

# Kenntnis der virtuellen NFVIS Netzwerke: OVS, DPDK und SR-IOV

## Inhalt

---

[Einleitung](#)

[Verwendete Komponenten](#)

[Überblick über die Netzwerkfunktionen von NFVIS](#)

[ENCS54XX-Plattform](#)

[Catalyst 8200 CPE](#)

[Catalyst 8300 uCPE 1N20](#)

[Technologien zur Netzwerkvirtualisierung](#)

[Offener vSwitch \(OVS\)](#)

[OVS-Bridges](#)

[Defizite bei der Kontextumschaltung](#)

[DPDK \(Data Plane Development Kit\)](#)

[Kopieren von Daten](#)

[PCIe-Passthrough](#)

[Single-Root-E/A-Virtualisierung \(SR-IOV\)](#)

[Physische Funktionen \(PF\)](#)

[Virtuelle Funktionen \(VFs\)](#)

[Empfohlene Treiber für die SR-IOV-Beschleunigung auf NFVIS-fähiger Hardware](#)

[Anwendungsbeispiele für DPDK und SR-IOV](#)

[DPDK-Präferenz](#)

[SR-IOV-Präferenz](#)

[Konfiguration](#)

[Aktivieren von DPDK](#)

[Neues Netzwerk erstellen und einer neuen OVS-Bridge zuordnen](#)

[Anschließen von VNFs](#)

[Verwandte Artikel und Dokumentation](#)

---

## Einleitung

In diesem Dokument wird das virtuelle Netzwerkschema beschrieben, das die NFVIS-Plattform für die Kommunikation von VNFs in Unternehmens- und Servicenetzwerken bereitstellt.

## Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf folgenden Hardware- und Softwarekomponenten:

- ENCS5412 mit NFVIS 4.7.1-FC4

- c8300 uCPE 1N20 mit NFVIS 4.12.1-FC2

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

## Überblick über die Netzwerkfunktionen von NFVIS

Für die VNF-Überwachung werden intern ein internes Managementnetzwerk (int-mgmt-net) und eine Bridge (int-mgmt-br) verwendet, die Management-IP-Adressen aus dem Subnetz 10.20.0.0/24 zuweisen.

### ENCS54XX-Plattform

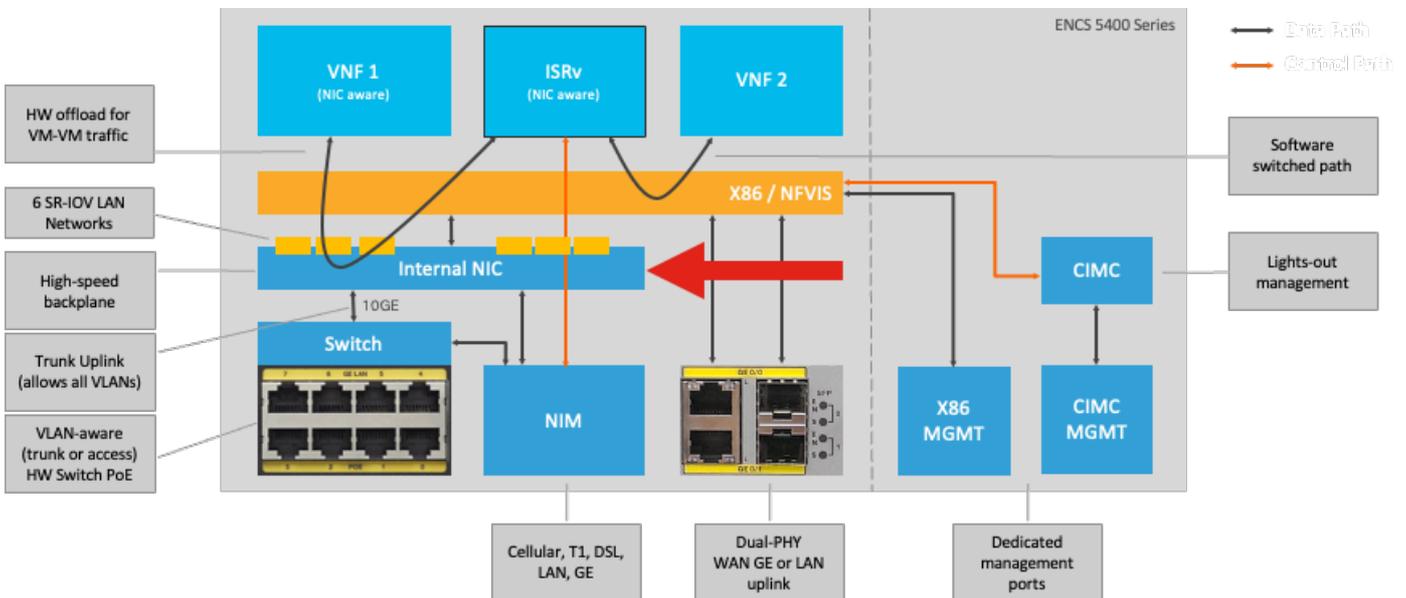


Abbildung 1. Interne Verbindungen von Hardware-Switch und WAN/LAN-Uplink-NICs

### Catalyst 8200 CPE

- Der Zugriff auf NFVIS ist standardmäßig über den WAN- oder GE0/2-LAN-Port für die Verwaltung möglich.
- Das WAN-Netzwerk (WAN-NET und WAN2-NET) und die WAN-Bridge (WAN-br und WAN2-br) sind standardmäßig für die Aktivierung von DHCP konfiguriert. GE0-0 ist standardmäßig mit der WAN-Bridge und GE0-1 mit der WAN2-Bridge verknüpft.
- Auf die Management-IP-Adresse 192.168.1.1 des Catalyst 8200 UCPE kann über GE0-2 zugegriffen werden.
- GE0-2 ist mit der LAN-Bridge verbunden.
- Ein internes Managementnetzwerk (int-mgmt-net) und eine Bridge (int-mgmt-br) werden

erstellt und intern für die Systemüberwachung verwendet.

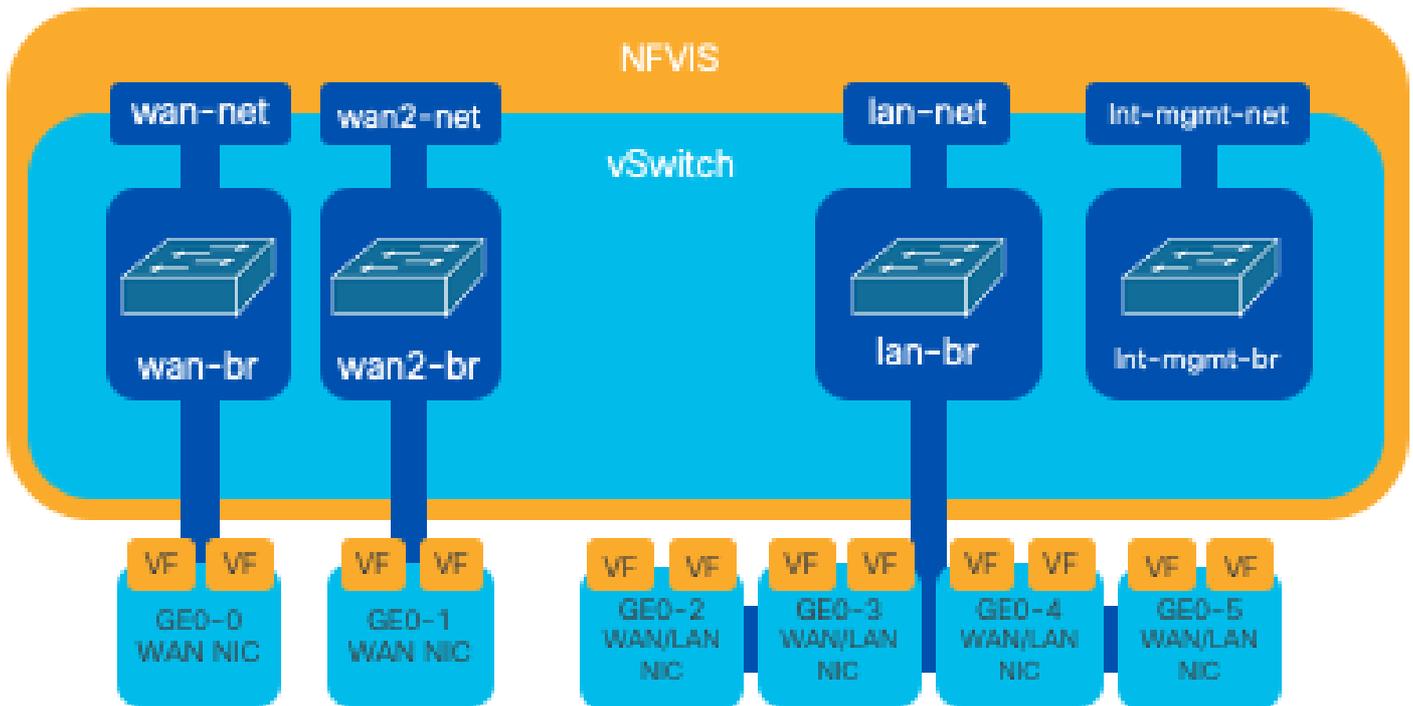


Abbildung 2. Internes Bridging und virtuelle Switches für die Netzwerkkarten der Serie 8200

### Catalyst 8300 uCPE 1N20

1. Der Zugriff auf NFVIS ist standardmäßig über die FPGE (Front Panel Gigabit Ethernet)-WAN-Ports oder über den GE0-2 LAN-Port für das Management möglich.
2. Zur Aktivierung von DHCP sind standardmäßig das WAN-Netzwerk (WAN-NET) und eine WAN-Bridge (WAN-BR) festgelegt. GE0-0 ist standardmäßig mit der WAN-Bridge verbunden.
3. WAN-Netzwerk (WAN2-NET) und WAN-Bridge (WAN2-BR) werden standardmäßig erstellt, sind jedoch keinen physischen Ports zugeordnet.
4. GE0-2 ist mit der LAN-Bridge verbunden, alle anderen Ports sind nicht mit OVS verbunden
5. Der Zugriff auf die Management-IP-Adresse 192.168.1.1 für C8300-uCPE erfolgt über GE0-2
6. Ein internes Managementnetzwerk (int-mgmt-net) und eine Bridge (int-mgmt-br) werden erstellt und intern für die Systemüberwachung verwendet.

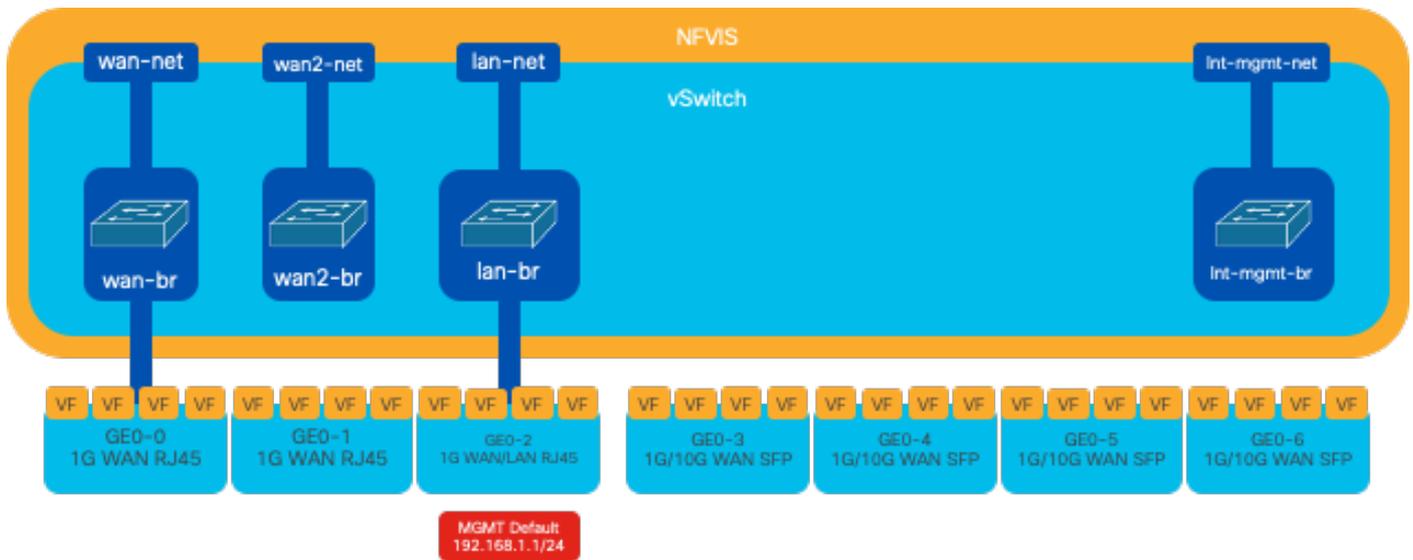


Abbildung 3: Internes Bridging und virtuelle Switches für die Netzwerkkarten der Serie 8300

## Technologien zur Netzwerkvirtualisierung

### Offener vSwitch (OVS)

Open vSwitch (OVS) ist ein virtueller Open-Source-Switch mit mehreren Ebenen, der die Netzwerkautomatisierung durch programmatische Erweiterungen ermöglicht und gleichzeitig standardmäßige Management-Schnittstellen und -Protokolle wie NetFlow, sFlow, IPFIX, RSPAN, CLI, LACP und 802.1ag unterstützt. Es wird häufig in großen virtualisierten Umgebungen eingesetzt, insbesondere bei Hypervisoren, um den Netzwerkverkehr zwischen virtuellen Systemen (VMs) zu verwalten. Es ermöglicht die Erstellung ausgefeilter Netzwerktopologien und Richtlinien, die direkt über die NFVIS-Schnittstelle verwaltet werden, und bietet eine vielseitige Umgebung für die Virtualisierung von Netzwerkfunktionen.

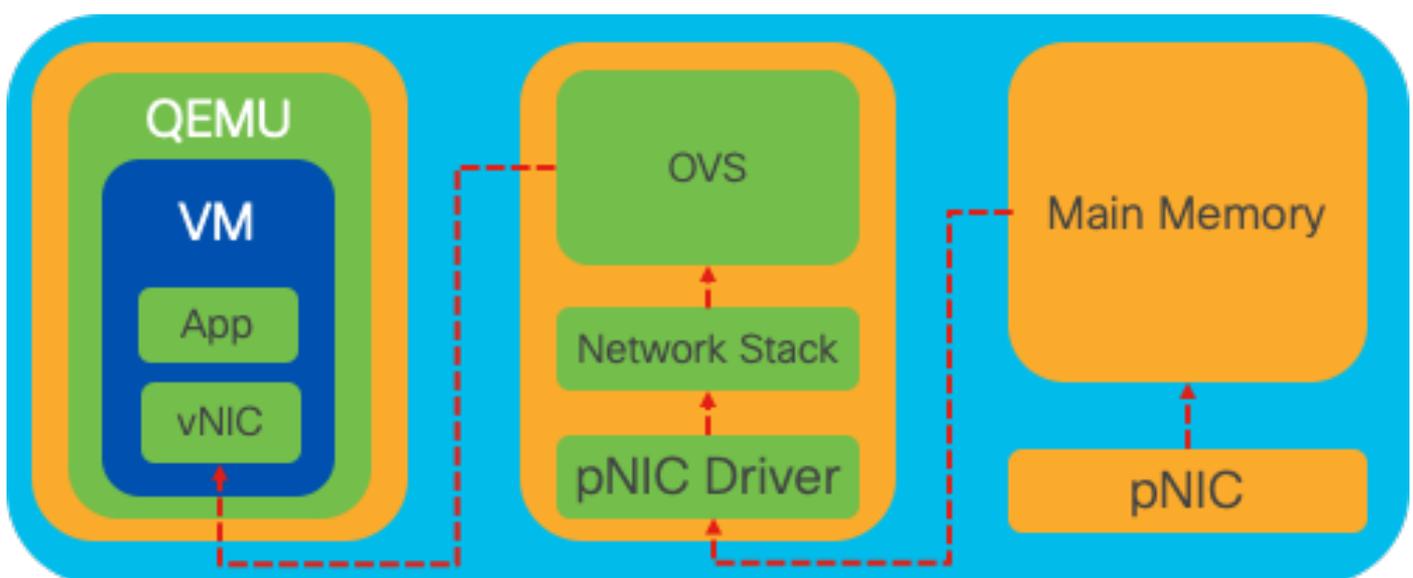


Abbildung 4: OVS-Konfiguration im Linux-Kernel

## OVS-Bridges

Dabei werden virtuelle Netzwerk-Bridges und Flows-Regeln zum Weiterleiten von Paketen zwischen Hosts verwendet. Sie verhält sich wie ein physischer Switch, der nur virtualisiert ist.

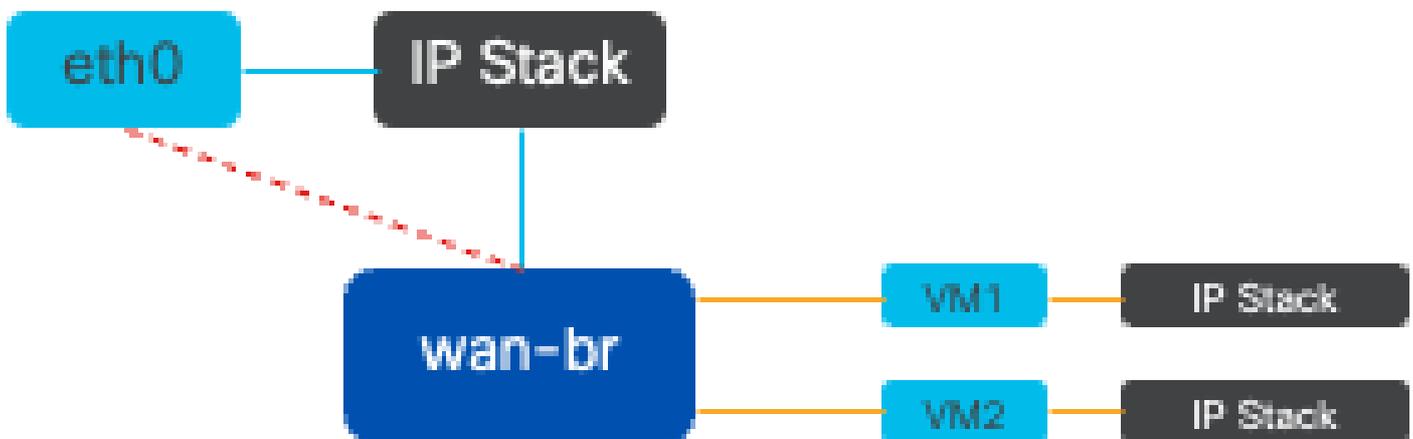


Abbildung 5: Beispiel für die Implementierung von 2 VMs oder VNFs, die an die WAN-BR-Bridge angeschlossen sind

### Defizite bei der Kontextumschaltung

Wenn ein Netzwerkpaket bei einer Netzwerkkarte (NIC) eingeht, löst es einen Interrupt aus. Dies ist ein Signal an den Prozessor, das darauf hinweist, dass es sofort bearbeitet werden muss. Die CPU unterbricht ihre aktuellen Aufgaben, um den Interrupt zu behandeln. Dieser Prozess wird als Interrupt-Verarbeitung bezeichnet. Während dieser Phase liest die CPU unter der Kontrolle des Betriebssystem-Kernels das Paket von der Netzwerkkarte in den Speicher und entscheidet die nächsten Schritte basierend auf dem Ziel und Zweck des Pakets. Das Ziel besteht darin, das Paket schnell zu verarbeiten oder an die vorgesehene Anwendung weiterzuleiten, wodurch die Latenz minimiert und der Durchsatz maximiert wird.

Kontext-Switching ist der Prozess, durch den eine CPU von der Ausführung von Aufgaben in einer Umgebung (Kontext) zu einer anderen wechselt. Dies ist besonders relevant, wenn zwischen Benutzermodus und Kernelmodus gewechselt wird:

- **Benutzermodus:** Dies ist ein eingeschränkter Verarbeitungsmodus, in dem die meisten Anwendungen ausgeführt werden. Anwendungen im Benutzermodus haben keinen direkten Zugriff auf Hardware oder Referenzspeicher und müssen mit dem Kernel des Betriebssystems kommunizieren, um diese Vorgänge auszuführen.
- **Kernel Mode (Kernelmodus):** Dieser Modus gewährt dem Betriebssystem vollständigen Zugriff auf die Hardware und den gesamten Speicher. Der Kernel kann jeden CPU-Befehl ausführen und auf jede Speicheradresse verweisen. Der Kernelmodus ist für die Ausführung von Aufgaben wie das Verwalten von Hardwaregeräten, Arbeitsspeicher und das Ausführen von Systemaufrufen erforderlich.

Wenn eine Anwendung einen Vorgang ausführen muss, der Berechtigungen auf Kernelebene

erfordert (z. B. das Lesen eines Netzwerkpakets), tritt ein Kontext-Switch auf. Die CPU wechselt vom Benutzermodus in den Kernelmodus, um den Vorgang auszuführen. Nach Abschluss des Vorgangs kehrt ein weiterer Kontextwechsel die CPU in den Benutzermodus zurück, um die Ausführung der Anwendung fortzusetzen. Dieser Switching-Prozess ist für die Aufrechterhaltung der Systemstabilität und -sicherheit von entscheidender Bedeutung, führt jedoch zu einem Overhead, der die Leistung beeinträchtigen kann.

OVS wird hauptsächlich im Benutzerbereich des Betriebssystems ausgeführt, was mit zunehmendem Datendurchsatz zu einem Engpass werden kann. Dies liegt daran, dass mehr Kontext-Switches erforderlich sind, damit die CPU in den Kernel-Modus wechseln kann, um Pakete zu verarbeiten, was die Leistung beeinträchtigt. Diese Einschränkung ist insbesondere in Umgebungen mit hohen Paketraten oder wenn ein präzises Timing entscheidend ist, zu beobachten. Um diese Leistungseinschränkungen zu beheben und die Anforderungen moderner Hochgeschwindigkeitsnetzwerke zu erfüllen, wurden Technologien wie DPDK (Data Plane Development Kit) und SR-IOV (Single Root I/O Virtualization) entwickelt.

## DPDK (Data Plane Development Kit)

DPDK ist eine Zusammenstellung von Bibliotheken und Treibern zur Beschleunigung der Paketverarbeitungs-Workloads auf einer Vielzahl von CPU-Architekturen. Durch Umgehung des herkömmlichen Kernel-Netzwerk-Stacks (Vermeidung von Kontext-Switching) kann DPDK den Durchsatz der Datenebene erheblich steigern und die Latenz verringern. Dies ist besonders für VNFs mit hohem Durchsatz von Vorteil, die Kommunikation mit niedriger Latenz erfordern, sodass NFVIS eine ideale Plattform für leistungsempfindliche Netzwerkfunktionen ist.

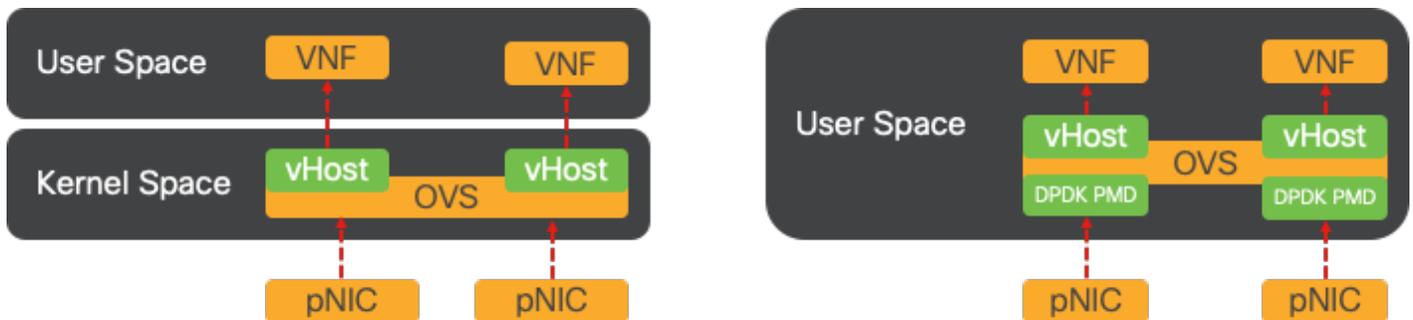


Abbildung 6: Optimierte kontextbasierte Switching-Lösungen für herkömmliche OVS (linke Seite) und DPDK OVS (rechte Seite)

Unterstützung für DPDK für OVS wurde in NFVIS 3.10.1 für ENCS und 3.12.2 für andere Plattformen gestartet.

- Service Chain-Durchsatz in der Nähe von SRIOV, besser als OVS ohne DPDK
- Für VNF ist ein Virentreiber erforderlich.
- Unterstützte Plattformen:
  - ENCS 3.10.1 und höher
  - UCSE, UCS-C, CSP5K 3.12.1 und höher
- DPDK für Port-Channels wird seit 4.12.1 unterstützt.
- Paket-/Datenverkehrserfassung: wird in DPDK nicht unterstützt.

- Spanning-Datenverkehr auf PNIC: wird in DPDK nicht unterstützt.
- Nachdem OVS-DPDK aktiviert wurde, kann es nicht als einzelne Funktion deaktiviert werden. Die einzige Möglichkeit, DPDK zu deaktivieren, ist ein Zurücksetzen auf die Werkseinstellungen.

## Kopieren von Daten

Bei herkömmlichen Netzwerkansätzen müssen Daten oft mehrmals kopiert werden, bevor sie ihr Ziel im Speicher des virtuellen Systems erreichen. Beispielsweise muss ein Paket von der Netzwerkkarte in den Kernelbereich, dann in den Benutzerbereich für die Verarbeitung durch einen virtuellen Switch (wie OVS) und schließlich in den VM-Speicher kopiert werden. Jeder Kopiervorgang verursacht eine Verzögerung und erhöht die CPU-Auslastung trotz der Leistungsverbesserungen, die DPDK bietet, indem der Kernel-Netzwerk-Stack umgangen wird.

Zu diesen Gemeinkosten gehören Speicherkopien und die Verarbeitungszeit, die erforderlich ist, um Pakete im Benutzerbereich zu verarbeiten, bevor sie an das virtuelle System weitergeleitet werden können. PCIe Passthrough und SR-IOV beseitigen diese Engpässe, indem sie die direkte gemeinsame Nutzung eines physischen Netzwerkgeräts (wie einer Netzwerkkarte) durch mehrere VMs ermöglichen, ohne das Host-Betriebssystem im gleichen Umfang zu nutzen wie herkömmliche Virtualisierungsverfahren.

## PCIe-Passthrough

Die Strategie beinhaltet die Umgehung des Hypervisors, um Virtual Network Functions (VNFs) den direkten Zugriff auf eine Netzwerkkarte (NIC) zu ermöglichen und so einen nahezu maximalen Durchsatz zu erzielen. Dieser Ansatz wird als PCI-Passthrough bezeichnet, bei dem eine komplette NIC ohne Hypervisor-Intervention einem Gastbetriebssystem zugewiesen werden kann. In dieser Konfiguration funktioniert das virtuelle System so, als wäre es direkt mit der Netzwerkkarte verbunden. Wenn beispielsweise zwei NIC-Karten zur Verfügung stehen, kann jede exklusiv verschiedenen VNFs zugewiesen werden, sodass diese direkten Zugriff erhalten.

Dieses Verfahren hat jedoch einen Nachteil: Wenn nur zwei NICs verfügbar sind und ausschließlich von zwei separaten VNFs verwendet werden, bleiben alle zusätzlichen VNFs, z. B. eine dritte, aufgrund des Fehlens einer dedizierten NIC, die für sie zur Verfügung steht, ohne NIC-Zugriff. Eine alternative Lösung besteht in der Verwendung von Single Root I/O Virtualization (SR-IOV).

## Single-Root-E/A-Virtualisierung (SR-IOV)

Eine Spezifikation, die es ermöglicht, dass ein einzelnes physisches PCI-Gerät, wie eine Netzwerkkarte (NIC), als mehrere separate virtuelle Geräte angezeigt wird. Diese Technologie bietet direkten VM-Zugriff auf physische Netzwerkgeräte und reduziert damit den Overhead und verbessert die E/A-Leistung. Es teilt ein einzelnes PCIe-Gerät in mehrere virtuelle Abschnitte auf, von denen jeder einzelnen VMs oder VNFs zugewiesen werden kann. Damit wird die durch eine begrenzte Anzahl von NICs verursachte Einschränkung wirksam behoben. Diese virtuellen Segmente, die als virtuelle Funktionen (Virtual Functions, VFs) bezeichnet werden, ermöglichen die gemeinsame Nutzung von NIC-Ressourcen zwischen mehreren VNFs. Die physische Funktion

(PF) bezieht sich auf die tatsächliche physische Komponente, die SR-IOV-Funktionen unterstützt.

Durch die Nutzung von SR-IOV kann NFVIS dedizierte NIC-Ressourcen bestimmten VNFs zuweisen und so eine hohe Leistung und eine niedrige Latenz gewährleisten, indem der direkte Speicherzugriff (Direct Memory Access, DMA) von Netzwerkpaketen direkt in den entsprechenden VM-Speicher erleichtert wird. Dieser Ansatz minimiert den CPU-Aufwand, da nur die Pakete verarbeitet werden müssen, und senkt somit die CPU-Auslastung. Dies ist besonders für Anwendungen nützlich, die eine garantierte Bandbreite benötigen oder strenge Leistungsanforderungen haben.

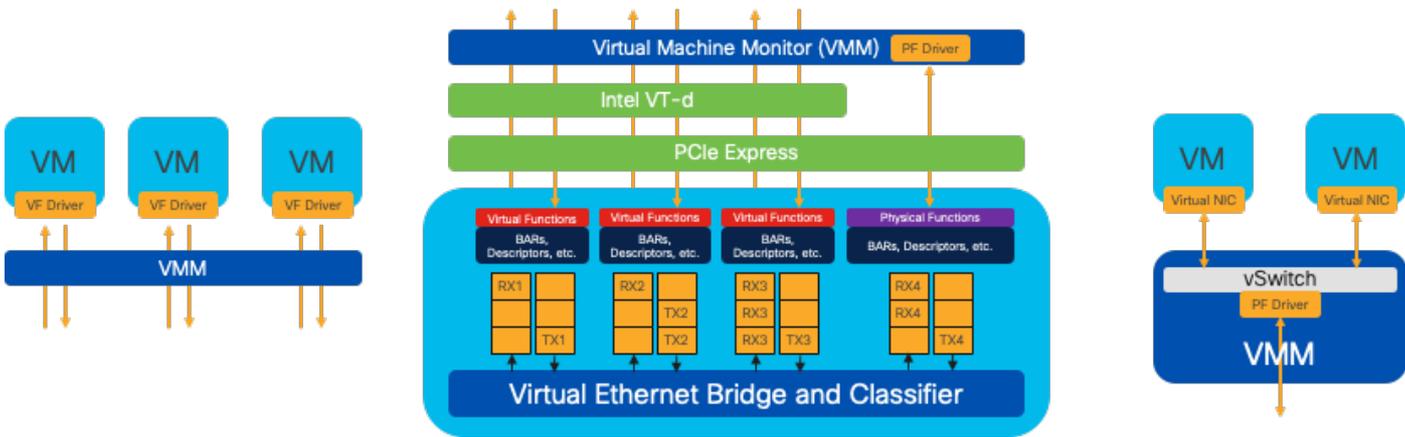


Abbildung 7: NFVIS SR-IOV PCIe-Ressourcentrennung durch Hardwarefunktionen

### Physische Funktionen (PF)

Es handelt sich dabei um PCIe-Funktionen mit vollem Funktionsumfang, die auf eine spezielle Hardware-Box für spezielle Netzwerkfunktionen zurückgreifen. Dabei handelt es sich um PCIe-Funktionen, die wie jedes andere PCIe-Gerät erkannt, verwaltet und manipuliert werden können. Zu den physischen Funktionen gehören die SR-IOV-Funktionen, mit denen ein PCIe-Gerät konfiguriert und gesteuert werden kann.

### Virtuelle Funktionen (VFs)

Es handelt sich um optimierte Funktionen mit minimalen Konfigurationsressourcen (geringes Gewicht), die sich ausschließlich auf die Verarbeitung von E/A als einfache PCIe-Funktionen konzentrieren. Jede virtuelle Funktion stammt aus einer physischen Funktion. Die Gerätehardware schränkt die mögliche Anzahl virtueller Funktionen ein. Ein Ethernet-Port, das physische Gerät, kann zahlreichen virtuellen Funktionen entsprechen, die dann verschiedenen virtuellen Systemen zugewiesen werden können.

### Empfohlene Treiber für die SR-IOV-Beschleunigung auf NFVIS-fähiger Hardware

Plattform	Netzwerkkarten	Netzwerktreiber
ENCS 54XX	Backplane-Switch	i40e
ENCS 54XX	GE0-0 und GE0-1	GbE
Catalyst 8200 CPE	GE0-0 und GE0-1	lxgbe

## Anwendungsbeispiele für DPDK und SR-IOV

### DPDK-Präferenz

Insbesondere in Szenarien, in denen der Netzwerkverkehr primär von Ost nach West fließt (d. h. innerhalb desselben Servers verbleibt), übertrifft DPDK SR-IOV. Die Überlegung ist einfach: Wenn der Datenverkehr intern innerhalb des Servers verwaltet wird, ohne auf die NIC zugreifen zu müssen, bietet SR-IOV keinen Vorteil. Tatsächlich könnte SR-IOV zu Ineffizienzen führen, indem der Datenverkehrspfad unnötig erweitert wird und NIC-Ressourcen belegt werden. Für die interne Verwaltung des Serverdatenverkehrs ist die Nutzung von DPDK daher die effizientere Wahl.

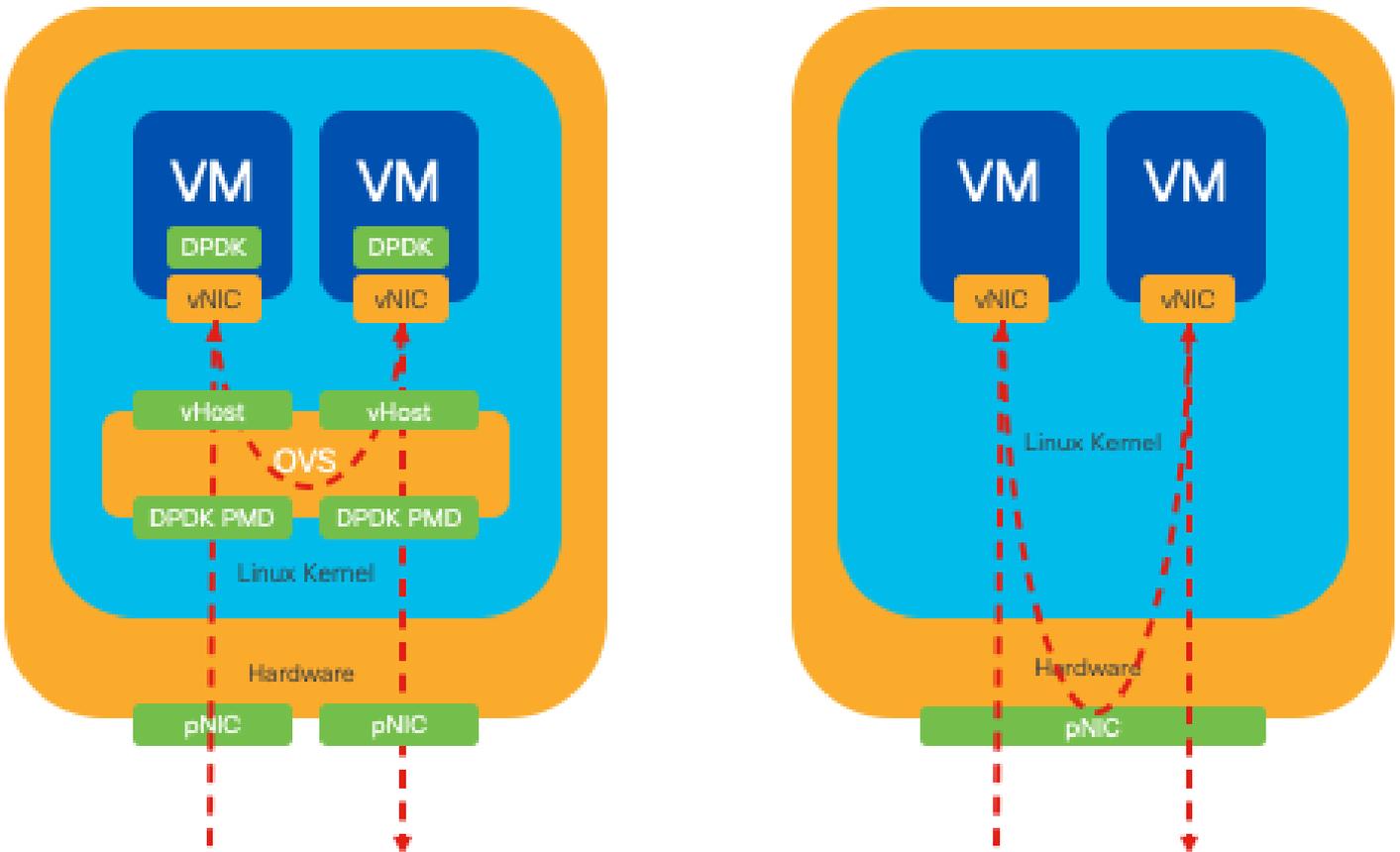


Abbildung 8: DPDK- und SR-IOV-Paketüberbrückung im Ost-West-Datenverkehr

### SR-IOV-Präferenz

In Situationen, in denen der Netzwerkverkehr von Nord nach Süd oder sogar von Ost nach West fließt, insbesondere zwischen Servern, erweist sich die Verwendung von SR-IOV als vorteilhafter als DPDK. Dies gilt insbesondere für die Kommunikation zwischen Servern. Da ein solcher Datenverkehr unweigerlich die Netzwerkkarte durchlaufen muss, kann die Entscheidung für ein DPDK-optimiertes OVS unnötigerweise zusätzliche Komplexität und potenzielle

Leistungseinschränkungen mit sich bringen. Aus diesem Grund wird SR-IOV in dieser Situation als bevorzugte Lösung erachtet und bietet einen einfachen und effizienten Pfad für die Verarbeitung des Datenverkehrs zwischen Servern.

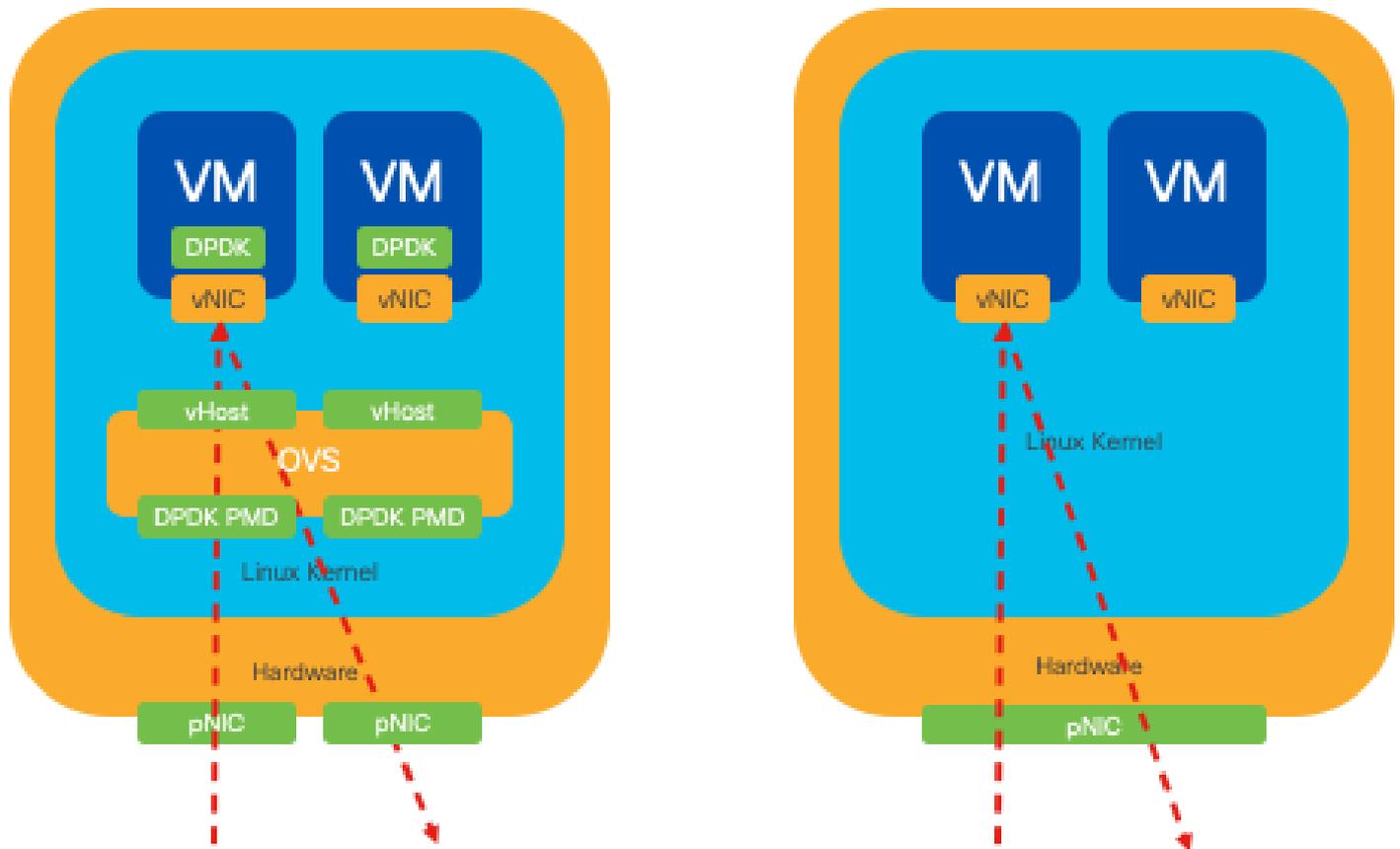
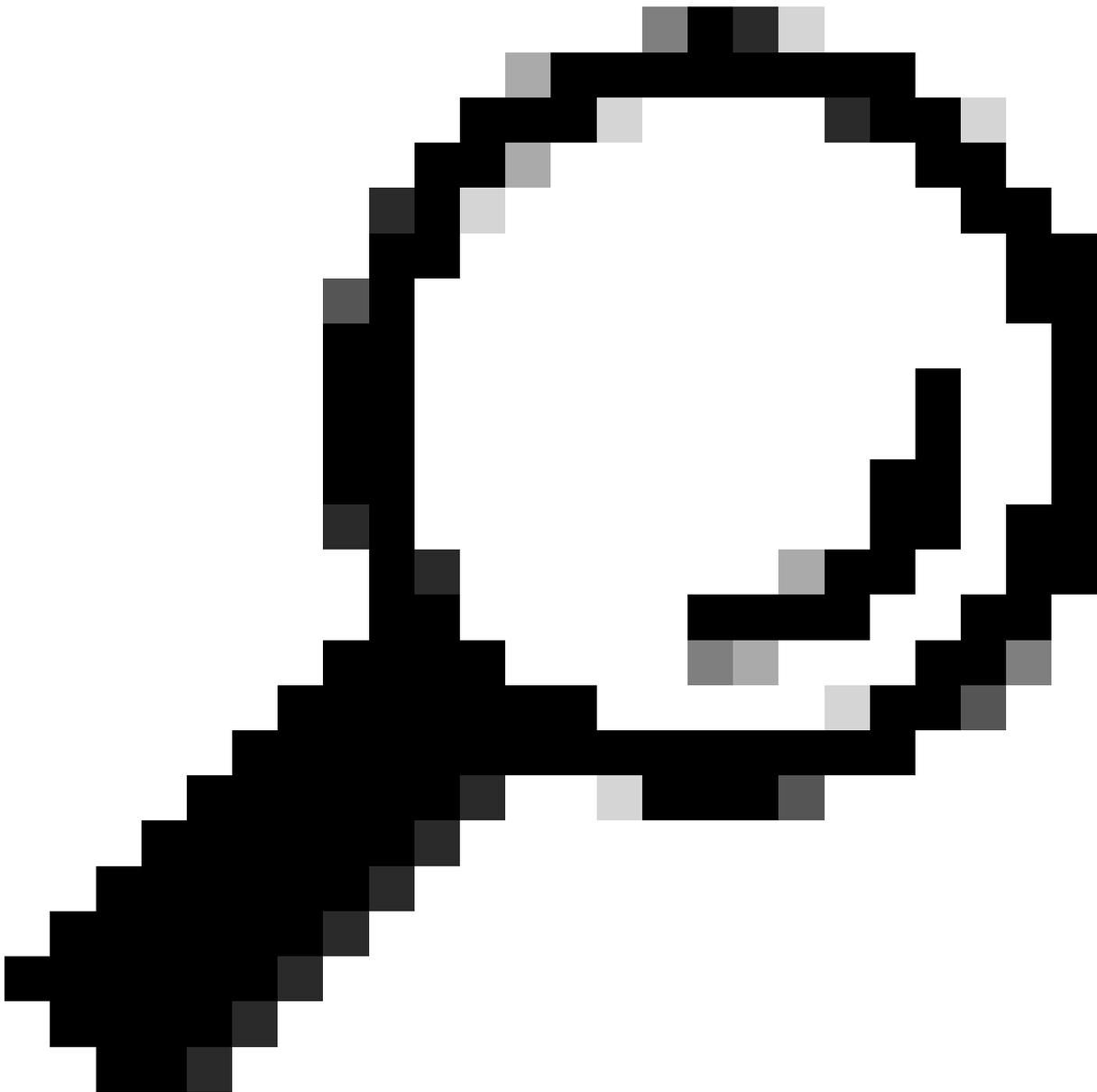


Abbildung 9: DDPK- und SR-IOV-Paketüberbrückung im Nord-Süd-Datenverkehr



Tipp: Denken Sie daran, dass es möglich ist, die Leistung einer SR-IOV-basierten Konfiguration zu verbessern, indem SR-IOV mit DPDK in eine Virtual Network Function (VNF) integriert wird. Dies gilt jedoch nicht für das Szenario, in dem DPDK wie zuvor beschrieben in Verbindung mit OVS verwendet wird.

---

## Konfiguration

### Aktivieren von DPDK

Um DPDK über die GUI zu aktivieren, müssen Sie zu Configuration > Virtual Machine > Networking > Networks (Konfiguration > Virtuelle Maschine > Netzwerk) navigieren. Klicken Sie im Menü auf den Switch, um die Funktion zu aktivieren.

# Networks

## Networks Information and Configuration

---

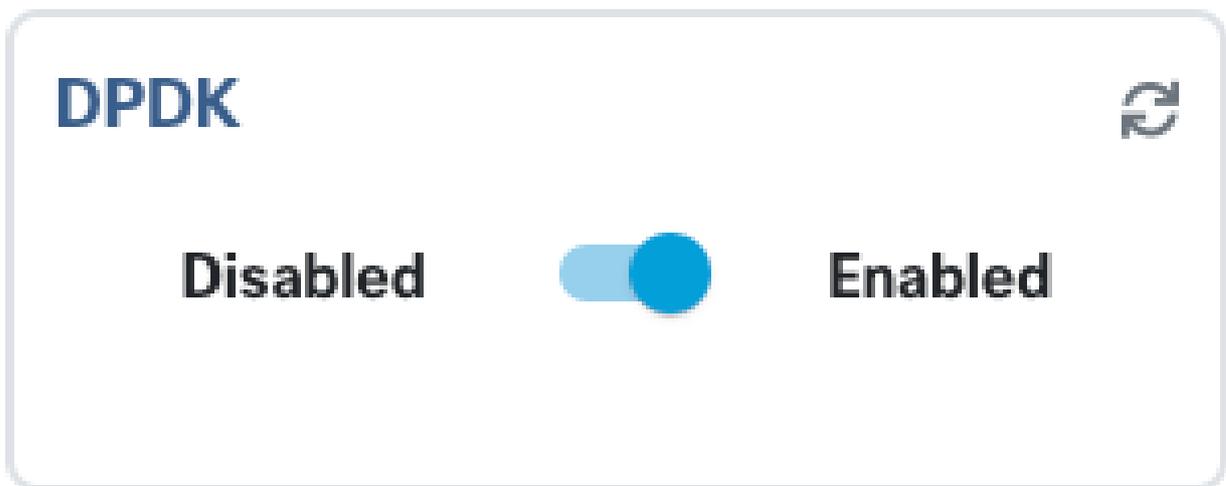
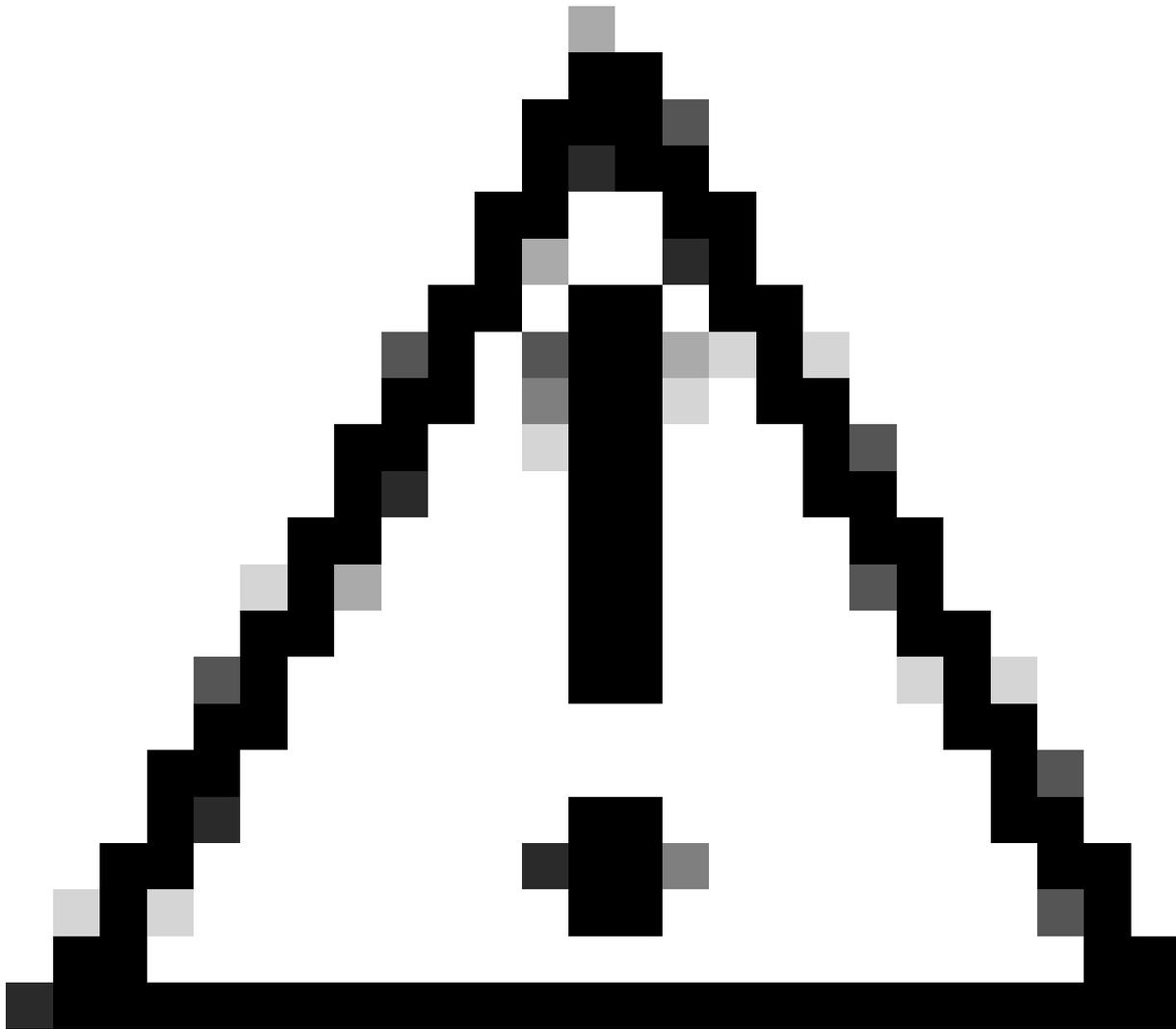


Abbildung 10: Folientaste auf der Benutzeroberfläche für die DPDK-Aktivierung verfügbar

Für die CLI müssen Sie sie in den globalen Systemeinstellungen im Konfigurationsmodus aktivieren.

```
nfvis(config)# system settings dpdk enable
```



Achtung: DPDK kann nur deaktiviert werden, wenn über NFVIS ein Zurücksetzen auf die Werkseinstellungen durchgeführt wird.

## Neues Netzwerk erstellen und einer neuen OVS-Bridge zuordnen

Navigieren Sie zu Configuration > Virtual Machine > Networking > Networks. Klicken Sie auf der Seite "Netzwerke" auf das Pluszeichen (+) oben links für die Tabelle "Netzwerke".

#	Network	Mode	Vlan	Vlan-Range	Native Vlan	Bridge	Interface	Action
1	wan-net	trunk				wan-br	GE0-0	 
2	wan2-net	trunk				wan2-br	GE0-1	 
3	lan-net	trunk				lan-br	GE0-2	 

Abbildung 11: Ansicht der Netzwerktabelle in der NFVIS-GUI

Nennen Sie das Netzwerk, und ordnen Sie es einer neuen Bridge zu. VLAN- und Schnittstellenbindungsoptionen können von den Anforderungen der Netzwerkinfrastruktur abhängen.

# Add Network

Network \*

inter-vnf-net

Mode \*

trunk

Vlan

Vlan-Range

Native Vlan

1

Bridge \*

Existing  Create New

Bridge

inter-vnf-br

Interface

Submit

Cancel

Reset

Abbildung 12. "Add Network"-Modus zur Erstellung virtueller Netzwerke in der NFVIS-GUI

Nachdem Sie auf die Schaltfläche Submit (Senden) geklickt haben, müssen Sie in der Lage sein, das neu erstellte Netzwerk zu überprüfen, das an die Tabelle Networks (Netzwerke) angehängt ist.



The screenshot shows the 'Add Network' mode in the NFVIS GUI. At the top, there is a '+ [document icon]' button and a red circular refresh icon. Below this is a header for the 'Networks' table, which includes 'Total Record: 4' and a search bar. The table has the following columns: #, Network, Mode, Vlan, Vlan-Range, Native Vlan, Bridge, Interface, and Action. There are four rows of network data.

#	Network	Mode	Vlan	Vlan-Range	Native Vlan	Bridge	Interface	Action
1	wan-net	trunk				wan-br	GE0-0	[edit] [delete]
2	wan2-net	trunk				wan2-br	GE0-1	[edit] [delete]
3	lan-net	trunk				lan-br	GE0-2	[edit] [delete]
4	inter-vnf-net	trunk			1	inter-vnf-br		[edit] [delete]

Abbildung 13: Ansicht der Netzwerktabelle aus der NFVIS-GUI, wobei sich das Symbol "Refresh" in der oberen rechten Ecke befindet (rot hervorgehoben)



Hinweis: Wenn das neue Netzwerk nicht in der Tabelle angezeigt wird, klicken Sie auf die Aktualisierungsschaltfläche oben rechts, oder aktualisieren Sie die gesamte Seite.

---

Wenn die Ausführung innerhalb von über die CLI erfolgt, wird jedes Netzwerk und jede Bridge im Konfigurationsmodus erstellt. Der Workflow entspricht der GUI-Version.

1. Erstellen Sie die neue Bridge.

```
nfvis(config)# bridges bridge inter-vnf-br2
nfvis(config-bridge-inter-vnf-br2)# commit
```

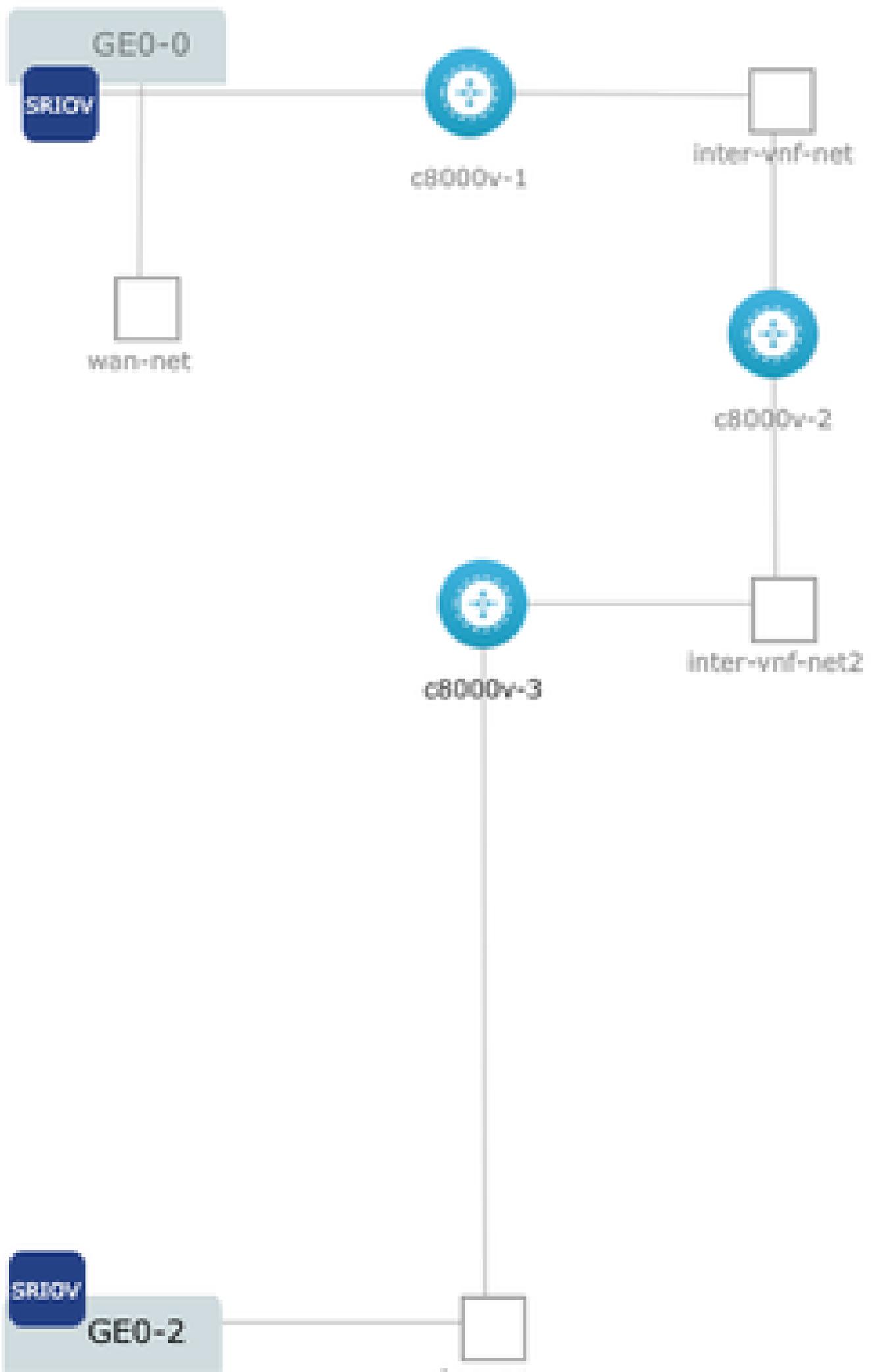
2. Erstellen Sie ein neues Netzwerk, und ordnen Sie es der zuvor erstellten Bridge zu.

```
nfvis(config)# networks network inter-vnf-net2 bridge inter-vnf-br2 trunk true native-vlan 1
```

```
nfvis(config-network-inter-vnf-net2)# commit
```

## Anschließen von VNFs

Um mit einer Netzwerktopologie oder einer einzelnen VFN-Bereitstellung zu beginnen, navigieren Sie zu Configuration > Deploy. Sie können eine VM oder einen Container aus der Auswahlliste in den Topologiebereich ziehen, um mit der Erstellung Ihrer virtualisierten Infrastruktur zu beginnen.



## Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.