



© belkova / 123RF.com

Sovereign Cloud Stack – eine echte Alternative

Freiheitskämpfer

Der Sovereign Cloud Stack verspricht nichts weniger als die Befreiung von den Fesseln eines Hersteller-Lock-ins. Cloud-Nutzer, die auf diese Technik setzen, können den Anbieter frei wählen und bei Bedarf wechseln, ganz ohne umzulernen oder zuzuzahlen. Kurt Garloff

„Es liegt nicht an irgendwelchen Hürden, dass die Deutschen nicht digitaler werden. Sie haben schlicht keine Lust dazu.“ Diesen Eindruck hat Patrick Bernau in einem lesenswerten Essay [🔗](#) in der FAS, in dem er feststellt, dass wir in Europa seit der Jahrtausendwende beim Produktivitätsfortschritt zurückgefallen sind. Er befürchtet, dass wir durch die Stagnation allmählich zum Wohlstandsmuseum geraten **1**. Eine Studie der EU-Kommission zu digitalen Kompetenzen [🔗](#) sieht ähnlich düster aus. In der Digitalisierung liegt

Deutschland weit zurück und weist zudem bei den grundlegenden Technologien extrem starke Abhängigkeiten vor allem von Plattformen aus den USA auf.

Es könnte schlimmer kommen: Wenn der Mann mit dem Machogehabe die US-Präsidentschaftswahlen im November gewinnt, könnte seine Nähe zu anderen autokratischen Herrschern unmittelbare Folgen für die Sicherheit insbesondere der osteuropäischen Staaten haben. Aber können die Europäer dann überhaupt ihre eigene Sicherheit verteidigen – politisch,

militärisch, technologisch, wirtschaftlich? Faktisch funktioniert noch nicht einmal die Einsatzplanung der frisch beschafften F35-Kampfflugzeuge ohne das Zutun der AWS Cloud Plattform, und ohne das Wohlwollen der Plattformbetreiber aus den USA oder der Volksrepublik China kann unsere Industrie benötigte Güter überhaupt nicht produzieren. Nur allzu gern würde man dieses Horrorszenerario als unrealistisch abtun.

Doch auch ohne solche düsteren Aussichten gibt es genug gute Gründe, für grundlegende Digitaltechnologie eigene Kompetenz und Innovationsfähigkeit aufzubauen oder wiederzuerlangen. Das betrifft nicht nur die Einhaltung unserer eigenen Regeln und Gesetze (etwa beim Datenschutz) und hat strategische Hintergründe (Risikomanagement). Wir wollen schlicht an der Wertschöpfung der immer wichtigeren digitalen Plattformen teilhaben können und dort durch einen funktionierenden Markt ohne Oligopole

auch die entsprechende Kosteneffizienz erzielen. Lassen Sie uns also Lust zur Digitalisierung entwickeln und aus dem Wohlstandsmuseum ausziehen!

Im Bereich grundlegender digitaler Hardware (Chiptechnologie) haben die EU und der deutsche Staat große Förderprogramme aufgelegt. Dabei handelt es sich um ein ambitioniertes Projekt, das wohl mehr als ein Jahrzehnt in Anspruch nehmen wird und zig Milliarden Euro an Investitionen erfordert.

Auch auf einem anderen Gebiet sieht es in Sachen Abhängigkeit nicht besser aus: Die moderne, automatisierte Bereitstellung von virtuellen IT-Umgebungen durch Cloud- und Container-Technologie hat einen enormen Qualitäts- und Produktivitätsschub bei der Entwicklung und Bereitstellung von Softwarelösungen ermöglicht. Doch die Plattformen dafür liegen in den Händen weniger großer Hightech-Unternehmen aus den USA. Wollen wir als europäische Gesellschaften die Freiheit haben, die Regeln dafür festzulegen und durchzusetzen, ohne vom digitalen Fortschritt abgeschnitten zu werden, benötigen wir Technologie unter eigener Kontrolle. Ansonsten wird wohl das Spiel weitergehen, dass die EU-Kommission die Regeln wirtschaftsfreundlich ignoriert und Max Schrems [☞](#) anschließend mit Hilfe des Europäischen Gerichtshofs (EuGH) an das geltende Recht erinnern muss.

Auch hier ist die Abhilfe ein langwieriger Prozess, weil ein ganzes Ökosystem auf den Plattformen aufbaut und Europa sich in dessen Abhängigkeiten begeben hat. Es gibt aber auch eine gute Nachricht: Die Basistechnologie ist als Open-Source-Software vorhanden und vielfältig (wenn auch fragmentiert) im Einsatz. Sie lässt sich ohne Milliardeninvestitionen zu einer konkurrenzfähigen offenen Plattform ausbauen.

Das konkret umzusetzen, ist das Ziel des Projekts Sovereign Cloud Stack (SCS). Die Idee entstand Ende 2019 und war als Erweiterung für die damals noch neue Gaia-X-Initiative gedacht. Nach einem erfolgreich abgearbeiteten Evaluierungsauftrag von der Bundesagentur für Sprunginnovationen (SPRIND) wurde die Idee als Projekt der Open Source Business Alliance e.V. (OSBA) aufgebaut, vom Bundesministerium für Wirtschaft



© Pixabay

1 Deutschland als Technologiemuseum? Einige Experten befürchten, dass es so weit kommen kann, wenn wir nicht rechtzeitig gegensteuern.

und Klimaschutz (BMWK) mit einer Förderung in Höhe einer niedrigen zweistelligen Millionensumme finanziert. Das Projekt wird von einem kleinen Projektteam bei der OSBA gesteuert und mithilfe von vergebenen Entwicklungsaufträgen ausgeführt. Der Autor, technischer Leiter des Projektteams, beschreibt das Projekt im Folgenden übersichtsartig aus der Insider-Perspektive.

Digital souverän?

Als SCS seinen Namen wählte, war digitale Souveränität noch nicht im Munde aller Marketing-Abteilungen. Vermutlich muss man es als Erfolg werten, wenn jetzt jedermann digitale Souveränität für sich in Anspruch nimmt. Gleichzeitig wird der Begriff jedoch verwässert und bis zur Unkenntlichkeit weichgespült, so wie zuvor „Green“ und „Open“. Möglicherweise wäre daher die von der OSBA im Jahr 2021 eingetragene Marke „Sovereign Cloud Stack“ so heute gar nicht mehr eintragungsfähig.

Bei Souveränität geht es darum, dass man Abhängigkeiten nicht eingehen muss, sondern bewusst wählen kann, wo man sie vermeidet und wo man sie für unproblematisch hält. Auf diese Art erhält man die Gestaltungsfreiheit in der immer wichtiger werdenden digitalen Welt. Diese Definition wurde so bereits auf dem Digitalgipfel 2018 festgelegt.

Das SCS-Team hat ein Stufenmodell entwickelt, das es Nutzern zu beurteilen erlaubt, inwieweit ihnen eine Plattform tatsächlich ermöglicht, souverän zu sein **2**. Es wurde in „The Cloud Report“ [☞](#) und der Zeitschrift „Datenschutz und Datensicherheit“ [☞](#) veröffentlicht.

Grundlegend (Stufe 1) ist dabei die Einhaltung der Datenschutzbestimmungen (GDPR/DSGVO): Wer personenbezogene Daten verarbeitet, kann das nicht ohne Weiteres in den Public Clouds von Anbietern außerhalb der Gültigkeit der europäischen Datenschutzbestimmungen tun. Die Angemessenheitsbeschlüsse der EU-Kommission mögen diese Tatsache bis zum nächsten Schrems-Urteil des EuGH kurzfristig verschleiern, sie sind allerdings nicht mehr als ein allzu offensichtlich erkennbares Feigenblatt.

Auf der Stufe 2 steht die Möglichkeit, zwischen verschiedenen Cloud-Betreibern zu wählen und gegebenenfalls selbst eine Plattform aufbauen und betreiben zu können. Diese Wahlmöglichkeit muss auch dann noch weiter bestehen bleiben, wenn man einmal gewählt und den Workload mitsamt Automatisierung einmal auf einen Anbieter hin optimiert hat. Das setzt anbieterübergreifend definierte Schnittstellen und ein ebensolches Systemverhalten voraus, eröffnet aber eine weitere Option: mehrere Anbieter zu verbinden und als große föderierte Cloud zu nutzen.

Stufen Digitaler Souveränität	SCS Zertifizierungsstufen
4: Transparenz und Kompetenz über den Betrieb vorhanden	4: "SCS-sovereign" – Offenheit von Betriebswerkzeugen und Nutzerverwaltung, Monitoring – Open Operations
3: Technologische Transparenz, Gestaltungs-/Innovationsfähigkeit	3: "SCS-open" – SBOM der funktionalen Software vorhanden und komplett offen gemäß der Four Opens
2: Freie Wahl zwischen kompatiblen Anbietern, insourcing Option	2: "SCS-compatible" – Technisch kompatibel (mit automatisierten Konformitätstests von CNCF, OIF, SCS)
1: Umsetzung rechtlicher Anforderungen (DSGVO)	1: Keine SCS eigene - Rückgriff auf ENISA / Gaia-X Labels / EU-CS / BSI
0: Keine	

© SCS

2 Das Stufen-Modell für digitale Souveränität und zugehörige SCS-Zertifizierungen.

Stufe 3 bezeichnet dann die Möglichkeit, die Plattform nach eigenen Wünschen zu gestalten. Wie funktioniert die Technologie genau, welche Konfigurationsmöglichkeiten gibt es, und lässt sich der Quellcode ändern, um neue Möglichkeiten zu eröffnen? Warum wurde eine Technologieentscheidung getroffen? Kann man erfolgreich Änderungen vorschlagen und einbringen? Das bietet nur eine Plattform, die vollständig und dauerhaft Open-Source-Code benutzt, den mehrere Parteien in einem offenen Prozess entwickeln. Das SCS-Team setzt hier auf die Four Opens  der Open Infra Foundation und hat eine Checkliste für Open Source Health entwickelt .

Das hochgradig zuverlässige Betreiben von Cloud-Infrastruktur stellt eine große Herausforderung dar. Dafür gilt es, Wissen und Kompetenz aufzubauen und die

richtigen Werkzeuge und Prozesse aufzusetzen. Bei der Stufe 4 geht es also um Transparenz für die betrieblichen Aspekte der Plattform. Das ermöglicht es, Wissen weiterzugeben, sodass der Wissensaufbau anbieterübergreifend gelingt. Die Open Operations Initiative, vom SCS-Team maßgeblich mitinitiiert, hat sich eben das zum Ziel gesetzt.

Mit diesem Modell kann man nun sogenannte souveräne Cloud-Angebote bewerten. Hier gibt es einmal die Ableger der großen amerikanischen Anbieter, die technologisch den Originalen entsprechen, bei denen aber der Betrieb von einem europäischen Partner (Delos, T-Systems) oder einem europäischen Ableger (Amazon, Oracle) übernommen wird. US-Behörden sollen in diesen Konstrukten technisch keinen Zugriff auf die Kundendaten haben, der Cloud Act gilt

für die Partner nicht. Im Fall der europäischen Ableger müssten diese faktisch so unabhängig von der Mutter sein, dass keine Weisungsbefugnis besteht und somit der Cloud Act (und wichtiger die FISA Orders) nicht durchgreifen. Beides – sowohl die Unmöglichkeit des technischen Zugriffs als auch das Nichtgreifen des Cloud Act – muss man sich sehr genau ansehen. Wenn hier alles sauber ist, würden diese Angebote tatsächlich Stufe 1 erreichen und somit eine rechtskonforme Verarbeitung von personenbezogenen Daten ermöglichen.

Eine Wahlfreiheit hat man allerdings mit keinem dieser Angebote: Es gibt jeweils nur einen Anbieter (plus gegebenenfalls einen Partner in Europa), der Technologie, verfügbare Features, Preise und Nutzungsbedingungen vorgibt. Sobald man sich darauf eingelassen hat, gelingt ein Wechsel nur noch mit großem technischem Aufwand. Eine Sonderstellung könnte VMware einnehmen: Die Technologie gibt es bei mehreren Anbietern, und man kann sie auch selbst betreiben. Hier wäre Stufe 2 erreichbar, die Wahlfreiheit. Allerdings gewinnt man den Eindruck, dass der neue Eigentümer Broadcom genau das gar nicht mehr will und lieber auf schnelles Wachstum mit wenigen großen Partnern setzt.

Die europäischen Betreiber von SCS-Plattformen unterliegen selbstverständlich den europäischen Datenschutzbestimmungen. Hier kann man sich an den Regeln der European Union Agency for Cybersecurity (ENISA) oder auch an den damit verwandten Labeln von Gaia-X orientieren. Für die Wahlfreiheit (Stufe 2) sind die SCS-Standards zuständig. Dabei handelt es sich um technische Standards, die Schnittstellen und Systemverhalten definieren und durch automatisierte Tests überprüft werden. Sie gewährleisten einen hohen Grad an Kompatibilität. SCS-Plattformen mit „SCS-compatible“-Zertifizierung bieten somit echte Wahlfreiheit. Dank der Föderierbarkeit ergibt sich für den Nutzer auch die Möglichkeit, mehrere Anbieter zu einer großen virtuellen Cloud zusammenzuschalten.

Die SCS-Software (Referenzimplementierung) implementiert sämtliche SCS-Standards. Sie ist eine vollständige Cloud- und Container-Plattform, anders als manch andere Referenzimplementie-



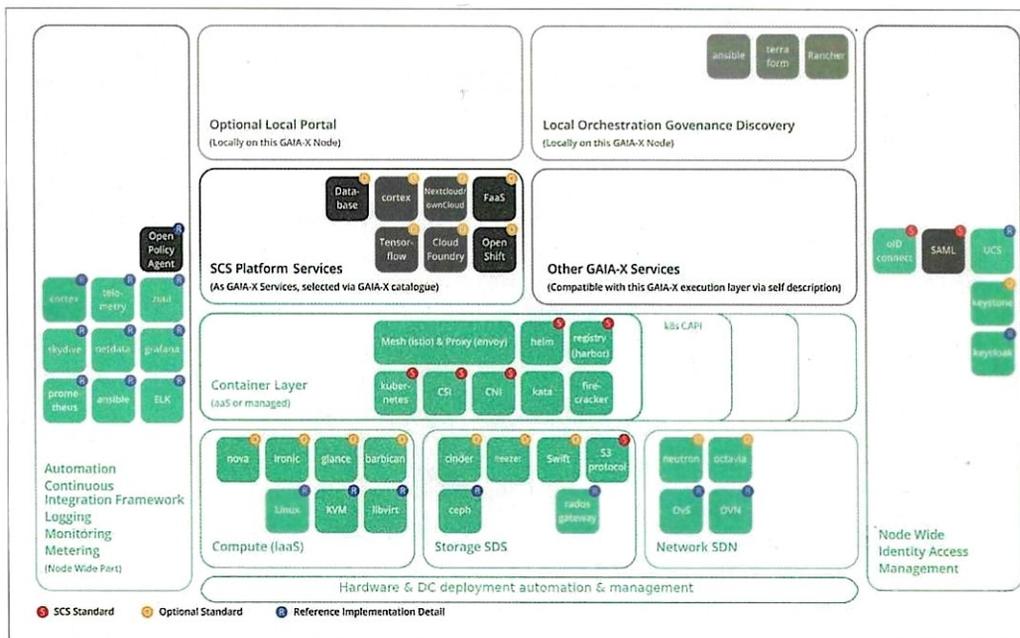
© SCS

3 Die Mission von SCS ruht auf den drei Säulen: Standards, Software, Wissen.

zung mit dem Anspruch, produktionsreife Software zu sein. Sie wird völlig nach den Prinzipien der Four Opens entwickelt: Open Source, Open Design (Decisions), Open Development, Open (and diverse) Community. Sie findet in enger Zusammenarbeit mit den Up-stream-Communities statt: Nutzer können sich auf vielfältige Weise einbringen und die Technologie mitgestalten. Anbieter, die die Funktionalität durch die SCS-Software erbringen oder gleichermaßen offen entwickelte Software einsetzen, können das durch die SCS-open-Zertifizierung nachweisen. Analog werden eine Offenheit bei den Betriebswerkzeugen sowie der Wissensaufbau und die Transparenz bei den Betriebsprozessen (Open Operations) durch die SCS-Sovereign-Zertifizierung nachweisbar sein **3**.

Architektur

Agile Entwicklungsmethoden haben in den vergangenen 20 Jahren einen Produktivitätssprung in der Softwareentwicklung ermöglicht. Neben der passenden Kultur ist eine agile Infrastruktur wichtig, um die Vorteile realisieren zu können: Eine Plattform, die es den Entwicklungsteams (oder besser DevOps- oder DevSecOps-Teams) ermöglicht, alle Schritte inklusive Deployment sowie Betrieb zu automatisieren und all die Integrationsschritte kontinuierlich und früh im Entwicklungsprozess auszuführen. Bei guter Testabdeckung erzielt man damit nicht nur eine höhere Geschwindigkeit, sondern zugleich auch bessere Qualität. Das Empowerment der Teams bei der Nutzung von Self-Service-Infrastruktur bringt eine ganz andere Motivation mit sich als das ewige Warten auf Genehmigungen von zu beantragenden, von Change Boards zu genehmigenden und am Ende manuell bereitstellenden Ressourcen.



4 Die Schichten von SCS mit IaaS und Container-Schicht sowie Betriebswerkzeugen (links) und IAM (rechts). Die dunklen Kacheln zeigen Ergänzungsmöglichkeiten, die SCS selbst (noch) nicht umsetzt.

Das ist freilich das Kernversprechen von Cloud-Umgebungen – echten Clouds, wohlgemerkt: Manchmal werden Virtualisierungsplattformen ja ohne alle Cloud-Attribute fälschlicherweise als Private Cloud vermarktet. Auf solchen Cloud-Umgebungen kann man Anwendungen modular, skalierbar und hochautomatisiert entwickeln, also sogenannte Cloud-Native-Anwendungen erstellen. Mittler-

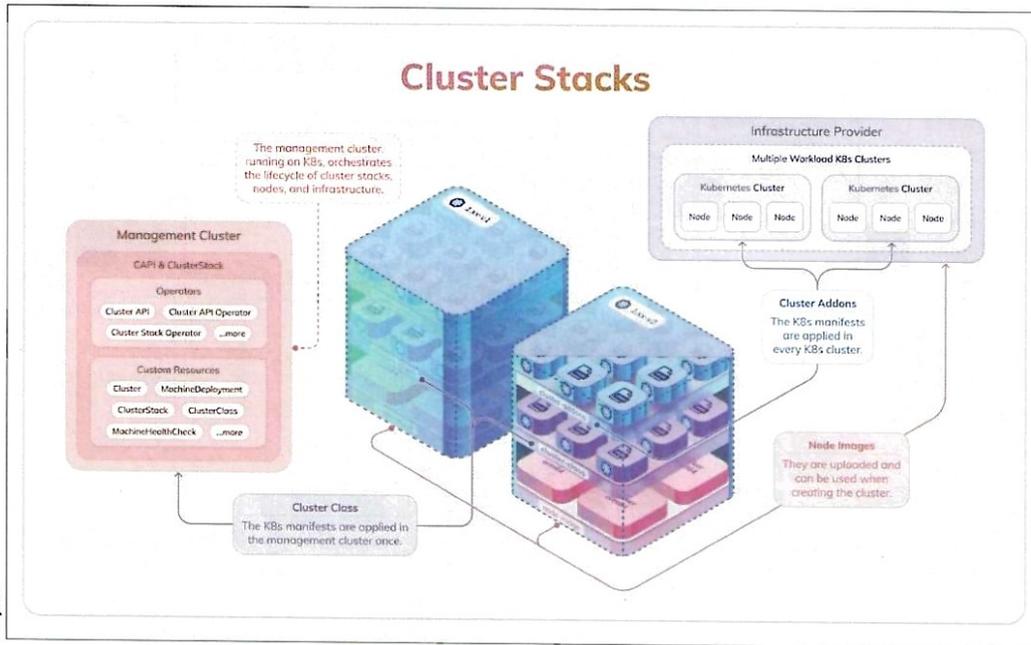
weile werden neue Cloud-Native-Workloads meistens als Container bereitgestellt. Zur Container-Orchestrierung hat sich in der IT-Industrie weitgehend Kubernetes durchgesetzt, sodass letztlich K8s die wichtigste Schnittstelle für die Anwendungsentwicklungsteams respektive DevOps-Teams darstellt.

Somit ist ein Fokus auf die Bereitstellung von Kubernetes das wichtigste

Vergabebeispiel: SCS-Container-Schicht

Mittels Cluster-API (CAPI) lassen sich komplette K8s-Cluster als Kubernetes-Objekte mittels Custom Resources verwalten. Das funktioniert auf vielerlei Infrastrukturschichten. Dafür gibt es dann jeweils einen Provider, zum Beispiel den Cluster-API-Provider für OpenStack (CAPO). Um das alles nutzbar zu machen, benötigt man jedoch noch einiges Drumherum: den Cloud Controller Manager (CCM), Netzwerkintegration (CNI), Storage Integration (CSI), Node-Images und Automatisierung für all diese Komponenten. Das hat das Projektteam (aufbauend auf Vorarbeit von B1 Systems) für OpenStack mithilfe einiger Skripte umgesetzt, als Version 1 für ein SCS-KaaS. Im Vergabepaket 5 wurde ausgeschrieben, das Ganze auf Cloud-Native-Manier umzusetzen und auch

allgemeiner und einfacher einsetzbar zu machen. Diese Ausschreibung hat die Syself GmbH gewonnen, die seitdem eng mit der SCS-Community daran arbeitet, die Version für das SCS-KaaS zu entwickeln. Das Ergebnis sind Cluster Stacks **5**, die einen Operator mitbringen, der die ganze Automatisierung durch entsprechende K8s-Objekte steuert und der auch die Node Images managt. Die Konfiguration dafür bezieht der Anwender idealerweise aus einem Git-Repository (GitOps). Das SCS-Projekt hat zum gegenwärtigen Zeitpunkt Cluster Stacks für OpenStack und für Docker implementiert. Die Architektur steht aber auch für weitere Umgebungen offen, und SCS und Syself freuen sich über weitere unterstützte Infrastruktur.



5 Cluster Stacks – eine SCS-Spezialität-- verwalten die Kubernetes-Cluster mittels Cluster-API und bringen die Node Images sowie die Integration als Cluster-Addons mit.

technische Ziel von SCS. Das Aufsetzen und Betreiben eines K8s-Clusters ist zur Erreichung der SCS-Ziele nicht genug: Die K8s-Cluster müssen wohldefinierte Eigenschaften haben, sodass der Betreiberwechsel problemlos gelingt. Ein großer zentraler Cluster erfüllt in den meisten Fällen nicht die Sicherheitsanforderungen der Anwender. Die Isolation durch die leichtgewichtigen Container hat viele Vorteile. Die bietet allerdings nicht das Maß an Isolation, das für die Absicherung notwendig ist, wenn man mit potenziell böswilligen Nutzern zu rechnen hat. Damit eignet sie sich ohne weitere Maßnahmen nicht für den harten Alltag in einer Public Cloud.

Das SCS-Projekt will privaten und Public Clouds gleichermaßen als Basis dienen und hat deshalb ein paar grundlegende Architekturentscheidungen getroffen. So stellt das Projekt die Möglichkeit bereit, einfach eigene K8s-Cluster zu erzeugen, zu verwalten (vergrößern, verkleinern, auf eine neue K8s-Version aktualisieren) und wieder zu entfernen. Das heißt bei SCS dann KaaS – Kubernetes-as-a-Service. Diese Cluster gehören exklusiv dem Kunden, der Plattformbetreiber kann beim Betrieb unterstützen. Mehr Details dazu finden Sie im Artikel zu KaaS in diesem Schwerpunkt.

Zur Bereitstellung von Clustern dient das anbieterübergreifende CNCF-Projekt Cluster-API. Es unterstützt zahlreiche Cloud-Technologien als darunterliegende Plattformen; außerdem ist ein Einsatz auf Bare Metal möglich.

Cloud-Plattformen erleichtern das Verwalten von K8s-Clustern, weshalb die überragenden Mehrheit der Nutzer zu dieser Variante greift. Darum setzen auch in der Referenzimplementierung von SCS die Kubernetes-Cluster standardmäßig auf einer souveränen Cloud-Technologie (IaaS, Infrastructure-as-a-Service) auf.

Für die IaaS-Plattform kommt Open-Stack zum Einsatz. Als einzige bewährte Plattform erfüllt es alle Anforderungen an Offenheit und hat in den Kernkomponenten einen hohen Reifegrad sowie eine beachtliche Verbreitung erreicht. Dies steht in Einklang mit Marktstudien über europäische Cloud-Anbieter.

Die IaaS- und die KaaS-Plattform lassen sich unter voller Einhaltung der Standards auch einzeln nutzen. Das SCS-Team favorisiert allerdings die Verwendung beider Schichten, da dadurch die größte Souveränität und Synergie in der Zusammenarbeit entsteht.

Neben den Schichten IaaS und KaaS erfordert das Aufsetzen und Betreiben einer Plattform weitere Software. Dazu

gehören Betriebswerkzeuge (Lifecycle-Management, Monitoring, Logging, Auditing, Metering, CI, ...) sowie das Nutzer- und Berechtigungsmanagement (IAM, Identity and Access Management). Sie sollen mit derselben Priorität entwickelt werden und nach Möglichkeit beide Schichten unterstützen können. Einen Überblick über die Architektur gibt Abbildung 4.

Damit bleiben einige Dinge bewusst außen vor. Die Entscheidung zugunsten von Cluster-API war umstritten, weil mit Gardener eine ziemlich ausgereifte Open-Source-Alternative existiert. Sie hängt allerdings komplett von den Zielen des Unternehmens SAP ab. Bislang waren weder

Versuche von Erfolg gekrönt, eine technologische Konvergenz mit Cluster-API zu organisieren, noch Ideen, Gardener unabhängiger von SAP zu strukturieren. Die Offenheit gibt es weiter, allerdings hängt in der SCS-Referenzimplementierung mittlerweile einiges von Cluster-API ab. Zumindest eine Kompatibilität bezüglich der Standards bleibt aber ohne große Umstände möglich.

Die Lösung enthält keine höheren Plattformdienste wie zum Beispiel Datenbanken, E-Mail, Messaging, Big Data oder AI. Dafür ist das SCS-Projekt zu klein. Andererseits sollte die IaaS-plus-KaaS-Lösung jedoch eine hervorragende Plattform für Open-Source-PaaS-Lösungen bieten – an einzelnen Beispielen wie Owncloud und Nextcloud wurde das auch bereits demonstriert.

Die Vision: Drittanbieter haben damit die Möglichkeit, Plattformdienste anzubieten und gut zu integrieren, inklusive automatischem Deployment, Monitoring, Nutzermanagement und Abrechnung. Sobald sich hier Lösungen als Standards durchsetzen, kann SCS mit den Partnern arbeiten und höherwertige Standards etablieren. Ideen gibt es dazu mit dem Eclipse-Projekt Xpanse und den Entwicklern von GlassCube.

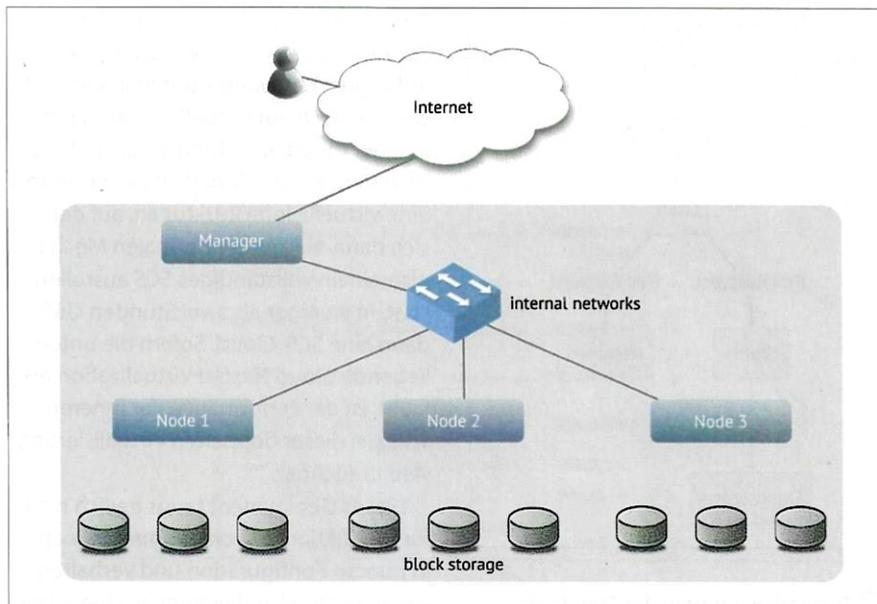
Arbeitspakete

All die Komponenten aus den Schichten und Werkzeugen müssen konsistent konfiguriert und gepflegt werden, so dass daraus ein Produkt entsteht, das sich wie aus einem Guss anfühlt. Für die Pflege, Weiterentwicklung und die Integration fanden rund um die einzelnen Technologieschwerpunkte Ausschreibungen statt, die Unternehmen dazu einluden, Konzepte dafür zu entwickeln und kompetent Arbeitskraft in das Projekt einzubringen. Für die großen Blöcke IaaS, KaaS, IAM und OpsTooling gingen dazu Vergaben an die OSISM GmbH, Sysself GmbH, Univention GmbH sowie (für die Betriebs-Tools) an die Gonicus GmbH und an die Firma dNation s.r.o.

Daneben wurden spezifische Bereiche identifiziert, die eine spezielle Weiterentwicklung gemäß den Zielen des Projekts erfordern. So soll die Netzwerkschicht (auf Basis von OVN) skalierbarer und sicherer werden und auch Verbindungen zwischen Clouds ermöglichen. Überdies ist geplant, die Mandantentrennung und Verschlüsselung in der Storage-Lösung zu verbessern, dasselbe gilt für die Komponenten zur Integration von Cluster-API und der darunterliegenden OpenStack-Schicht. Außerdem gilt es, die Sicherheitsarchitektur zu analysieren, Angriffstests vorzunehmen, die Nutzungsdatensammlung und Systemüberwachung zu optimieren sowie Standards mit automatisierten Tests zu entwickeln. Mittlerweile gibt es ein Testlabor, in dem sich Tests auf echter Hardware unter produktionsähnlichen Bedingungen abbilden lassen. Entsprechende Ausschreibungen wurden an B1 Systems, Cloud&Heat, Secustack, Proventa und JH Computers vergeben. Einen Überblick darüber finden Sie auf der SCS-Webseite [🔗](#).

CI und Cloud-in-a-Box

Die SCS-Entwicklung findet in virtuellen Teams statt, die wöchentlich per Videokonferenz zu Team-Meetings zusammenkommen, um die Fortschritte abzustimmen und Abhängigkeiten aufzulösen. Unter Leitung des zuständigen Product Owners (PO, ein Mitglied des SCS-Projektteams) werden die Themen vorbereitet; die Entscheidungen fallen in den



6 Das SCS-Testbed mit drei hyperkonvergenten Knoten und einem Manager.

Team-Meetings typischerweise im Konsens. Alle Termine sind öffentlich und werden im Community Calendar veröffentlicht. Für teamübergreifende Themen gibt es das Produkt-Board, in dem neben den POs weitere Community-Mitglieder sitzen, die beispielsweise als Leiter von Special Interest Groups (SIGs) Verantwortung übernommen haben.

An der Terminologie kann man schon erahnen: Der Prozess ist an Scrum angelehnt, und es finden zweiwöchige Sprints statt, in denen User Stories bearbeitet

werden. Sie gehören typischerweise zu größeren Epics und speisen sich aus dem Product Backlog. Teamweite Dailies gibt es nicht, wohl aber Backlog-Refinements in kleineren Gruppen und kurzfristige Touchpoints. Videokonferenzen für Absprachen oder Hacking-Sessions können jederzeit auf dem SCS-eigenen Jitsi stattfinden. Bei der Entwicklung kommt intensiv Github zum Einsatz, um Code dort einzupflegen, zu diskutieren und zu mergen. Viel Entwicklungsarbeit fließt in die Upstream-Open-Source-Projekte zurück.

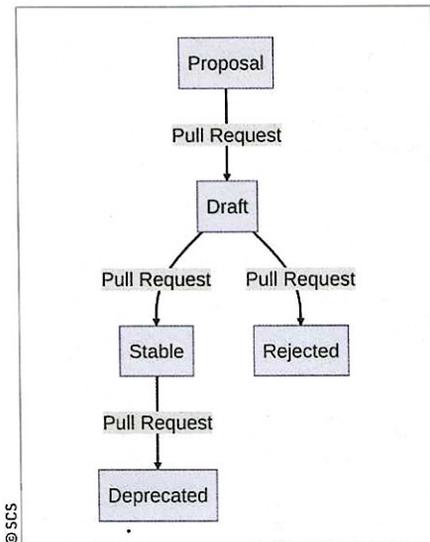
Vergabebeispiel: Föderierung

In einem Zielszenario, in dem Betreiber mit mehreren SCS-Cloud-Anbietern arbeiten und sie als große virtuelle Cloud nutzen, wollen diese die Nutzer und deren Berechtigungen nicht auf jeder Cloud einzeln verwalten. In OpenStack ist die Keystone-Komponente mittels Open ID Connect (OIDC) auf Föderierung eingerichtet, K8s kann von Haus aus OIDC. Allerdings ist das alles nicht ganz so einfach.

Im IAM-Vergabepaket von SCS arbeitet deshalb Univention daran, die Föderierbarkeit zu verbessern. Um Nutzerföderierung über die Schichten hinweg zu ermöglichen, wurde in SCS eine Keycloak-Komponente integriert, die über OIDC als Identity Provider und als Broker dienen kann. Dafür musste der De-

vice-Authorization-Grant-Workflow in Keystone ans Laufen gebracht werden, den das SCS-Projekt erfolgreich in das OpenStack-Projekt einbrachte.

Kunden von SCS-Betreibern sollen als Verwalter von OpenStack-Domänen innerhalb dieser Nutzer, Berechtigungen und Projekte selbst verwalten können. Auch das hat Beiträge zum OpenStack-Projekt notwendig gemacht, damit die Manager-Rolle für Domänen in OpenStack funktioniert. Die Verbesserungen kommen somit OpenStack-Betreibern auch unabhängig von SCS zugute und verringern die Divergenz zwischen diesen. Nach den Baustellen auf der IaaS-Schicht ist nun noch die Nutzerverwaltung auf der Container-Schicht im Visier.



7 Zustandsdiagramm der Standards.

Das SCS-Team sucht die Abstimmung mit den Upstream-Communities und trägt Code auch bevorzugt dort bei.

Die Protokolle der Meetings lagern ebenfalls in Github. Sie entstehen live und kollaborativ während der Termine in Hedgedoc, einem kollaborativen Echtzeiteditor mit Markdown-Rendering. Daneben steht eine Nextcloud zur Verfügung, auf der knapp 200 Mitglieder flexibel Dateien, Kontakte, Umfragen und vieles mehr bearbeiten und teilen können. Die Mitglieder stehen automatisch auch auf Mail-Verteilern, daneben findet viel Kollaboration in den Matrix-Chaträumen von SCS statt.

SCS liefert eine vollständige eigene Cloud. Die OSISM-Automatisierung nutzt Ansible-Playbooks (das OSISM-Framework verwendet Kolla-Ansible), um automatisiert Infrastruktur-Container und die Open-Stack-Container auf der Hardware zu installieren sowie die Dienste zu starten und zu verwalten. Auch wenn sich damit der Installationsprozess (und insbesondere auch das Upgrade) auf Hardware hochgradig automatisieren lässt, so ist er doch für Entwickler typischerweise nicht zugänglich: Die meisten haben kein kleines Rechenzentrum für Testzwecke im Keller. Während sich Anwendungen für SCS also ganz wunderbar auf existierenden SCS-Umgebungen testen lassen, haben es die SCS-Developer selbst insbesondere bei der Infrastrukturschicht deutlich schwerer.

Die OSISM-Entwickler haben dafür einen Trick auf Lager: Anstelle einer Installation auf echter Hardware kann man die Dienste auch auf virtuellen Maschinen in einer Cloud ausführen. Dazu legt man mittels OpenTofu (früher: Terraform) eine virtuelle Infrastruktur an, auf der sich dann mittels der normalen Mechanismen ein vollständiges SCS ausrollen lässt. In weniger als zwei Stunden läuft dann eine SCS-Cloud. Sofern die unten liegende Cloud Nested Virtualization erlaubt, ist die Performance der inneren VMs bei dieser doppelten Virtualisierung sehr brauchbar.

Dieses Deployment taugt freilich nicht für Produktionszwecke, kommt jedoch in puncto Konfiguration und Verhalten einem echten Deployment aus logischer Sicht ziemlich nahe. Es eignet sich somit dazu, Entwicklern als Test- oder Demo-Umgebung zu dienen oder Betriebspersonal auszubilden. Dieses Testbed **6** ist derzeit für OpenStack-Clouds implementiert, was den Vorteil hat, dass SCS ein SCS-Testbed hosten kann. Eine Umsetzung auf anderen Clouds (etwa AWS oder Azure) wäre möglich, wurde bislang aber weder vom Projektteam priorisiert noch von einem Mitglied der SCS-Community aus eigenem Interesse realisiert.

Dieses Testbed wird täglich installiert und automatisiert durchgetestet. Der aktuelle Codestand kann damit niemals länger als einen Tag unbemerkt defekt sein. Neben einer Testbed-Neuinstallation findet auch täglich ein Upgrade-Test statt, bei dem der Aktualisierungsprozess vom letzten Release auf den aktuellen Codestand durchgespielt wird. Die Automatisierung erfolgt mittels des sehr flexiblen CI-Frameworks Zuul.

Manchmal ist es praktischer oder sogar notwendig, auf realer Hardware zu testen. Zu diesem Zweck gibt es eine weitere Installationsvariante, die Cloud-in-a-Box. Dort wird ein speziell konfiguriertes Deployment-Profil auf einer einzigen Workstation installiert. Als Single-Node-Deployment unterscheidet es sich stärker von echten Installationen als ein Testbed. Dafür ist man hier für ein paar Tausend Euro Hardwareinvestition Herr über eine eigene kleine Cloud ohne doppelte Virtualisierung. Das lässt sich durchaus auch als Option für echte, produktionsreife Edge-Clouds ausbauen.

Weniger Aufwand verursacht das Testen der K8s-Schicht. Sie setzt ja im Normalfall auf existierende Infrastruktur auf. Man kann bequem in weniger als einer Viertelstunde einen Kubernetes-Cluster anlegen und diesen dann auf Herz und Nieren testen. Bei der Automatisierung in der V1 der KaaS-Implementierung war man auf OpenStack als darunterliegende Infrastrukturschicht beschränkt. Cluster-API kann aber mehr, es verwaltet Cluster auf VMware, AWS, OpenStack, Azure und vielen weiteren Umgebungen. In der V2 der SCS-KaaS (Cluster-Stacks) wurde das auch umgesetzt: Aus dem SCS-Projekt kommt Unterstützung für OpenStack und lokale Deployments in Docker, für weitere Integration sucht man die Zusammenarbeit mit Partnern.

Das Ausrollen von Clustern lässt sich wunderbar in das vorhandene Zuul einhängen. Als Standardtest dienen im SCS-Projekt üblicherweise die CNCF-E2E-Tests, die mittels Sonobuoy laufen. Das dauert allerdings rund zwei Stunden. Das schnelle Ausrollen der Cluster kann also die Cycle Time dennoch nicht unter ein paar Stunden drücken, wenn man das volle Testprogramm fahren will. Für Kurztests gibt es deshalb auch noch eine abgekürzte Version.

Standards und Releases

Konzepte genügen nicht: Es erfordert konkret entwickelte und nutzbare Software, um Cloud-Betreibern und letztlich deren Nutzern ein Angebot zu machen, das sie annehmen können und damit tatsächlich Fortschritte bei ihrer digitalen Souveränität machen. Die Nutzung durch und das Feedback von echten Anwendern ist auch Grundvoraussetzung dafür, dass Software reift und die Bedürfnisse und Ansprüche der Nutzer erfüllt. Was nach einer Binsenwahrheit klingt, wird allerdings dennoch oft übersehen.

Im SCS-Projekt wurden von Anfang an Software-Releases eingeführt. Obgleich die CI grundsätzlich dafür aufgesetzt wurde, den Cloud-Betreibern ein minimalinvasives wöchentliches Update der Plattform zu ermöglichen, entspricht das nicht der Arbeitsweise der derzeitigen SCS-Partner. Upstream gibt es bei Ubuntu, OpenStack oder Kubernetes immer wieder Releases mit Änderungen,

die dem Betreiber oder sogar dem Nutzer Anpassungen abverlangen. Daher stellt das SCS-Projekt in regelmäßigem Rhythmus Releases bereit, die es zusätzlich noch einmal manuell testet und bei denen es wichtige Änderungen in den Release Notes dokumentiert.

Nach dem ersten Release R0, das außer der Reihe fast direkt nach dem verspäteten Projektstart im Juli 2021 erfolgte, gab es regelmäßig im September und im März jedes Jahres SCS-Releases. Seit Ende März 2024 ist der aktuelle Stand SCS R6. Jedes Release erhält bis sechs Wochen nach dem Erscheinen der Folgeversion Wartungs-Updates, die Fehler beheben und insbesondere Sicherheitslücken adressieren. Letztere sind auch im SCS-Blog [nachzulesen](#).

Parallel zu den Releases hat das Projekt Standards für die Nutzer der Cloud-Angebote erarbeitet. Diese erlauben, Workloads mit identischer Automatisierung auf unterschiedlicher SCS-Infrastruktur auszurollen, ohne viele Parameter anpassen oder – schlimmer noch – vorab die Leistungsbeschreibungen studieren zu müssen, um zu verstehen, welches Systemverhalten bei welchen Ressourcen garantiert wird.

Ein Beispiel bietet der SCS-Flavor-Standard, der nicht erst ein mühevolleres Studium der Eigenschaften von Flavors jedes Cloud-Betreibers erforderlich macht. Die Ergebnisse der Standardisierungsbemühungen lassen sich auf der Docs-Seite von SCS [nachlesen](#). Wie

Name	Description	Operator	IaaS Compliance Check	HealthMon
ge-sec	Dev environment provided for SCS & GAIA-K context	plusserver GmbH	✓	HMA
pluscloud open-prod1	Public cloud for customers	plusserver GmbH	✓	HMA
pluscloud open-prod2	Public cloud for customers	plusserver GmbH	✓	HMA
pluscloud open-prod3	Public cloud for customers	plusserver GmbH	✓	HMA
pluscloud open-prod4	Public cloud for customers	plusserver GmbH	✓	HMA
WaveStack	Public cloud for customers	noris network AG/ Wavecon GmbH	✓	HMA
REGIOcloud	Public cloud for customers	OSDM GmbH	✓	HMA
CND5	Public cloud for customers	amcodis UG	✓	HMA
soy	Community cloud for customers	soy IT Services GmbH	✓	HMA

8 Der tägliche Kompatibilitätscheck der öffentlichen SCS-Clouds vom 28. März 2024. Neben dem Kompatibilitätscheck (vorletzte Spalte) offerieren alle Anbieter auch einen OpenStack Health Monitor (letzte Spalte), der die Funktionsfähigkeit permanent überwacht.

der gesamte Code werden auch die Standards auf Github durch Pull Requests entwickelt, reviewt und dann stabilisiert **7**. Den Prozess der Standardisierung beschreibt ein eigener Standard mit der Nummer 0001. Die einzelnen Standards haben verschiedene Zustände, die in den Metadaten des jeweiligen Dokuments vermerkt sind. Über die Pull-Requests entscheiden die zuständigen technischen Teams in einer Abstimmung.

Zertifizierungskataloge fassen die einzelnen Standards zusammen, getrennt nach den Schichten IaaS und Container

(KaaS). Sie unterliegen einem Lebenszyklus, in dem neue Anforderungen alte ablösen. In den meisten Fällen handelt es sich bei den neuen Regeln um eine Obermenge der alten, sodass sich für Nutzer daraus keine unnötigen Anpassungen ergeben. Dank der Standards können User Anwendungen, Betriebsprozesse und Automatisierung einmal entwickeln und dann auf allen SCS-kompatiblen Clouds installieren und betreiben. Man kann also einfach umziehen oder auch föderiert auf mehreren arbeiten – auch in der eigenen, privaten SCS-Cloud, wenn man das will.

INODES, INIT & IDEEN

LINUX UND OPEN SOURCE
topaktuell mit unserem Newsletter

www.linux-magazin.de/subscribe

Die Standards werden Hand in Hand mit der Referenzimplementierung entwickelt. Das soll sicherstellen, dass sie sich in der Praxis umsetzen lassen. Da bereits einige Cloud-Betreiber mit der Referenzimplementierung arbeiten, gibt es mittlerweile auch Rückmeldungen von Cloud-Betreibern und deren Nutzern.

So viel Nähe zur Referenzimplementierung fördert die Qualität der Standards, birgt andererseits aber die Gefahr, dass sie sich auch nur durch die Referenzimplementierung ohne Verrenkungen erfüllen lassen. Aus diesem Grund sucht das Projekt bewusst den Kontakt mit den Upstream-Communities und Projekten im Umfeld von SCS, die zwar ähnliche Technologien einsetzen, aber nicht die Referenzimplementierung. Deren Umsetzung der SCS-kompatiblen Standards ist ausdrücklich erwünscht.

Diese Bemühungen haben zu einer engen Zusammenarbeit mit dem Verein ALASCA e.V. bei der Standardisierung geführt. Er war initial nur dazu gedacht, dem Yaook-Projekt eine neutrale Heimat zu bieten, trat dann aber doch mit dem größeren Anspruch an, die digitale Souveränität für Cloud-Nutzer zu verbessern. Damit teilt der Verein sehr viele Grundlagen und Ziele mit dem SCS-Projekt, eine enge Zusammenarbeit bot sich an. Sie trägt nun Früchte: Viele Beschäftigte von ALASCA-Mitgliedsunternehmen tragen zu SCS bei, insbesondere das Arbeitspaket rund um die Standardisierung wird von Mitarbeitern von Cloud&Heat reali-

siert. Die ersten Yaook-basierten Umgebungen (die ausdrücklich den Großteil der SCS-IaaS-Referenzimplementierung OSISM nicht benutzen) sind gerade dabei, SCS-kompatible Standards auf IaaS-Ebene erfolgreich umzusetzen.

Sämtliche Standards kommen mit automatisierten Tests, die nach Möglichkeit alle wesentlichen Eigenschaften der Standards prüfen. Somit sind die Cloud-Betreiber jederzeit darüber informiert, ob sie alle Standards erfüllen und wo gegebenenfalls noch Lücken klaffen. Die meisten Tests erfordern keine besonderen Berechtigungen, sodass auch Nutzer sie jederzeit ausführen können, um die Kompatibilität zu überprüfen.

SCS-Cloud-Angebote

Ohne produktive Nutzung kann Software weder Relevanz noch Praxistauglichkeit erlangen. SCS hatte das Glück, dass neben der OSISM GmbH mit ihrer Test-Cloud Betacloud auch schnell ein Anbieter mit größeren Ambitionen auf SCS setzte: die Plusserver GmbH mit der Plus-Cloud Open (PCO). Die ersten Versionen der PCO basierten noch auf SCS vor dem R0-Release-Stand. Die Implementation wurde mit den SCS-Releases auf den jeweils aktuellen Stand gebracht.

OpenStack-Upgrades gelten gemeinhin als schwierig. Viele Anbieter schaffen es nicht, auf dem aktuellen Stand zu bleiben, und tragen auf diese Art dazu bei, den Markt hier stark zu fragmentieren.

Außer mit Eigenheiten in der Konfiguration der OpenStack-Cloud haben die Anwender auch noch mit alten Versionsständen zu tun – nicht so bei SCS.

Die Containerisierung der OpenStack-Dienste mit Kolla-Ansible, die Automatisierung durch OSISM sowie die nächsten Upgrade-Tests haben den Aktualisierungen ihren Schrecken genommen. Zum Zeitpunkt, als dieser Artikel entstanden ist (Ende März 2024), betrieb Plusserver die verschiedenen Test- und Entwicklungsplattformen bereits auf dem frisch erschienenen SCS R6 (mit Open-Stack 2023.2 „Bobcat“), und die Vorbereitungen für die Upgrades der mittlerweile vier produktiven Regionen liefen auf Hochtouren.

Es gibt mittlerweile vier Regionen und eine BSI-CS-Zertifizierung bei Plusserver, und das Wachstum der PCO ist sehr stark. Laut öffentlicher Aussagen auf der Plus-Con im Herbst 2023 liegt es deutlich über dem der weiterhin bestehenden VMware-basierten Lösung PlusCloud. Ein Vorzeigeprojekt ist die Bayern Cloud Schule (ByCS), bei der für Hunderttausende Schüler sowie deren Lehrer und Eltern eine große skalierbare Plattform benötigt wurde. Die Infrastruktur dafür setzte Plusserver mit der PCO um, was sich als Feuertau für die Skalierbarkeit von SCS ansehen lässt.

Das Versprechen von SCS, anbieterübergreifende Lösungen zu bieten, wäre verfehlt, gäbe es nur die sehr erfolgreiche PCO von Plusserver. Zwischen dem Herbst 2022 und dem März 2024 kamen der Wavestack (von Noris), die Regio-Cloud (JH/OSISM), die aov Community Cloud (AOV IT Services) und die CNDS Cloud (Artcodix) dazu. Viele weitere SCS-Clouds befinden sich im Aufbau, unter anderem bei Kyndryl, T-Systems, Dataport und ScaleUp. Daneben gibt es in der Industrie auch als Private Clouds aufgebaute SCS-Projekte. Sie setzen alle auf der SCS-Referenzimplementierung auf.

Für Public Clouds wird die Einhaltung der SCS-kompatible-Standards automatisiert überwacht und lässt sich auf den SCS-Dokumentationsseiten jederzeit einsehen [8](#). Hier dürften bald die ersten Clouds auftauchen, die zwar nicht auf der SCS-Referenzimplementierung aufbauen, aber dennoch auf der IaaS-Ebene vollständig SCS-kompatibel sind.

Standardisierungsbeispiel: Flavors

Ein Flavor in der IaaS-Implementierung ist eine Schablone für eine virtuelle Maschine (VM): Sie gibt vor, wie viele virtuelle CPUs und wie viel Arbeitsspeicher es gibt und wie groß eine optionale lokale Festplatte sein soll, von der die VM bootet. Optional kann man weitere virtuelle Hardwareeigenschaften definieren wie Grafikkarten, besondere CPU-Typen und spezielle Hypervisoren.

SCS definiert dabei auch die virtuellen Eigenschaften von CPU und Speicher: So kann sich der Nutzer bei einem Flavor mit dem CPU-Typ V sicher sein, dass er mindestens ein Fünftel eines echten CPU-Kerns erhält (oder alternativ ein Drittel eines Hyperthreads). Er weiß, dass der aktuelle Mikro-

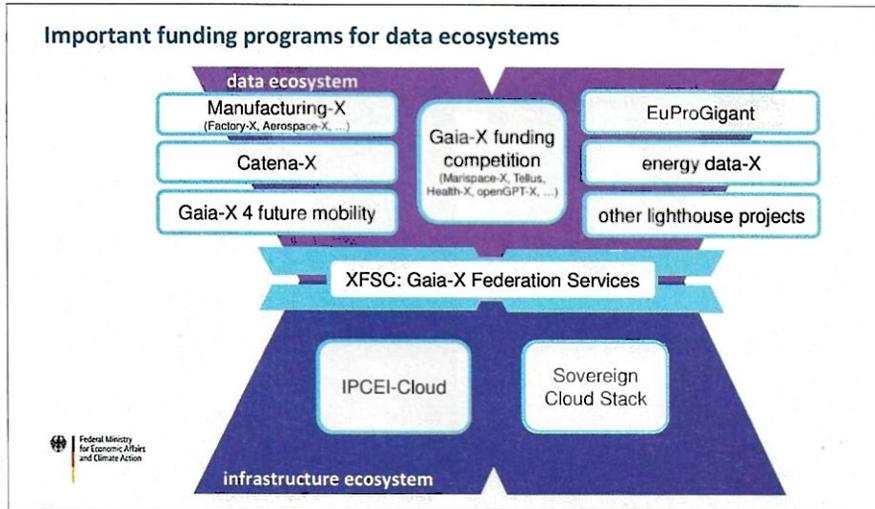
code installiert sein muss und die in Linux und KVM standardmäßig aktivierten Schutzmechanismen zur Abmilderung der CPU-Sicherheitslücken tatsächlich zum Einsatz kommen. Der zugewiesene Arbeitsspeicher wiederum darf gar nicht überbucht sein.

Es existiert im SCS-Flavor-Namensschema die Möglichkeit, Flavors zu definieren, die diese Garantien nicht geben – sie lassen sich dann jedoch klar und eindeutig an ihrem Namen erkennen. SCS arbeitet eng mit der Upstream-Community daran, solche Eigenschaften in standardisierten Attributen zu kodieren, sodass der Anwender langfristig diese Information nicht mehr unbedingt dem Namen entnehmen muss.

In der Aufstellung der Clouds mit SCS-compatible-Test sieht man auch ein Element von Open Operations: Das Monitoring der Cloud-Umgebungen mit dem OpenStack Health Monitor macht innerhalb der SCS-Community den Zustand der Clouds sichtbar. Statt eine perfekte Welt vorzutäuschen, wird der echte Zustand schonungslos transparent gemacht. Auftretende Herausforderungen werden damit für alle sichtbar; eine Aufarbeitung kann gemeinsam erfolgen, und der Lernerfolg daraus kommt allen zugute. Dass das Monitoring auf den produktiven Clouds mittlerweile sehr gute Ergebnisse zeigt, stellt eine erwünschte Nebenwirkung dar: Nur gemessene und transparent gemachte Qualität erfährt zuverlässig die Aufmerksamkeit, die sie verdient.

Die SCS-Community hat Formate wie einen Lean Operator Coffee und Open Operations Meetups für die Betreiber-Community organisiert; der Wissensaustausch dort hilft den Mitarbeitenden der Cloud-Betreiber, auf die Erfahrung einer großen Community aufzubauen, statt alle Fehler selbst machen zu müssen. Das lindert auch den Mangel an hochqualifiziertem Betriebspersonal.

Digitale Souveränität ist für die öffentliche Verwaltung zumindest auf Stufe 1 (Datenschutz) eine Grundvoraussetzung für die Nutzung digitaler Plattformen. Gleichzeitig hat die Fragmentierung der IT-Landschaft durch den Föderalismus das Entstehen von effizienzsteigernden Plattformen eher behindert und sich



9 Übersicht über einige zentrale Gaia-X/BMWK-Förderprojekte.

damit bei der Digitalisierung in der deutschen Verwaltung als wenig hilfreich erwiesen. Das höchst sinnvolle Einer-für-alle-Prinzip mit der Nachnutzung von Anwendungen durch andere Kommunen oder Länder und das gemeinsame Entwickeln auf OpenCoDE können erst dann richtig greifen, wenn es auch gleichartige Ausführungsumgebungen gibt. Der IT-Planungsrat arbeitet daran und hat hier mit der Deutschen-Verwaltungscloud-Strategie geeignete Richtlinien festgezurr. Sie referenzieren SCS als Beispiel für einen gelungenen souveränen und föderierten Ansatz.

In einem Digitalisierungswettbewerb des Landes Schleswig-Holstein wurde

SCS als Cloud-Schicht gefordert, und die WG Cloud der OSBA arbeitet mit der FITKO (Föderale IT-Kooperation) daran, die Transportabilität von Anwendungen mithilfe von SCS-Standards zu validieren. Die Anforderung, dauerhaft SCS-compatible Standards einzuhalten, gibt dem Nutzer die Sicherheit, nicht in Abhängigkeit von einem einzelnen Anbieter zu geraten. Es werden sicher einige weitere Anbieter gerade im öffentlichen Umfeld ihre Interoperabilität durch SCS-Standards sicherstellen, sei es auf der IaaS-Ebene oder zunehmend auf der Container-Ebene mit den SCS-Cluster-Stacks und dazugehörigen Standards. Das ist dann eine ideale Umgebung für

Hier finden Sie Linux-Profis in Ihrer Nähe!

IT PROFIMARKT

Finden Sie Anbieter in Ihrer Nähe



LINUX MAGAZIN

Mit DVD

DELUG-DVD

Kali Linux 2021.2

Software & Tools

CLT 2021

Griml 2021.07

Patch-Management und Penetrationstests

Rundum sich

Print, im Marktteil

linuxuser

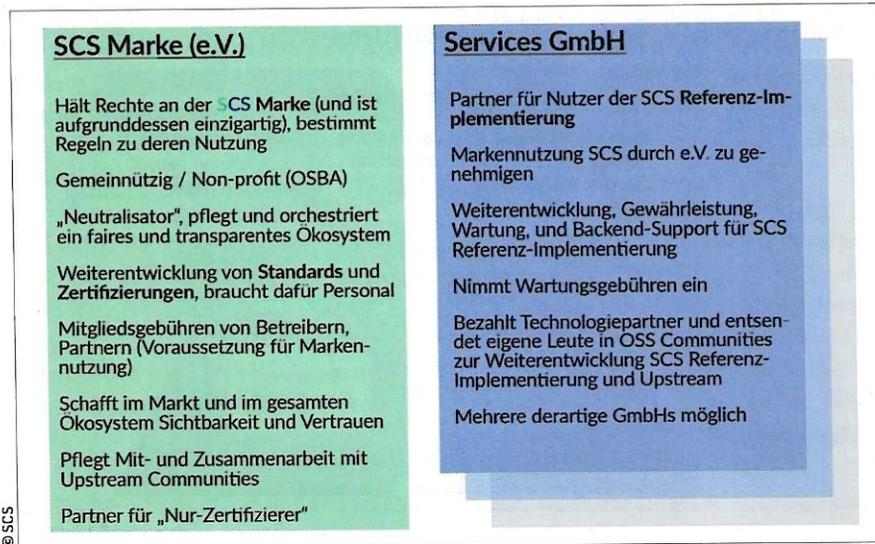
IT PROFIMARKT

Service IT-Profimarkt

Finden Sie Anbieter in Ihrer Nähe

Name	Adresse	Telefon	Web	1	2	3	4	5
IT-Profimarkt	✓	✓	✓	✓	✓
...	✓	✓	✓	✓	✓

www.it-profimarkt.de



© SCS

10 Dualität zwischen dem Verein für die SCS-Marke und deren Standards und möglichen Service-GmbHs, die für konkrete Umsetzungen Dienstleistungen anbieten.

ein Projekt wie OpenDesk, was gemeinsam mit dem ZenDiS erprobt wird. Der erste große Anbieter im öffentlichen Bereich ist das Thüringer Landesrechenzentrum, das bereits seit 2023 eine große Plattform auf Basis der SCS-Referenzimplementierung aufbaut.

In anderen europäischen Ländern gibt es ebenfalls Bedarf, wie Einladungen in die Schweiz, die Zusammenarbeit mit Cleura (Schweden), die Kontakte in die Niederlande (TNO) und die Kooperation mit dem internationalen Projekt Govstack  bestätigen.

Zukunft

Die neuen Lizenzierungsbedingungen bei VMware nach der Übernahme durch Broadcom waren ein Weckruf für die

Betreiber von IT-Infrastruktur – eine schmerzhaft Erinnerung daran, dass Abhängigkeiten teure Folgen haben können. Kleinere Nutzer von VMware finden eher mit Proxmox einen Ersatz für VMware, für größere Umgebungen kann SCS eine gute Wahl sein kann – sei es weiter im Eigenbetrieb oder auch in der Kombination mit Public-SCS-Angeboten. Aufgrund von Gesprächen mit Interessenten hat das SCS-Projekt das Feature VM-HA für die Roadmap priorisiert. Hier geht es darum, dass durch die Infrastruktur virtuelle Maschinen automatisch woanders neu gestartet werden, wenn ein Host ausfällt.

Noch bis zum 30. September 2024 läuft die Förderung des SCS-Projekts durch das BMWK **9**. Damit ist dieses Projekt dann aber noch nicht zu Ende, vielmehr muss es anschließend aus eigener Kraft überleben. Dem SCS-Team wurde in zahlreichen Gesprächen widerspiegelt, dass die nicht gewinnorientierte Aufstellung in einem anbieterübergreifenden Verband mit öffentlichen Fördergeldern viel Glaubwürdigkeit schafft. Die Fortführung soll daher nicht allein in einem kommerziellen Unternehmen stattfinden. Stattdessen wird es zwei Entitäten geben.

Die Markenrechte (derzeit bei der OSBA), die Standardisierungsbemühungen, die Zertifizierung, die Mitarbeit in den Upstream-Communities und das

neutrale Management des Ökosystems von Partnern und der Community soll eine nicht gewinnorientierte Entität betreiben. Sie sorgt für die Offenheit der Community und der Prozesse. Derzeit ist noch offen, ob es ein eigener Verein wird oder eine recht selbstständig agierende Abteilung in der OSBA.

Auf der anderen Seite gibt es die Nutzer der Referenzimplementierung. Sie erwarten eine Weiterentwicklung und Pflege, auch für die Upstream-Projekte, sowie professionelle Angebote für Wartung und Support. Das ist ein klassisches Geschäftsmodell für ein Open-Source-Unternehmen **10**. Es kann diese Leistungen durch die Zusammenarbeit mit den Technologiepartnern und die Bezahlung effektiv bündeln und sicherstellen. Dafür wird es Wartungsgebühren durch die Cloud-Betreiber einnehmen.

Das SCS-Projekt hat von vorneherein darauf geachtet, dass nicht einzelne Hersteller allein die Software bestimmen und dass das Copyright verteilt bleibt, sodass eine Monopolisierung nach dem Muster von zuletzt Redis nicht möglich ist. Das Unternehmen muss mit der Implementierung all die Anforderungen an die Offenheit (Four Opens) erfüllen, damit es die Marke SCS nutzen darf. Das steht auch weiteren Unternehmen offen.

Damit ist die Zukunft von SCS erst einmal ohne weitere öffentliche Mittel gesichert. Seine genaue Größe ab Oktober 2024 hängt noch vom Ausgang einiger Gespräche ab. Klar ist, dass es noch sehr viele Ideen gibt, SCS bedeutsamer zu machen. Das betrifft etwa Standardlösungen im Bereich klassischer Plattformdienste wie einen Datenbankdienst oder Initiativen im Bereich künstlicher Intelligenz, wo man beim Feintuning von großen Sprachmodellen oder gar beim grundlegenden Training nicht die Daten der Bürger oder des Unternehmens Dritten offenlegen will. Das SCS-Projekt freut sich über Unterstützung, private wie öffentliche, um solche Ideen auch mit der nötigen Geschwindigkeit angehen zu können. (jcb) ■

Der Autor

Kurt Garloff hat nach einer Ausbildung als Diplom-Physiker sein Berufsleben als Linux-Kernel-Entwickler sowie Team-, Abteilungs- und Entwicklungsleiter in der Open-Source-Welt verbracht, unter anderem bei Suse. Die letzten 12 Jahre beschäftigte er sich mit Open Infrastructure und verantwortete beispielsweise als Chefarchitekt bei T-Systems den Aufbau der Open Telekom Cloud. Seit 2020 leitet er das Projekt Sovereign Cloud Stack der OSBA.



Weitere Infos und interessante Links

www.lm-online.de/qr/50651