierte Eigenschaft einem Blueprint hinzu. Ob Sie sie als Maschinen- oder Netzwerkeigenschaft hinzufügen, hängt von der jeweiligen Eigenschaft ab. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint](#).

Bindung von benutzerdefinierten Eigenschaften zum Erstellen einer hierarchischen Beziehung

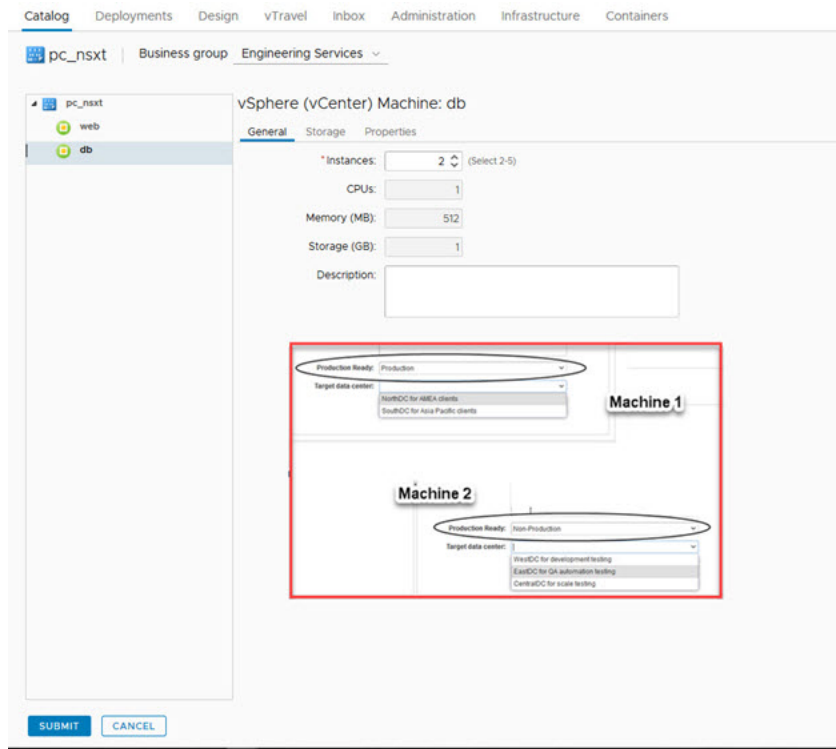
Zum Erstellen einer hierarchischen Beziehung zwischen benutzerdefinierten Eigenschaften binden Sie die übergeordnete Eigenschaft an die untergeordnete Eigenschaft. Wenn Sie die übergeordneten und die untergeordneten benutzerdefinierten Eigenschaften einem Blueprint hinzufügen, wählt der anfordernde Benutzer einen Wert für die übergeordnete Eigenschaft aus. Der Wert der übergeordneten Eigenschaft bestimmt die möglichen Werte für die untergeordnete Eigenschaft.

- Die übergeordnete benutzerdefinierte Eigenschaftsdefinition kann eine statische Liste oder ein externer Wert sein, der von einer vRealize Orchestrator-Aktion bestimmt wird. Sie stellt mögliche Eingabeparameter für eine untergeordnete Eigenschaftsdefinition zur Verfügung.
- Die untergeordnete benutzerdefinierte Eigenschaftsdefinition muss eine vRealize Orchestrator-Aktion aufrufen. In der untergeordneten benutzerdefinierten Eigenschaft binden Sie die übergeordnete benutzerdefinierte Eigenschaft, sodass sie einen Eingabeparameterwert zur Verfügung stellt.

Angenommen, Ihr Entwicklungsteam arbeitet auf Produktions- und Nicht-Produktionssystemen. Außerdem stehen Ihnen fünf Datacenter zur Verfügung. Drei der Datacenter dienen als Test-Datacenter zu Entwicklungszwecken und über die anderen beiden stellen Sie Ihren internen Kunden Dienste zur Verfügung. Um sicherzustellen, dass die Entwickler die gleichen Blueprints in jeder Umgebung bereitstellen können, d. h. in den Datacentern für Testzwecke oder für interne Kunden, erstellen und binden Sie zwei benutzerdefinierte Eigenschaftsdefinitionen. Mithilfe der ersten benutzerdefinierten Eigenschaft kann der Benutzer entweder die Produktions- oder die Nicht-Produktionsumgebung auswählen. Ausgehend von der Umgebung, die der Benutzer im Anforderungsformular auswählt, zeigt die zweite benutzerdefinierte Eigenschaft einen der folgenden Werte an:

- Die Liste der drei Test-Datacenter für die Nicht-Produktionsumgebungen.
- Die zwei Datacenter für interne Kunden als Produktionsumgebungen.

In folgendem Bildschirm wird die Kataloganforderungsseite für Maschine 1 (db) mit einem Abschnittsbereich dargestellt, der die Eigenschaft zum Binden von Maschine 1 (db) an die Eigenschaft in Maschine 2 (web) veranschaulicht.



Ziel dieses Verfahrens ist die Erstellung von zwei benutzerdefinierten Eigenschaften, die in einer hierarchischen Beziehung gebunden werden. Beim Einrichten der Bindung können Sie den entsprechenden Standort basierend auf dem ausgewählten Produktionsstatus auswählen.

Voraussetzungen

- Erstellen Sie für dieses Beispiel eine vRealize Orchestrator-Aktion, die Datencenter-Namen als Standortinformationen zur Verfügung stellt. Legen Sie für die Aktion den Namen „datacenters_prod“ fest, fügen Sie einen Eingabeparameter namens „prod“ als Zeichenfolgentyp hinzu und verwenden Sie dieses Beispielskript als Skript für die Aktion.

```
if(prod == null) {
    return ['Empty1', 'Empty2'];
} else if (prod.equals('nonprod')) {
    return ['WestDC for development testing', 'EastDC for QA automation testing', 'CentralDC for scale testing'];
} else {
    return ['NorthDC for AMEA clients', 'SouthDC for Asia Pacific clients'];
}
```

Informationen zum Entwickeln von Workflows sowie zum Erstellen und Verwenden von vRealize Orchestrator-Skriptaktionen finden Sie unter *Entwickeln mit VMware vRealize Orchestrator* in der [vRealize Orchestrator-Produktdokumentation](#).

- Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

- 1 Erstellen Sie eine benutzerdefinierte Eigenschaftsdefinition, sodass Benutzer die Produktions- oder Nicht-Produktionsumgebung auswählen können.
 - a Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsdefinitionen** aus.
 - b Konfigurieren Sie die benutzerdefinierte Eigenschaft „production.ready“.

Option	Beispielwerte
Name	production.ready
Bezeichnung	Environment
Beschreibung	Wählen Sie die Produktions- oder Nicht-Produktionsumgebung aus.
Reihenfolge der Anzeige	1 Sie wählen „1“ aus, um sicherzustellen, dass diese benutzerdefinierte Eigenschaft zuerst im Blueprint angezeigt wird.
Datentyp	String
Anzeigen als	Dropdown
Werte	Statische Liste
Werte der statischen Liste	Fügen Sie die folgenden Schlüsselpaare hinzu. <ul style="list-style-type: none"> ■ Produktion und prod ■ Nicht-Produktion und nonprod

- c Klicken Sie auf **OK**.

Die benutzerdefinierte Eigenschaft production.ready ist konfiguriert und kann jetzt verwendet werden.

- 2 Erstellen Sie eine benutzerdefinierte Eigenschaftsdefinition der vRealize Orchestrator-Aktion, die die Aktion für Ihren benutzerdefinierten Standort ausführt.

- Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsdefinitionen** aus.
- Konfigurieren Sie die benutzerdefinierte Eigenschaft „datacenter.target“.

The screenshot shows the 'Create Property Definition' window in vRealize Orchestrator. The left sidebar contains a navigation menu with options like Approval Policies, Directories Management, Users & Groups, Catalog Management, Property Dictionary, Property Definitions, Property Groups, Component Profiles, Reclamation, Branding, Notifications, Events, vRO Configuration, Active Directory Policies, Health, Message Board Whitelist, and Request History. The main area is titled 'Create Property Definition' and contains the following fields:

- Name:** datacenter.target (with a note: To avoid conflict with vRealize Automation properties, use a prefix such as a company or feature name followed by a dot for all custom property names.)
- Label:** Target datacenter
- Display as:** Dropdown
- Values:** Static list (selected), External values
- Visibility:** All tenants, This tenant (selected)
- Description:** Select the datacenter based on whether you are deploying a production or non-production blueprint.
- Display order:** 2 (with a note: You can control the order in which custom properties display on request forms. Set an order index of 1 to display this property at the top of the list.)
- Data type:** String
- Required:** No
- Script action:** com.vmware.library... (with a 'CHANGE...' button)
- Input parameters:** A table with columns Name, Bind, and Value. It contains one row: prod, Yes, production.ready.

At the bottom are 'OK' and 'CANCEL' buttons.

Option	Beispielwerte
Name	datacenter.target
Bezeichnung	Ziel-Datencenter
Beschreibung	Wählen Sie das Datencenter auf Grundlage davon aus, ob Sie einen Produktions- oder einen Nicht-Produktions-Blueprint bereitstellen.
Reihenfolge der Anzeige	2 Sie wählen „2“ aus, um sicherzustellen, dass diese benutzerdefinierte Eigenschaft nach der benutzerdefinierten Eigenschaft production.ready im Blueprint aufgelistet wird.
Datentyp	String
Anzeigen als	Dropdown
Werte	Externe Werte
Skriptaktion	Klicken Sie auf Auswählen und suchen Sie nach der Aktion „datacenters_prod“.

Die Tabelle mit Eingabeparametern enthält den Parameter „prod“.

- Wählen Sie in der Tabelle mit Eingabeparametern die Zeile „prod“ aus und klicken Sie auf **Bearbeiten**.

- d Aktivieren Sie das Kontrollkästchen **Binden**.
- e Wählen Sie im Dropdown-Menü **production.ready** aus.
- f Klicken Sie auf **OK**.
- g Klicken Sie auf **OK**.

Die benutzerdefinierte Eigenschaft `datacenter.target` ist konfiguriert und kann jetzt verwendet werden.

Nächste Schritte

- Fügen Sie aufgrund der Beziehung zwischen den beiden Eigenschaftsdefinitionen die beiden Eigenschaftsdefinitionen einer Eigenschaftsgruppe hinzu. Siehe [Erstellen einer Eigenschaftsgruppe](#).
- Fügen Sie Ihre Eigenschaftsgruppe für das Produktions-Datencenter einem Blueprint hinzu. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft oder einer Eigenschaftsgruppe mit der Registerkarte „Eigenschaften“ auf einer Maschinenkomponente eines Blueprints](#).

Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint

Sie können vielen Teilen von vRealize Automation benutzerdefinierte Eigenschaften hinzufügen, einschließlich Richtlinien, Business-Gruppen, Endpoints und Reservierungsrichtlinien. Jedoch werden von Ihnen als Eigenschaftsdefinitionen konfigurierte Anzeigoptionen nur von den Maschinen-Blueprints unterstützt. Das Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint ist eine einfache Möglichkeit zur Sicherstellung, dass die benutzerdefinierte Eigenschaft auf der Benutzeroberfläche so angezeigt wird, wie Sie dies in der Eigenschaftsdefinition vorgesehen haben.

Manche benutzerdefinierten Eigenschaften sind dem Blueprint der virtuellen Maschine auf der Registerkarte **Eigenschaften** zugeordnet, und andere befinden sich auf der Registerkarte **Netzwerk**.

- [Hinzufügen einer benutzerdefinierten Eigenschaft oder einer Eigenschaftsgruppe mit der Registerkarte „Eigenschaften“ auf einer Maschinenkomponente eines Blueprints](#)

Sie fügen eine benutzerdefinierte Eigenschaft als eine benutzerdefinierte Maschineneigenschaft hinzu, damit Benutzer des Servicekatalogs die Werte beim Anfordern des Elements auswählen oder konfigurieren können. Sie können einzelne Eigenschaften oder Eigenschaftsgruppen hinzufügen.

- [Hinzufügen einer benutzerdefinierten Eigenschaft unter Verwendung der Registerkarte „Netzwerk“ zu einer Blueprint-Maschinenkomponente](#)

Fügen Sie eine benutzerdefinierte Eigenschaft als benutzerdefinierte Netzwerkeigenschaft hinzu, sodass Benutzer des Servicekatalogs bei der Blueprint-Bereitstellung den erforderlichen Netzwerkprofilwert auswählen können.

Hinzufügen einer benutzerdefinierten Eigenschaft oder einer Eigenschaftsgruppe mit der Registerkarte „Eigenschaften“ auf einer Maschinenkomponente eines Blueprints

Sie fügen eine benutzerdefinierte Eigenschaft als eine benutzerdefinierte Maschineneigenschaft hinzu, damit Benutzer des Servicekatalogs die Werte beim Anfordern des Elements auswählen oder konfigurieren können. Sie können einzelne Eigenschaften oder Eigenschaftsgruppen hinzufügen.

In diesem Workflow fügen Sie die benutzerdefinierten Eigenschaften hinzu, um zu überprüfen, ob sie in Blueprints wie erwartet funktionieren. Sie können benutzerdefinierte Eigenschaften auch Business-Gruppen, Richtlinien und anderen Komponenten hinzufügen.

Voraussetzungen

- Vergewissern Sie sich, dass Sie die erforderliche Eigenschaftsdefinition erstellt haben. Siehe [Erstellen benutzerdefinierter Eigenschaftsdefinitionen](#).
- Vergewissern Sie sich beim Hinzufügen einer Eigenschaftsgruppe, dass Sie der Eigenschaftsgruppe die entsprechenden Eigenschaftsdefinitionen hinzugefügt haben. Siehe [Erstellen einer Eigenschaftsgruppe](#). Zum Testen der visuellen Funktionen der Eigenschaftsdefinitionen müssen Sie beim Hinzufügen der Eigenschaft zur Gruppe **In Anforderung anzeigen** auswählen.
- Wenn Sie eine vRealize Orchestrator-Aktion als benutzerdefinierte Eigenschaft hinzufügen, überprüfen Sie die Konfigurationsdetails, um sicherzustellen, dass Sie die benutzerdefinierte Eigenschaft am richtigen Speicherort hinzugefügt haben. Siehe [Konfigurationsdetails für die benutzerdefinierten Eigenschaftsdefinitionen für die vRealize Orchestrator-Aktion](#).
- Stellen Sie sicher, dass Sie den Blueprint, dem Sie die benutzerdefinierte Eigenschaft hinzufügen, erstellt haben. Siehe *Konfigurieren von vRealize Automation*.
- Melden Sie sich bei vRealize Automation als **Infrastrukturarchitekt** an.

Verfahren

- 1 Wählen Sie **Design > Blueprints** aus.
- 2 Wählen Sie den Blueprint aus, dem Sie die benutzerdefinierte Eigenschaft hinzufügen, und klicken Sie auf **Bearbeiten**.
- 3 Klicken Sie auf die Zielmaschinenkomponente.
Die Konfigurationsoptionen für die virtuelle Maschine werden auf der Arbeitsfläche angezeigt.
- 4 Klicken Sie auf die Registerkarte **Eigenschaften** und klicken Sie dann auf die Registerkarte **Benutzerdefinierte Eigenschaften** oder die Registerkarte **Eigenschaftsgruppen**.
 - Um eine benutzerdefinierte Eigenschaft hinzuzufügen, klicken Sie auf **Neu** und wählen Sie im Dropdown-Menü die Eigenschaftsdefinition aus.

Option	Beschreibung
Name	Name der ausgewählten benutzerdefinierten Eigenschaftsdefinition.
Wert	(Optional) Geben Sie einen Standardwert ein.

Option	Beschreibung
Verschlüsselt	Wenn Sie benutzerdefinierte Eigenschaften hinzufügen, die vRealize Orchestrator-Aktionen ausführen, verschlüsseln Sie den Wert nicht.
Überschreibbar	Wählen Sie diese Option aus, um sicherzustellen, dass der Benutzer, der die Anforderung stellt, im Anforderungsformular einen Wert auswählen kann.
In Anforderung anzeigen	Wählen Sie diese Option aus, um sicherzustellen, dass der Benutzer, der die Anforderung stellt, die Eigenschaft anzeigen und im Anforderungsformular einen Wert auswählen kann.

- Um eine Eigenschaftsgruppe hinzuzufügen, klicken Sie auf **Hinzufügen** und wählen Sie die Gruppe aus.

5 Klicken Sie auf **OK**.

Die benutzerdefinierte Eigenschaft wird dem Blueprint hinzugefügt.

6 Klicken Sie auf **Beenden**.

7 Veröffentlichen Sie den fertiggestellten Blueprint.

Ergebnisse

Der Blueprint enthält die benutzerdefinierte Eigenschaft.

Nächste Schritte

Testen Sie die benutzerdefinierte Eigenschaft im Anforderungsformular. Siehe [Überprüfen der benutzerdefinierten Eigenschaft im Kataloganforderungsformular](#).

Hinzufügen einer benutzerdefinierten Eigenschaft unter Verwendung der Registerkarte „Netzwerk“ zu einer Blueprint-Maschinenkomponente

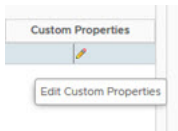
Fügen Sie eine benutzerdefinierte Eigenschaft als benutzerdefinierte Netzwerkeigenschaft hinzu, sodass Benutzer des Servicekatalogs bei der Blueprint-Bereitstellung den erforderlichen Netzwerkprofilwert auswählen können.

Voraussetzungen

- Vergewissern Sie sich, dass Sie über die erforderliche benutzerdefinierte Eigenschaftsdefinition verfügen. Siehe [Erstellen einer benutzerdefinierten Eigenschaftsdefinition, die eine vRealize Orchestrator-Aktion enthält](#).
- Wenn Sie eine vRealize Orchestrator-Aktion als benutzerdefinierte Eigenschaft hinzufügen, überprüfen Sie die Konfigurationsdetails, um sicherzustellen, dass Sie die benutzerdefinierte Eigenschaft am richtigen Speicherort hinzugefügt haben. Siehe [Konfigurationsdetails für die benutzerdefinierten Eigenschaftsdefinitionen für die vRealize Orchestrator-Aktion](#).
- Stellen Sie sicher, dass Sie den Blueprint, dem Sie die benutzerdefinierte Eigenschaft hinzufügen, erstellt haben. Siehe *Konfigurieren von vRealize Automation*.
- Melden Sie sich bei vRealize Automation als **Infrastrukturarchitekt** an.

Verfahren

- 1 Wählen Sie **Design > Blueprints** aus.
- 2 Wählen Sie den Blueprint aus, den Sie bearbeiten möchten.
Der Blueprint wird in der Design-Arbeitsfläche geöffnet.
- 3 Klicken Sie in der Design-Arbeitsfläche auf die virtuelle Maschinenkomponente, die Sie bearbeiten möchten.
Die Konfigurationsoptionen für die virtuelle Maschine werden auf der Arbeitsfläche angezeigt.
- 4 Klicken Sie auf die Registerkarte **Netzwerk** für die Maschinenkomponente.
- 5 Klicken Sie auf **Neu**, um eine neue Netzwerkzeile hinzuzufügen.
- 6 Wählen Sie in der neuen Zeile ein Netzwerk und einen Zuweisungstyp (statische IP oder DHCP) aus, geben Sie bei Verwendung einer statischen IP eine Adresse an und klicken Sie auf **OK**.
- 7 Klicken Sie in der neuen Zeile auf das Symbol „Bearbeiten“ in der Spalte „Benutzerdefinierte Eigenschaften“, um eine benutzerdefinierte Eigenschaft zuzuweisen.



- 8 Klicken Sie auf **Neu**, wählen Sie die benutzerdefinierte Eigenschaft aus, konfigurieren Sie die in der folgenden Tabelle beschriebenen Optionen und klicken Sie auf **OK**.

Option	Beschreibung
Name	Wählen Sie den Namen einer vorhandenen benutzerdefinierten Eigenschaft aus dem Dropdown-Menü aus.
Wert	(Optional) Geben Sie einen Standardwert ein.
Verschlüsselt	Wenn Sie benutzerdefinierte Eigenschaften hinzufügen, die vRealize Orchestrator-Aktionen ausführen, verschlüsseln Sie den Wert nicht.
Überschreibbar	Wählen Sie diese Option aus, um sicherzustellen, dass der Benutzer, der die Anforderung stellt, im Anforderungsformular einen Wert auswählen kann.
In Anforderung anzeigen	Wählen Sie diese Option aus, um sicherzustellen, dass der Benutzer, der die Anforderung stellt, die Eigenschaft anzeigen und im Anforderungsformular einen Wert auswählen kann.

Das Netzwerk wird mitsamt seiner konfigurierten benutzerdefinierten Eigenschaft zum Blueprint hinzugefügt.

- 9 Klicken Sie auf **Fertig stellen**.
- 10 Veröffentlichen Sie den fertiggestellten Blueprint.

Ergebnisse

Der Blueprint enthält die benutzerdefinierte Eigenschaft.

Nächste Schritte

Testen Sie die benutzerdefinierte Eigenschaft im Anforderungsformular. Siehe [Überprüfen der benutzerdefinierten Eigenschaft im Kataloganforderungsformular](#).

Überprüfen der benutzerdefinierten Eigenschaft im Kataloganforderungsformular

Als Ersteller der benutzerdefinierten Eigenschaftsdefinitionen, die vRealize Orchestrator-Aktionen ausführen, sollten Sie Ihre benutzerdefinierten Eigenschaften testen, um sicherzustellen, dass im Anforderungsformular die korrekten Werte angezeigt werden.

Voraussetzungen

- Fügen Sie die benutzerdefinierte Eigenschaft am geeigneten Speicherort im Blueprint hinzu. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint](#).
- Überprüfen Sie, ob Sie für die Nutzung des Blueprints berechtigt sind, sodass Sie die benutzerdefinierten Eigenschaften im Blueprint testen können. Siehe *Konfigurieren von vRealize Automation*.
- Melden Sie sich bei vRealize Automation als Benutzer mit Zugriff auf den Test-Blueprint an.

Verfahren

- 1 Klicken Sie auf **Katalog**, um die Katalogelemente anzuzeigen, zu deren Verwendung Sie berechtigt sind.

Veröffentlichte Blueprints werden auf der Seite „Katalog“ als Katalogelemente angezeigt.
- 2 Klicken Sie auf dem Katalogelement auf **Anfordern**.
- 3 Klicken Sie im Anforderungsformular auf die Maschine, der Sie die benutzerdefinierte Eigenschaft hinzugefügt haben.
- 4 Wählen Sie auf der Registerkarte **Eigenschaften** die benutzerdefinierte Eigenschaft aus und klicken Sie auf den Dropdown-Pfeil.

Die vRealize Orchestrator-Aktion wird ausgeführt und ruft die gemäß ihrer Konfiguration anzuzeigenden Werte ab. Vergewissern Sie sich, dass die erwarteten Werte angezeigt werden.

Nächste Schritte

Fügen Sie die benutzerdefinierte Eigenschaft nach Bedarf Ihren Produktions-Blueprints hinzu.

Konfigurationsdetails für die benutzerdefinierten Eigenschaftsdefinitionen für die vRealize Orchestrator-Aktion

Sie erstellen benutzerdefinierte Eigenschaftsdefinitionen, die vRealize Orchestrator-Aktionen zum Abrufen von Schlüssel/Wert-Paaren aus externen Dateien oder aus den vRealize Automation-Konfigurationsinformationen ausführen. Sie fügen benutzerdefinierte Eigenschaften zu Blueprints hinzu, sodass sie in den Kataloganforderungsformularen angezeigt werden.

Der Benutzer des Servicekatalogs, der das Element anfordert, kann einen Wert auswählen, der in der Bereitstellung enthalten sein soll. Wenn der Benutzer zur Auswahl eines Werts auf das Dropdown-Menü klickt, wird die vRealize Orchestrator-Aktion ausgeführt, und diese ruft die Daten ab, die dem Benutzer im Menü zur Auswahl angezeigt werden.

Die Konfigurations-Workflows für jede Eigenschaftsdefinition der vRealize Orchestrator-Aktionen verlaufen immer ähnlich, unterscheiden sich jedoch bezüglich mancher Details. Es gibt beispielsweise Unterschiede bei den Voraussetzungen und den Einschränkungen, und der Ort, an dem Sie die benutzerdefinierte Eigenschaft im Blueprint anwenden, kann variieren.

- **Benutzerdefinierte Eigenschaftsdefinition für Netzwerk**

Sie fügen eine benutzerdefinierte Eigenschaftsdefinition zum Abrufen von Netzwerknamen aus der vRealize Automation-Datenbank hinzu, wenn Sie möchten, dass Benutzer das Netzwerk im Anforderungsformular auswählen. Die benutzerdefinierte Eigenschaft für die Netzwerkauswahl verwendet zum Abrufen der Werte eine vRealize Orchestrator-Aktion.

- **Benutzerdefinierte Eigenschaftsdefinition für Reservierungsrichtlinie**

Sie fügen eine benutzerdefinierte Eigenschaftsdefinition zum Abrufen von Reservierungsrichtliniennamen hinzu, die für die die Anforderung stellenden Benutzer gelten, wenn diese die Richtlinie im Anforderungsformular auswählen. Die benutzerdefinierte Eigenschaftsdefinition für die Reservierungsrichtliniauswahl verwendet zum Abrufen der Werte eine vRealize Orchestrator-Aktion.

- **Benutzerdefinierte Eigenschaftsdefinition für PowerShell-Skript**

Sie fügen eine benutzerdefinierte Eigenschaft zum Ausführen eines PowerShell-Skripts hinzu, wenn Sie ein Skript verwenden möchten, um die Daten zum Ausfüllen der benutzerdefinierten Eigenschaft im Anforderungsformular abzurufen. Die benutzerdefinierte Eigenschaft für das PowerShell-Skript verwendet eine vRealize Orchestrator-Aktion zum Ausführen des Skripts und Abrufen der Werte.

- **Benutzerdefinierte Eigenschaftsdefinition für Datenbankabfrage**

Sie fügen eine benutzerdefinierte Eigenschaft zur Abfrage einer Datenbank hinzu, wenn Sie aus dieser Datenbank Werte zum Ausfüllen der benutzerdefinierten Eigenschaft im Anforderungsformular abrufen möchten. Die benutzerdefinierte Datenbankeigenschaft verwendet eine vRealize Orchestrator-Aktion zum Ausführen der Abfrage und Abrufen der Werte.

- **Benutzerdefinierte Eigenschaftsdefinition für benutzerdefinierte Aktion**

Sie fügen eine benutzerdefinierte Eigenschaft zum Abrufen von Daten aus einer Quelle mithilfe einer benutzerdefinierten vRealize Orchestrator-Aktion hinzu, wenn Sie möchten, dass Benutzer die abgerufenen Werte im Anforderungsformular auswählen.

Benutzerdefinierte Eigenschaftsdefinition für Netzwerk

Sie fügen eine benutzerdefinierte Eigenschaftsdefinition zum Abrufen von Netzwerknamen aus der vRealize Automation-Datenbank hinzu, wenn Sie möchten, dass Benutzer das Netzwerk im

Anforderungsformular auswählen. Die benutzerdefinierte Eigenschaft für die Netzwerkauswahl verwendet zum Abrufen der Werte eine vRealize Orchestrator-Aktion.

Einschränkungen

Planen Sie die folgenden Einschränkungen ein, wenn Sie die benutzerdefinierte Eigenschaft für die Netzwerkauswahl verwenden.

- Der Name der benutzerdefinierten Eigenschaft muss „VirtualMachine.Network0.Name“ lauten. Es muss dieser Name verwendet werden. Sie können nicht mehrere Eigenschaftsdefinitionen für die Netzwerkauswahl erstellen.
- Diese Aktion ruft alle Netzwerknamen für den anfordernden Benutzer ab, ohne zu überprüfen, ob sie für die Ziel-vCenter Server-Instanz gilt. Ein Benutzer eines Servicekatalogs wählt möglicherweise ein Netzwerk aus, das für das ausgewählte Ziel nicht anwendbar ist. Wird das falsche Netzwerk ausgewählt, schlägt die Kataloganforderung fehl.
- Die Aktion ruft Netzwerknamen ausschließlich für den Benutzer ab, der die Anforderung stellt. Wenn Sie eine Anforderung im Auftrag eines anderen Benutzers erstellen, gelten die Netzwerke für Sie. Beispiel: Netzwerk A und Netzwerk C sind der Business-Gruppe 1 zugeordnet, sodass BG 1-Benutzer nur die Netzwerke A und C, nicht jedoch Netzwerk B sehen.

Voraussetzungen

Überprüfen Sie bei Verwendung eines externen vRealize Orchestrator-Servers, dass dieser ordnungsgemäß eingerichtet wurde. Siehe *Konfigurieren von vRealize Automation*.

Konfigurationswerte für benutzerdefinierte Eigenschaft

Mithilfe dieser Optionen erstellen Sie die benutzerdefinierte Eigenschaft.

Tabelle 1-49. Konfigurationswerte für benutzerdefinierte Netzwerkeigenschaft

Option	Wert
Name	Sie müssen VirtualMachine.Network0.Name verwenden. Siehe Benutzerdefinierte Eigenschaften V .
Datentyp	String
Anzeigen als	Dropdown
Werte	Extern
Aktionsordner	com.vmware.vra.networks
Skriptaktion	getApplicableNetworks Bei dieser Skriptaktion handelt es sich um ein Beispielskript. Sie können spezifische Aktionen für Ihre Umgebung erstellen.
Eingabeparameter	Keine erforderlichen Parameter.

Blueprint-Konfiguration

Fügen Sie die benutzerdefinierte Eigenschaft auf der Registerkarte **Netzwerk** des Blueprints hinzu. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft unter Verwendung der Registerkarte „Netzwerk“ zu einer Blueprint-Maschinenkomponente](#).

Benutzerdefinierte Eigenschaftsdefinition für Reservierungsrichtlinie

Sie fügen eine benutzerdefinierte Eigenschaftsdefinition zum Abrufen von Reservierungsrichtliniennamen hinzu, die für die die Anforderung stellenden Benutzer gelten, wenn diese die Richtlinie im Anforderungsformular auswählen. Die benutzerdefinierte Eigenschaftsdefinition für die Reservierungsrichtlinienauswahl verwendet zum Abrufen der Werte eine vRealize Orchestrator-Aktion.

Einschränkungen

Planen Sie die folgenden Einschränkungen ein, wenn Sie die benutzerdefinierte Eigenschaft für die Reservierungsrichtlinienauswahl verwenden.

- Der Name der benutzerdefinierten Eigenschaft muss „ReservationPolicyID“ lauten. Es muss dieser Name verwendet werden. Sie können nicht mehrere Eigenschaftsdefinitionen für die Reservierungsrichtlinienauswahl erstellen.
- Die Aktion ruft alle für den Benutzer, der die Anforderung stellt, anwendbaren Reservierungsrichtlinien ab, ohne zu überprüfen, ob sie für den Ziel-Endpoint gilt, z. B. eine vCenter Server-Instanz oder eine andere Plattform. Ein Benutzer eines Servicekatalogs wählt möglicherweise eine Reservierung aus, die für das ausgewählte Blueprint-Zielsystem nicht anwendbar ist. Wenn der Benutzer die falsche Reservierung auswählt, schlägt die Kataloganforderung fehl.
- Die Aktion ruft Reservierungsrichtlinien ausschließlich für den Benutzer ab, der die Anforderung stellt. Wenn Sie eine Anforderung im Auftrag eines anderen Benutzers erstellen, gelten die Reservierungsrichtlinien für Sie. Beispiel: Reservierung 1 und Reservierung 3 sind der Business-Gruppe 1 zugeordnet, sodass BG 1-Benutzer nur die Reservierungen 1 und 3, nicht jedoch Reservierung 2 sehen.

Voraussetzungen

Überprüfen Sie bei Verwendung eines externen vRealize Orchestrator-Servers, dass dieser ordnungsgemäß eingerichtet wurde. Siehe *Konfigurieren von vRealize Automation*.

Konfigurationswerte für benutzerdefinierte Eigenschaft

Mithilfe dieser Optionen erstellen Sie die benutzerdefinierte Eigenschaft.

Tabelle 1-50. Konfigurationswerte für benutzerdefinierte Eigenschaft für Reservierungsrichtlinie

Option	Wert
Name	Sie müssen „ReservationPolicyID“ verwenden.
Datentyp	String
Anzeigen als	Dropdown
Werte	Extern

Tabelle 1-50. Konfigurationswerte für benutzerdefinierte Eigenschaft für Reservierungsrichtlinie (Fortsetzung)

Option	Wert
Aktionsordner	com.vmware.vra.reservations
Skriptaktion	getApplicableReservationPolicies Bei dieser Skriptaktion handelt es sich um ein Beispielskript. Sie können spezifische Aktionen für Ihre Umgebung erstellen.
Eingabeparameter	Keine erforderlichen Parameter.

Blueprint-Konfiguration

Sie können der Registerkarte **Eigenschaften** des Blueprints eine benutzerdefinierte Eigenschaft hinzufügen, um die Eigenschaft dem gesamten Blueprint zuzuordnen.

Benutzerdefinierte Eigenschaftsdefinition für PowerShell-Skript

Sie fügen eine benutzerdefinierte Eigenschaft zum Ausführen eines PowerShell-Skripts hinzu, wenn Sie ein Skript verwenden möchten, um die Daten zum Ausfüllen der benutzerdefinierten Eigenschaft im Anforderungsformular abzurufen. Die benutzerdefinierte Eigenschaft für das PowerShell-Skript verwendet eine vRealize Orchestrator-Aktion zum Ausführen des Skripts und Abrufen der Werte.

Als Cloud-Administrator verfügen Sie beispielsweise über ein PowerShell-Skript, das Benutzer-IDs aus dem bei vRealize Automation registrierten Active Directory abrufen. Der Zweck des Skripts besteht darin, „John Smith“ abzurufen und anzuzeigen, wenn der tatsächliche Wert in Active Directory „JSmith01“ lautet.

Die Verwendung der PowerShell-Skriptaktion hat den Vorteil, dass es einen zentralen Speicherort für das Skript gibt. Sie können das Skript entweder auf einem zentralen Server speichern und dann auf virtuellen Zielmaschinen ausführen, oder Sie können es in vRealize Orchestrator speichern und dann auf den Zielmaschinen ausführen. Ein zentraler Speicherort bedeutet einen geringeren Zeitaufwand für die Pflege. Wenn Sie die Skripte in vRealize Orchestrator speichern und Sichern und Wiederherstellen konfiguriert haben, ist sichergestellt, dass Sie die Skripte bei einem Systemausfall wiederherstellen können.

Voraussetzungen

Vergewissern Sie sich, dass Sie über ein funktionierendes PowerShell-Skript verfügen, das Schlüssel/Wert-Paare zurückgibt. Das Skript muss auf einem zugänglichen Server verfügbar sein oder in vRealize Orchestrator hochgeladen werden.

Konfigurationswerte für benutzerdefinierte Eigenschaft

Mithilfe dieser Optionen erstellen Sie die benutzerdefinierte Eigenschaft.

Tabelle 1-51. Konfigurationswerte für benutzerdefinierte Eigenschaft für PowerShell-Skript

Option	Wert
Name	Sie können eine beliebige Zeichenfolge verwenden.
Datentyp	String
Anzeigen als	Dropdown
Werte	Extern
Aktionsordner	com.vmware.vra.powershell
Skriptaktion	<p>Wählen Sie eine Aktion basierend darauf, wo sich das PowerShell-Skript befindet.</p> <ul style="list-style-type: none"> ■ Wenn sich das PowerShell-Skript auf einem zentralen Server befindet, verwenden Sie „executeExternalPowerShellScriptOnHostByName“. ■ Wenn das PowerShell-Skript in vRealize Orchestrator hochgeladen wurde, verwenden Sie „executePowerShellScriptFromResourceOnHostByName“. <p>Diese Skriptaktionen sind Beispielskripte. Sie können spezifische Aktionen für Ihre Umgebung erstellen.</p> <p>Das PowerShell-Beispielskript <code>Resources/Sample/vRA/PowerShell/countries.ps1</code> wird im vRealize Orchestrator-Client als Referenz zur Verwendung mit der Aktion „executePowerShellScriptFromResourceOnHostByName“ verwendet.</p>
Eingabeparameter	<p>Konfigurieren Sie die Eingabeparameter basierend auf der ausgewählten Aktion.</p> <p>Definieren von Parametern</p> <ul style="list-style-type: none"> ■ Wenn Sie „executeExternalPowerShellScriptOnHostByName“ verwenden: <ul style="list-style-type: none"> ■ hostName: Name des zentralen Servers, auf dem sich das Skript befindet. ■ externalPowerShellScript: Pfad zur PowerShell-Datei auf dem Host. ■ Argumente: Parameter, die an das Skript weitergegeben werden. Argumente werden durch Komma getrennt, z. B.: <code>Argument1,Argument2</code>. ■ Wenn Sie „executePowerShellScriptFromResourceOnHostByName“ verwenden: <ul style="list-style-type: none"> ■ vRealize Orchestrator. Name der vRealize Orchestrator-Instanz, die Sie als Host verwenden. ■ scriptResourcePath: Pfad zur PowerShell-Datei auf dem Host. ■ scriptResourceName: Pfad der PowerShell-Datei als hochgeladene Ressource in vRealize Orchestrator.

Blueprint-Konfiguration

Sie können der Registerkarte **Eigenschaften** des Blueprints eine benutzerdefinierte Eigenschaft hinzufügen, um die Eigenschaft dem gesamten Blueprint zuzuordnen.

Benutzerdefinierte Eigenschaftsdefinition für Datenbankabfrage

Sie fügen eine benutzerdefinierte Eigenschaft zur Abfrage einer Datenbank hinzu, wenn Sie aus dieser Datenbank Werte zum Ausfüllen der benutzerdefinierten Eigenschaft im Anforderungsformular abrufen möchten. Die benutzerdefinierte Datenbankeigenschaft

verwendet eine vRealize Orchestrator-Aktion zum Ausführen der Abfrage und Abrufen der Werte.

Diese Aktion wird für die folgenden Datenbanken unterstützt:

- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL

Einschränkungen

Alle abgerufenen Werte werden in Zeichenfolgen umgewandelt.

Voraussetzungen

Stellen Sie sicher, dass das SQL-Plug-in für vRealize Orchestrator installiert und so konfiguriert ist, dass eine Verbindung zur Zieldatenbank hergestellt wird.

Konfigurationswerte für benutzerdefinierte Eigenschaft

Mithilfe dieser Optionen erstellen Sie die benutzerdefinierte Eigenschaft.

Tabelle 1-52. Konfigurationswerte für benutzerdefinierte Eigenschaft für Datenbankabfrage

Option	Wert
Name	Sie können eine beliebige Zeichenfolge verwenden.
Datentyp	String
Anzeigen als	Dropdown
Werte	Extern
Aktionsordner	com.vmware.vra.sql
Skriptaktion	executeSQLSelectOnDatabase Bei dieser Skriptaktion handelt es sich um ein Beispielskript. Sie können spezifische Aktionen für Ihre Umgebung erstellen.
Eingabeparameter	<ul style="list-style-type: none"> ■ databaseName: Name der Datenbank, mit der vRealize Orchestrator verbunden ist. ■ sqlSelectQuery: Die Select-Abfrage in SQL, die Sie auf der Datenbank ausführen, um die Werte abzurufen. Beispiel: select * <Tabellenname>. ■ keyColumnName: Name der Datenbankspalte, die als Schlüssel für das Schlüssel/Wert-Paar fungiert. ■ valueColumnName: Name der Datenbankspalte, aus der Sie die Werte abrufen.

Blueprint-Konfiguration

Sie können der Registerkarte **Eigenschaften** des Blueprints eine benutzerdefinierte Eigenschaft hinzufügen, um die Eigenschaft dem gesamten Blueprint zuzuordnen.

Benutzerdefinierte Eigenschaftsdefinition für benutzerdefinierte Aktion

Sie fügen eine benutzerdefinierte Eigenschaft zum Abrufen von Daten aus einer Quelle mithilfe einer benutzerdefinierten vRealize Orchestrator-Aktion hinzu, wenn Sie möchten, dass Benutzer die abgerufenen Werte im Anforderungsformular auswählen.

Einschränkungen

Zu den unterstützten Skriptaktionen gehören:

- Beliebige und Array/Beliebige
- Array/Zeichenfolge und Array/Eigenschaften, wenn Sie den Datentyp „Zeichenfolge“ im Definitionsformular auswählen
- Array/Zahl, wenn Sie den Datentyp „Ganzzahl“ oder „Dezimal“ im Definitionsformular auswählen

Voraussetzungen

Vergewissern Sie sich, dass Sie über eine funktionierende vRealize Orchestrator-Aktion verfügen. Informationen zum Entwickeln von Workflows sowie zum Erstellen und Verwenden von vRealize Orchestrator-Skriptaktionen finden Sie unter *Entwickeln mit VMware vCenter Orchestrator*.

Das Aktionsskript muss die Werte für die Eingabeparameter akzeptieren. Sie können die Werte als Schlüssel/Wert-Paare konfigurieren. Sie können mithilfe von Schlüssel/Wert-Paaren für den Benutzer lesbare Namen anstelle weniger gut lesbarer Bezeichner darstellen.

Konfigurationswerte für benutzerdefinierte Eigenschaft

Mithilfe dieser Optionen erstellen Sie die benutzerdefinierte Eigenschaft.

Tabelle 1-53. Konfigurationswerte für benutzerdefinierte Eigenschaft für benutzerdefinierte Aktion

Option	Wert
Name	Sie können eine beliebige Zeichenfolge verwenden.
Datentyp	Dezimal, Integer oder Zeichenfolge
Anzeigen als	Dropdown
Werte	Extern
Aktionsordner	Speicherort Ihrer benutzerdefinierten Aktion.
Skriptaktion	Name Ihrer benutzerdefinierten Aktion.
Eingabeparameter	Hängt von Ihrer benutzerdefinierten Aktion ab.

Blueprint-Konfiguration

Normalerweise fügen Sie die benutzerdefinierte Eigenschaft auf der Registerkarte „Eigenschaften“ des Blueprints hinzu. Ob Sie sie auf der Registerkarte „Eigenschaften“ hinzufügen, hängt von Ihrer Aktion ab. Siehe [Hinzufügen einer benutzerdefinierten Eigenschaft zu einem Blueprint](#).

Verwenden von Eigenschaftsgruppen

Sie können Eigenschaftsgruppen erstellen, um einzelne Eigenschaften in einer Einheit zu erfassen.

Eigenschaftsgruppen sind logische und wiederverwendbare Gruppen von Eigenschaften, die von Ihnen erstellte Eigenschaftsdefinitionen oder bereitgestellte benutzerdefinierte Eigenschaften enthalten können. Mit Eigenschaftsgruppen wird das Hinzufügen von Eigenschaften zu Blueprints oder anderen vRealize Automation-Elementen, für die sie verfügbar sind, vereinfacht. Hiermit können logische Gruppierungen von Eigenschaften effizienter hinzugefügt werden, als dies beim Hinzufügen einzelner Eigenschaften der Fall ist.

Eine Eigenschaftsgruppe enthält in der Regel Eigenschaften, die häufig zusammen verwendet werden. Beispielsweise können Sie die Eigenschaftsgruppe „WimImagingProperties“ erstellen, die die Eigenschaften enthält, die häufig für die WIM-basierte Bereitstellung verwendet werden:

- `Image.ISO.Location`
- `Image.ISO.Name`
- `Image.Network.Password`
- `Image.Network.User`
- `Image.WIM.Index`
- `Image.WIM.Name`
- `Image.WIM.Path`

Sie können auch eine Eigenschaftsgruppe für die vCloud Air- oder vCloud Director-Maschinenbereitstellung erstellen, die die folgenden Eigenschaften enthält:

- `VirtualMachine.Network0.Name`
- `VCloud.Template.MakeIdenticalCopy`
- `VMware.SCSI.Type`
- `Sysprep.Identification.DomainAdmin`
- `Sysprep.Identification.DomainAdminPassword`
- `Sysprep.Identification.JoinDomain`

Mit dem Befehl `vra content list --type property-definition` von vRealize CloudClient können Sie alle Eigenschaftsdefinitionen im aktuellen vRealize Automation-Mandanten auflisten. Mit dem Befehl `vra content list --type property-group` von vRealize CloudClient können Sie auch alle Eigenschaftsgruppen auflisten.

Erstellen einer Eigenschaftsgruppe

Sie können bestimmte benutzerdefinierte Eigenschaften zu Eigenschaftsgruppen zusammenfassen, damit Sie auf einfachere Weise mehrere benutzerdefinierte Eigenschaften zu Blueprints hinzufügen können.

Voraussetzungen

Melden Sie sich bei vRealize Automation als **Mandantenadministrator** oder **Fabric-Administrator** an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Eigenschaftsgruppen** aus.

- 2 Klicken Sie auf **Neu** (+).

- 3 Geben Sie den Namen und die ID der neuen Eigenschaftsgruppe ein.

Wenn Sie zuerst eine Eingabe im Feld **Name** vornehmen, wird das Textfeld **ID** mit demselben Wert aufgefüllt.

- 4 Wählen Sie im Abschnitt **Sichtbarkeit** die Option **Alle Mandanten** oder **Dieser Mandant** aus, um zu bestimmen, wo die Eigenschaft verfügbar sein soll.

Wenn Sie nur mit Mandantenadministratorrechten angemeldet sind, ist nur **Dieser Mandant** verfügbar. Wenn Sie nur mit Fabric-Administratorrechten angemeldet sind, ist nur **Alle Mandanten** verfügbar.

Die Einstellung **Alle Mandanten** oder **Dieser Mandant** kann nach dem Erstellen des Elements nicht geändert werden.

- 5 (Optional) Geben Sie eine Beschreibung der Eigenschaftsgruppe ein, beispielsweise **My Cloning Properties vSphere**.

- 6 Klicken Sie auf **Neu** und fügen Sie eine Eigenschaft zur Gruppe hinzu.

Option	Beschreibung
Name	Fügen Sie eine neue Eigenschaft hinzu oder wählen Sie eine vorhandene Eigenschaft aus dem Dropdown-Menü aus. Geben Sie beispielsweise VirtualMachine.Storage.ReserveMemory ein.
Wert	(Optional) Geben Sie einen Standard-Eigenschaftswert ein. Geben Sie beispielsweise True ein.
Verschlüsselt	Wählen Sie diese Option, um anzugeben, dass der Eigenschaftswert verschlüsselt werden soll. Wenn der Wert beispielsweise ein Kennwort oder ein sonstiger sicherer Eintrag sein soll, werden mit der Option „Verschlüsselt“ die Zeichen des Werts ausgeblendet. Wenn Sie benutzerdefinierte Eigenschaften hinzufügen, die vRealize Orchestrator-Aktionen ausführen, verschlüsseln Sie den Wert nicht.
In Anforderung anzeigen	Wählen Sie diese Option aus, um anzugeben, dass der Benutzer, der die Anforderung stellt, bei der Anforderung der Maschinenbereitstellung die Eigenschaft anzeigen und im Anforderungsformular einen Wert auswählen kann.

- 7 Klicken Sie auf **OK**, um die Eigenschaft zur Gruppe hinzuzufügen.

- 8 Fügen Sie der Gruppe zusätzliche Eigenschaften hinzu.

- 9 Klicken Sie auf **OK**.

Definieren der Einstellungen für Komponentenprofile

Mithilfe von Komponentenprofileinstellungen können Sie Verwaltungsfunktionen für erweiterte Eigenschaften in vRealize Automation-Blueprints konfigurieren. Bereitsteller können dann die Size- und Image-Komponentenprofile in einem Blueprint verwenden, um vordefinierte Wertsätze auszuwählen.

Mithilfe der Komponentenprofile Size und Image und ihrer angegebenen Wertsätze können Sie logische Gruppierungen wie „Klein“, „Mittel“ und „Groß“ oder „Dev“, „Test“ und „Produktion“ zuordnen. Verwenden Sie diese Einstellungen, um die Anzahl der zu verwaltenden Blueprints zu reduzieren.

Mit einem Komponentenprofil werden Einstellungen für eine vSphere-Maschinenkomponente in einem Blueprint definiert. Sie können beispielsweise ein Komponentenprofil für die Bereitstellung einer kleinen virtuellen Maschine definieren. Ein weiteres Komponentenprofil können Sie dann für die Bereitstellung einer großen virtuellen Maschine definieren. Mithilfe von vRealize Automation können Sie die folgenden Komponentenprofiltypen definieren:

- Größe

Siehe [Konfigurieren von Größeneinstellungen des Komponentenprofils für Katalogbereitstellungen](#).

- Image

Siehe [Konfigurieren von Komponentenprofil-Image-Einstellungen für Katalogbereitstellungen](#).

Sie können mehrere benannte Wertsätze in den Komponentenprofiltypen Size und Image definieren und einen oder mehrere Wertsätze den Maschinenkomponenten in einem Blueprint hinzufügen. Jeder Wertsatz, den Sie für den Komponentenprofiltyp definieren, enthält die folgenden konfigurierbaren Einstellungen:

- Der Name, den Auftraggeber sehen, wenn sie eine Maschine bereitstellen
- Der eindeutige Bezeichner für den Mandanten
- Beschreibung
- Auswahlmöglichkeiten für jede Option im Wertsatz

Sie können keine anderen Komponentenprofiltypen definieren.

Wenn Sie die Bereitstellung anfordern, können Sie verfügbare Optionen für Size und Image auswählen. Wenn Sie eine Wertmenge auswählen, werden die entsprechenden Eigenschaftswerte mit der Anforderung verknüpft.

Konfigurieren von Komponentenprofil-Image-Einstellungen für Katalogbereitstellungen

Sie können die Image-Einstellung des Komponentenprofils konfigurieren, um Build-Informationen für vSphere-Maschinenkomponenten im Blueprint zu steuern.

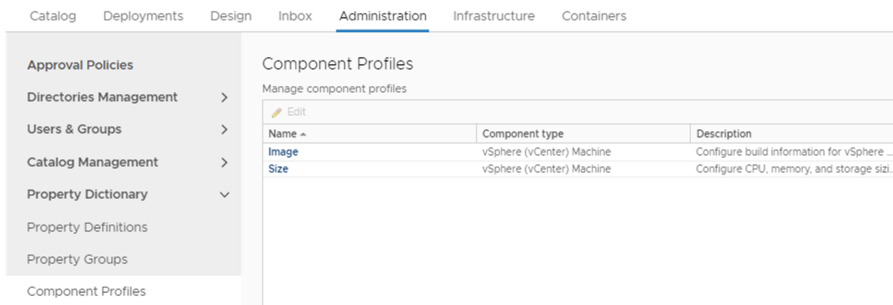
Nachdem Sie Wertsätze für das Komponentenprofil **Image** definiert haben, können Sie dem Komponentenprofil für eine vSphere-Maschinenkomponente in einem Blueprint einen oder mehrere Wertsätze hinzufügen. Benutzer können dann einen **Image**-Wertsatz auswählen, wenn sie ein Katalogelement anfordern.

Voraussetzungen

Melden Sie sich bei vRealize Automation als Administrator mit **Mandantenadministrator**- und **IaaS-Administrator**-Zugriffsrechten an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Komponentenprofile** aus.



- 2 Klicken Sie in der Spalte „Name“ auf **Image**.

Es werden Informationen über die angegebene Image-Komponenteneigenschaft angezeigt.

- 3 Klicken Sie auf die Registerkarte **Wertsätze**.

- 4 Klicken Sie zum Definieren eines neuen Wertsatzes auf **Neu** und konfigurieren Sie dann die **Image**-Einstellungen.

- a Geben Sie in das Feld **Anzeigename** einen Wert ein, um das ValueSet-Trennzeichen hinzuzufügen (zum Beispiel **CloneA**).
- b Akzeptieren Sie den im Textfeld **Name** angezeigten Standardwert (zum Beispiel **ValueSet.CloneA**) oder geben Sie einen benutzerdefinierten Namen ein.
- c Geben Sie im Textfeld **Beschreibung** eine Beschreibung ein. Beispiel: **Build-Einstellungen für Klon-Szenario A**.
- d Wählen Sie im Dropdown-Menü **Status** die Option **Aktiv** oder **Inaktiv** aus.
Wählen Sie **Aktiv** aus, damit der Wertsatz im Anforderungsformular für die Katalogbereitstellung sichtbar ist.
- e Wählen Sie **Server** oder **Desktop** als Blueprint-Typ aus.

- f Wählen Sie die Build-Aktion für diesen Wertsatz aus, zum Beispiel **Klonen**.

Weitere Aktionen sind:

- **Erstellen**
- **Klonen**
- **Verknüpfter Klon**
- **NetApp FlexClone**

- g Wählen Sie den Bereitstellungsworkflow CloneWorkflow aus.

- h (Optional) Wählen Sie eine Quellmaschine aus, von der aus geklont werden soll, zum Beispiel **centos7264**.

- i (Optional) Geben Sie den Pfad zu einer vSphere-Anpassungsspezifikation ein.

- 5 Klicken Sie auf **Speichern**.

- 6 Wenn Sie mit Ihren Einstellungen zufrieden sind, klicken Sie auf **Fertig stellen**.

The screenshot shows the vRealize Automation console interface. The top navigation bar includes 'Catalog', 'Deployments', 'Design', 'vTravel', 'Inbox', 'Administration' (selected), 'Infrastructure', and 'Containers'. The left sidebar lists various management sections like 'Approval Policies', 'Directories Management', 'Users & Groups', 'Catalog Management', 'Property Dictionary', 'Property Definitions', 'Property Groups', 'Component Profiles', 'Reclamation', 'Branding', 'Notifications', 'Events', 'vRO Configuration', 'Active Directory Policies', 'Health', and 'Message Board Whitelist'. The main content area is titled 'Edit Component Profile: Image' and contains a sub-header 'Manage component profile value set that will be available while creating blueprints'. Below this, the 'Value Sets' tab is selected, showing a table with columns 'Name', 'Display name', 'Description', and 'Status'. The table contains one entry: 'CloneA' with status 'Active'. Below the table is the 'Create Value Set' form with the following fields:

- * Display name: CloneA
- * Name: ValueSet CloneA
- Description: Build settings for cloning scenario A
- Status: Active
- * Blueprint type: Server
- * Action: Clone
- * Provisioning workflow: CloneWorkflow
- Clone from: centos72x64
- Customization spec: (empty)

 At the bottom of the form are buttons for 'SAVE', 'FINISH', and 'CANCEL'.

Nächste Schritte

Fügen Sie über die Registerkarte **Profile** einer vSphere-Maschinenkomponente einen oder mehrere Wertsätze zum Komponentenprofil Image hinzu. Siehe *Konfigurieren von vRealize Automation*.

Konfigurieren von Größeneinstellungen des Komponentenprofils für Katalogbereitstellungen

Sie können die Einstellung Size des Komponentenprofils konfigurieren, um CPU-, Arbeitsspeicher- und Speicherdimensionierung für vSphere-Maschinenkomponenten im Blueprint anzugeben.

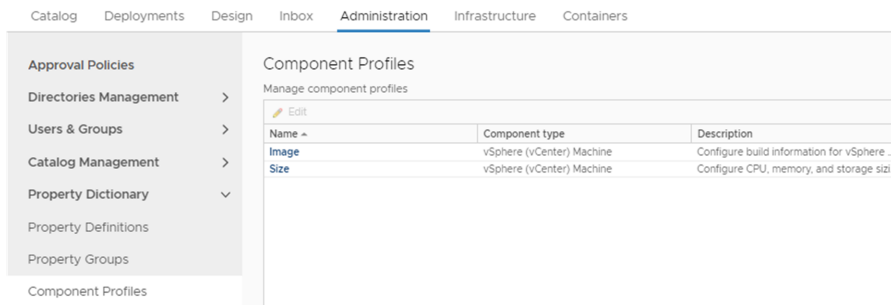
Nachdem Sie Wertsätze für das Komponentenprofil **Size** definiert haben, können Sie einem Komponentenprofil für eine vSphere-Maschinenkomponente in einem Blueprint einen oder mehrere Wertsätze hinzufügen. Benutzer können dann einen Wertsatz vom Typ **Size** auswählen, wenn sie ein Katalogelement anfordern.

Voraussetzungen

Melden Sie sich bei vRealize Automation als Administrator mit **Mandantenadministrator-** und **IaaS-Administrator-**Zugriffsrechten an.

Verfahren

- 1 Wählen Sie **Administration > Eigenschaftenwörterbuch > Komponentenprofile** aus.



- 2 Klicken Sie in der Spalte „Name“ auf **Größe**.

Informationen über das angegebene Komponentenprofil **Size** werden auf der Registerkarte **Allgemein** angezeigt.

- 3 Klicken Sie auf die Registerkarte **Wertsätze**.

- 4 Um einen neuen Wertsatz beispielsweise für eine große Bereitstellung zu definieren, klicken Sie auf **Neu** und konfigurieren Sie die Einstellungen für **Size**.

- a Geben Sie in das Feld **Anzeigename** einen Wert ein, um das ValueSet-Trennzeichen hinzuzufügen (zum Beispiel **small_1**).
- b Akzeptieren Sie den im Textfeld **Name** angezeigten Standardwert (zum Beispiel **ValueSet.small_1**) oder geben Sie einen benutzerdefinierten Namen ein.
- c Geben Sie in das Feld **Beschreibung** eine Beschreibung ein, zum Beispiel **kleine Bereitstellung**.
- d Wählen Sie im Dropdown-Menü „Status“ die Option **Aktiv** oder **Inaktiv** aus.
Wählen Sie **Aktiv** aus, damit der Wertsatz im Anforderungsformular für die Katalogbereitstellung sichtbar ist.
- e Geben Sie die Anzahl der virtuellen CPUs ein, auf denen die Bereitstellung ausgeführt werden kann, zum Beispiel „1“.

- f Geben Sie die Größe des Arbeitsspeichers ein, der von den virtuellen Maschinen in der Bereitstellung verwendet werden kann, zum Beispiel „2 MB“.
- g Geben Sie die Speichermenge ein, die von den virtuellen Maschinen in der Bereitstellung verwendet werden kann, zum Beispiel „1 GB“.

5 Klicken Sie auf **Speichern**.

6 Wenn Sie mit Ihren Einstellungen zufrieden sind, klicken Sie auf **Fertig stellen**.

Nächste Schritte

Fügen Sie über die Registerkarte **Profile** einer vSphere-Maschinenkomponente einen oder mehrere Wertsätze zum Komponentenprofil Size hinzu. Siehe *Konfigurieren von vRealize Automation*.