

5G - SDN- NFV

In particolare il 5G: il Software-Defined Networking (SDN) e la Network Functions Virtualization (NFV)¹.

SDN (Software-Defined Networking)

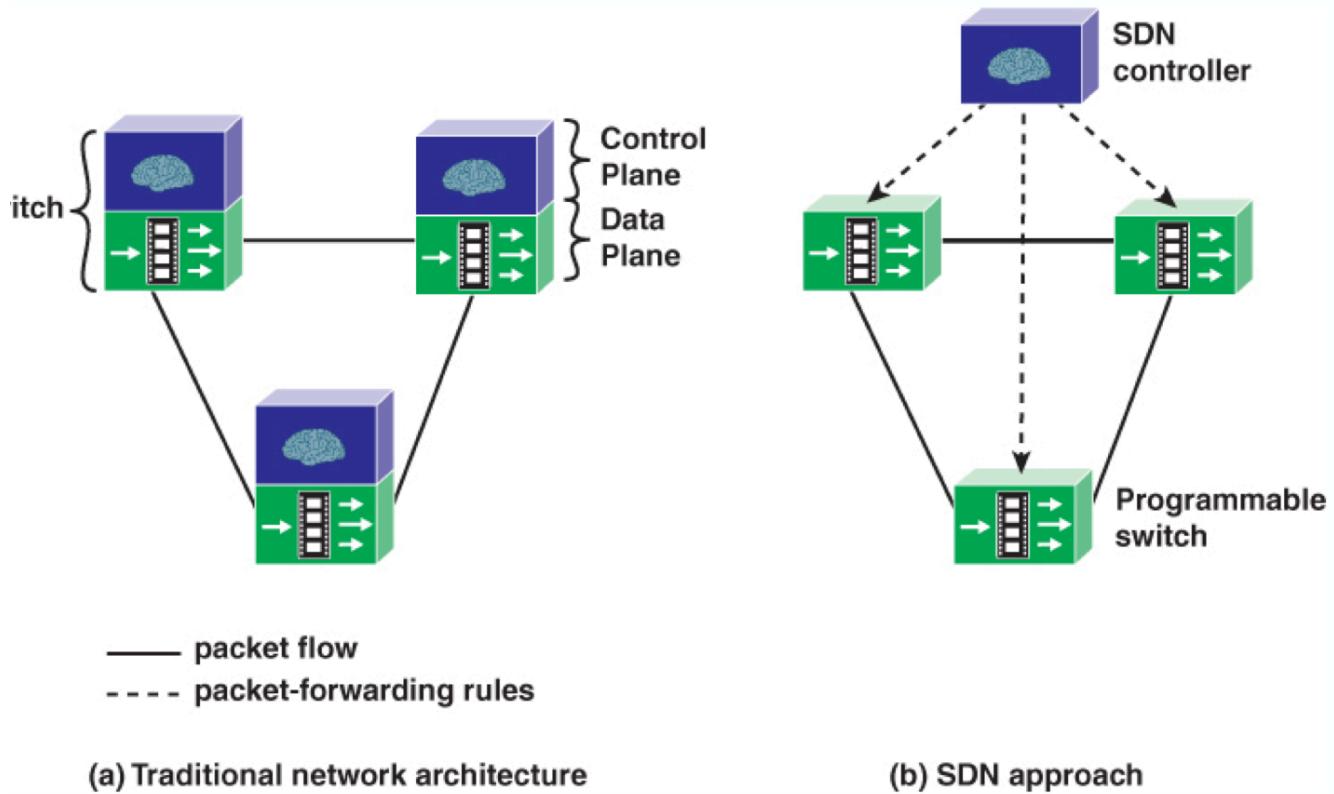
Il documento inizia descrivendo le sfide delle reti IP tradizionali di fronte alle crescenti esigenze di banda, bassa latenza e alta affidabilità del 5G. Le reti IP tradizionali sono basate su un approccio decentralizzato e auto-organizzante (sistemi autonomi, AS), con una gestione complessa delle tabelle di routing dinamiche e delle interconnessioni.

L'aumento dell'uso dei dispositivi mobili e la virtualizzazione dei server hanno introdotto modelli di traffico dinamici e requisiti di flessibilità che le infrastrutture di rete convenzionali faticano a soddisfare, richiedendo riconfigurazioni lunghe e complesse.

Cos'è l'SDN? L'SDN è stato sviluppato per semplificare l'amministrazione della rete, proponendo un approccio centralizzato. I suoi principi chiave includono la gestione centralizzata, il controllo di rete direttamente programmabile (ottenuto separando il piano di inoltro dati dal piano di controllo) e una maggiore agilità¹¹.

Componenti e Architettura SDN:

- Controllore SDN centralizzato: Gestisce logicamente la rete.
- Switch compatibili con SDN: Hardware/Software che supporta i protocolli SDN.
- Protocolli di gestione: Come OpenFlow e NETCONF.



L'SDN consente agli amministratori di rete di programmare e ottimizzare le risorse di rete tramite script automatici e standard aperti, prevenendo il vendor lock-in.

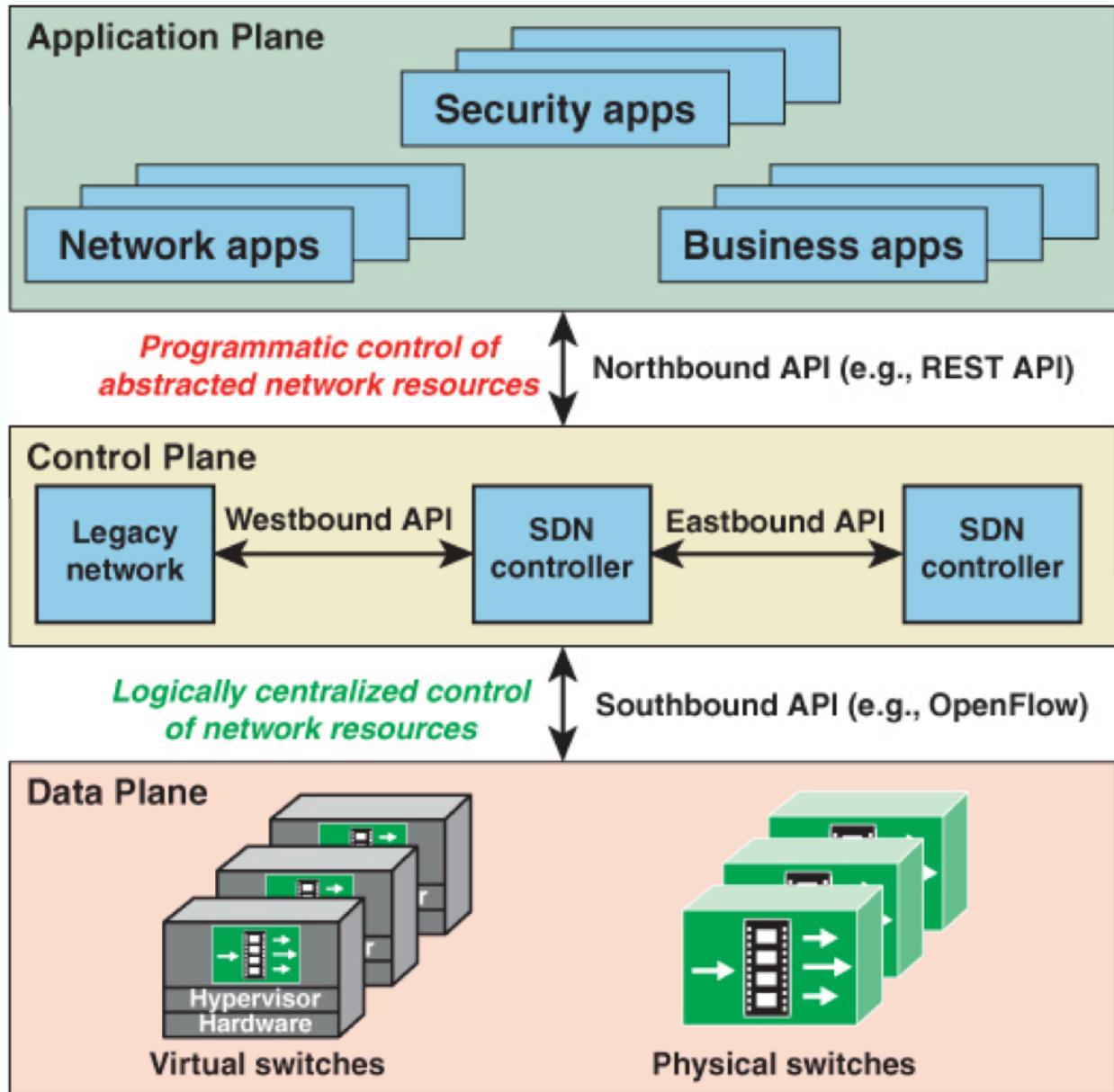
Il **vendor lock in** viene prevenuto poichè il controllo centralizzato è comandato tramite software e ,quindi, gli amministratori possono configurare e ottimizzare la rete indipendentemente dall' hardware sottostante.

Vengono usati protocolli aperti e interscambiabili con OpenFlow e NETCONF.

La rete viene programmata rapidamente tramite API senza passare da interfacce proprietarie.

La separazione tra piano dati (inoltro pacchetti) e piano di controllo (intelligenza di routing e gestione politiche) è il concetto centrale. L'architettura SDN si articola su:

- Interfaccia Southbound: Utilizzata dal controllore per programmare il piano dati (es. OpenFlow), definendo come gli switch fisici e virtuali inoltrano i pacchetti.
- Interfaccia Northbound: Permette alle applicazioni di comunicare i requisiti di rete al controllore SDN (es. API REST), offrendo una vista astratta delle risorse di rete.
 - + Permette alle applicazioni di specificare cosa vogliono ottenere dalla rete (es. “crea una VLAN per questa applicazione”, “priorità alta per questo traffico”)
 - Non richiede alle applicazioni di sapere come farlo (niente dettagli su switch, porte, indirizzi fisici, ecc.)
- Interfacce Eastbound/Westbound: Consentono la comunicazione tra controllori distribuiti (per scalabilità e privacy) e l'integrazione con reti non-SDN (legacy).



Ruolo del Controllore SDN:

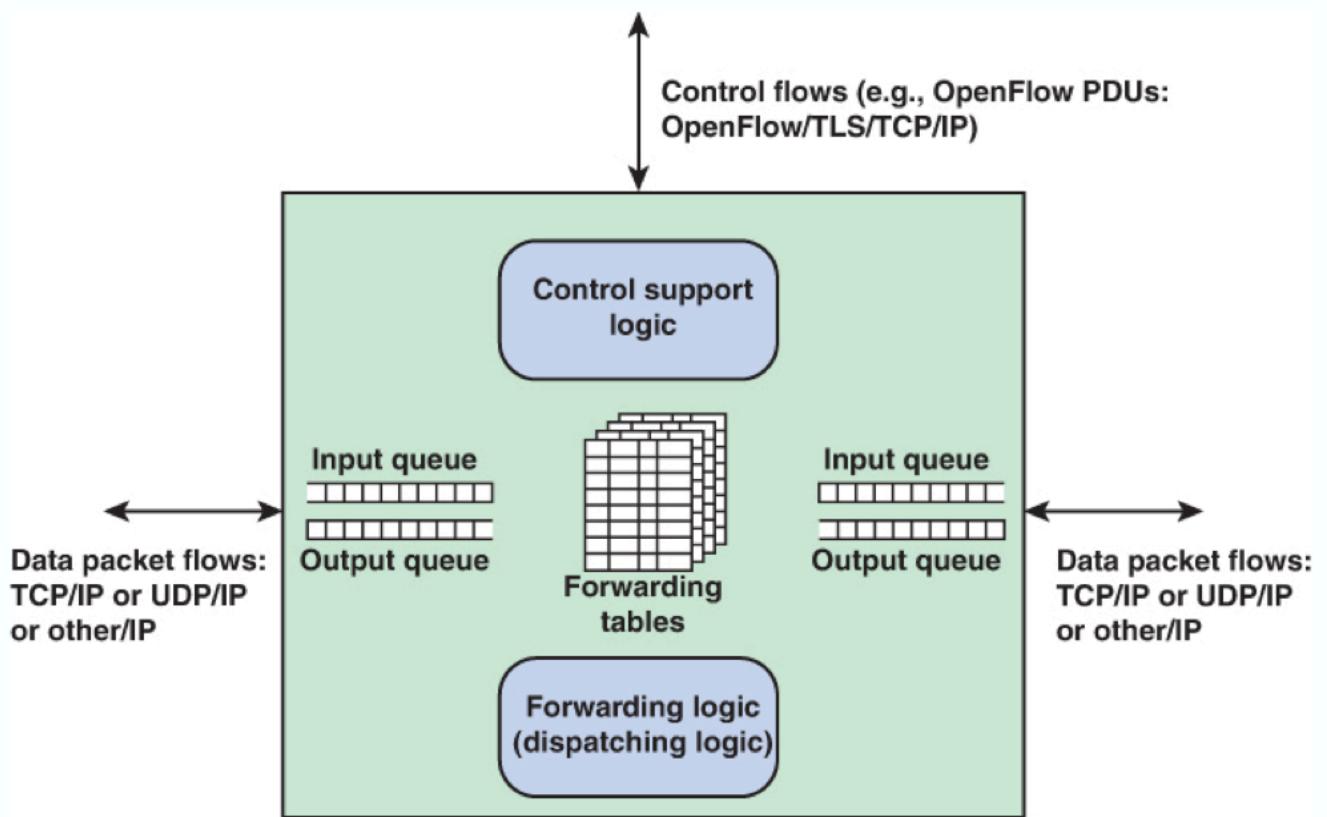
Il controllore SDN è essenzialmente un sistema operativo di rete (NOS) che gestisce lo stato di inoltro degli switch, traduce le richieste di servizio in comandi di rete specifici e fornisce alle applicazioni dettagli sulla topologia e l'attività del piano dati. Offre funzioni essenziali come la gestione della topologia, delle statistiche, dei dispositivi, e dei meccanismi di sicurezza.

SDN Data Plane e OpenFlow:

Il piano dati SDN (o livello risorsa) è composto da dispositivi di inoltro di rete che eseguono funzioni di trasporto e elaborazione dati basate sulle decisioni del piano di controllo. Il protocollo OpenFlow è ampiamente utilizzato come interfaccia southbound.

- Flow Table: Ogni switch OpenFlow ha una tabella di flusso che memorizza le regole di inoltro. Le voci della tabella includono:
 - Match Fields (identificano i pacchetti).
 - Actions (istruzioni su cosa fare con il pacchetto, es. inoltro, scarto, modellazione della banda).
 - Contatori.
- Workflow: Quando un pacchetto arriva, lo switch cerca una corrispondenza nella sua tabella di flusso. Se non trova una corrispondenza (primo pacchetto di un flusso), chiede istruzioni al controllore, che risponde con una nuova regola. La regola viene memorizzata e applicata ai pacchetti successivi, riducendo la necessità di continua interazione con il controllore.

Precisamente, vediamo la seguente figura:



Vediamo che si ha una logica di supporto per il controllo che si occupa di interagire il livello di Controllo dell'SDN, attraverso questa lo switch riesce a comunicare con il controller. Quest'ultimo lo gestisce tramite protocollo OpenFlow.

Si hanno delle funzioni di Forwarding (Inoltro), accetta i dati che fuiscono dagli altri dispositivi e inoltra i dati in percorsi stabiliti tramite le regole definite dal controllore.

La tavola di inoltro sappiamo bene che indica il prossimo dispositivo da raggiungere sapendo la categoria di pacchetto. Esso può alterare i pacchetti prima di inoltrarli o anche scartarli.

Le code di input e di output servono per contenere i messaggi FIFO

Il flusso di pacchetti usa il protocollo IP.

Le tabelle di inoltro può definire entrate basate su campi del livello soprastante all'IP (UDP/TCP), di conseguenza il dispositivo esamina l'header IP e anche gli altri header per prendere decisioni.

OpenFlow

È un protocollo largamente supportato e sviluppato per le interfacce Southbound dell'SDN.

Le componenti chiavi sono:

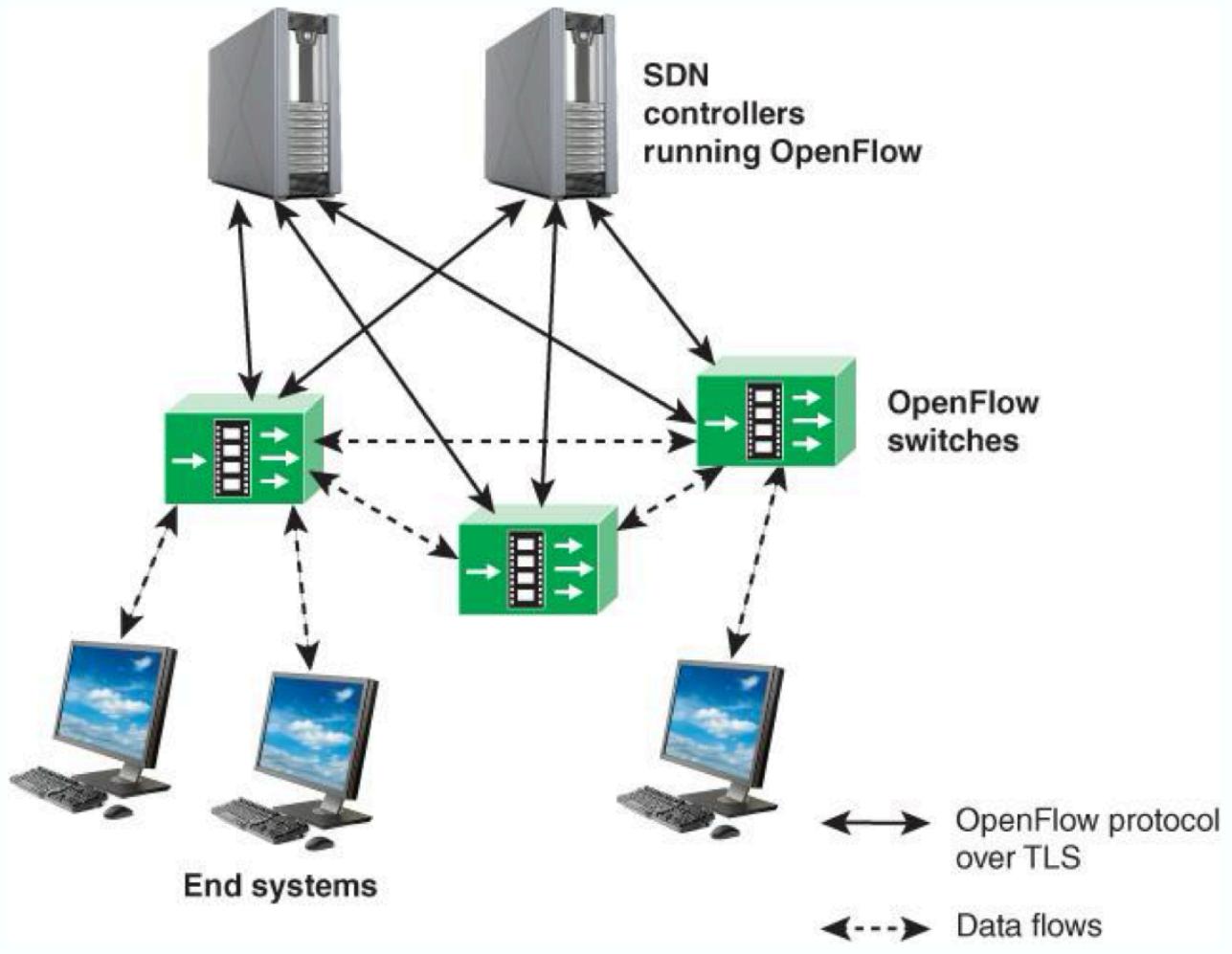
- OpenFlow-Enabled Devices: sono dispositivi come gli switch e i router che supportano il protocollo OpenFlow, hanno le tabelle di flusso per salvare le informazioni sul traffico della rete.
- OpenFlow Controller: rappresenta il controller centralizzato responsabile nel prendere decisioni sul gestire il traffico della rete. Comunica tramite protocollo OpenFlow.

Questo protocollo facilita le comunicazioni tra controller e dispositivi di rete, specifica la **struttura logica della funzionalità della rete di switch** definito nel OpenFlow Switch Specification tramite Open Networking Foundation [#ONF](#).

Termini Chiave:

- OpenFlow Switch: insieme di risorse openflow gestite come una singola entità, includendo un percorso dati e un canale di controllo. È connesso logicamente agli altri openflow switch tramite le porte openflow.
- OpenFlow Port: sono punti di uscita e di entrata per i pacchetti. Questi pacchetti sono inoltrati tramite queste porte.

- OpenFlow Channel: interfaccia tra switch e controller, usato per la gestione dello switch.

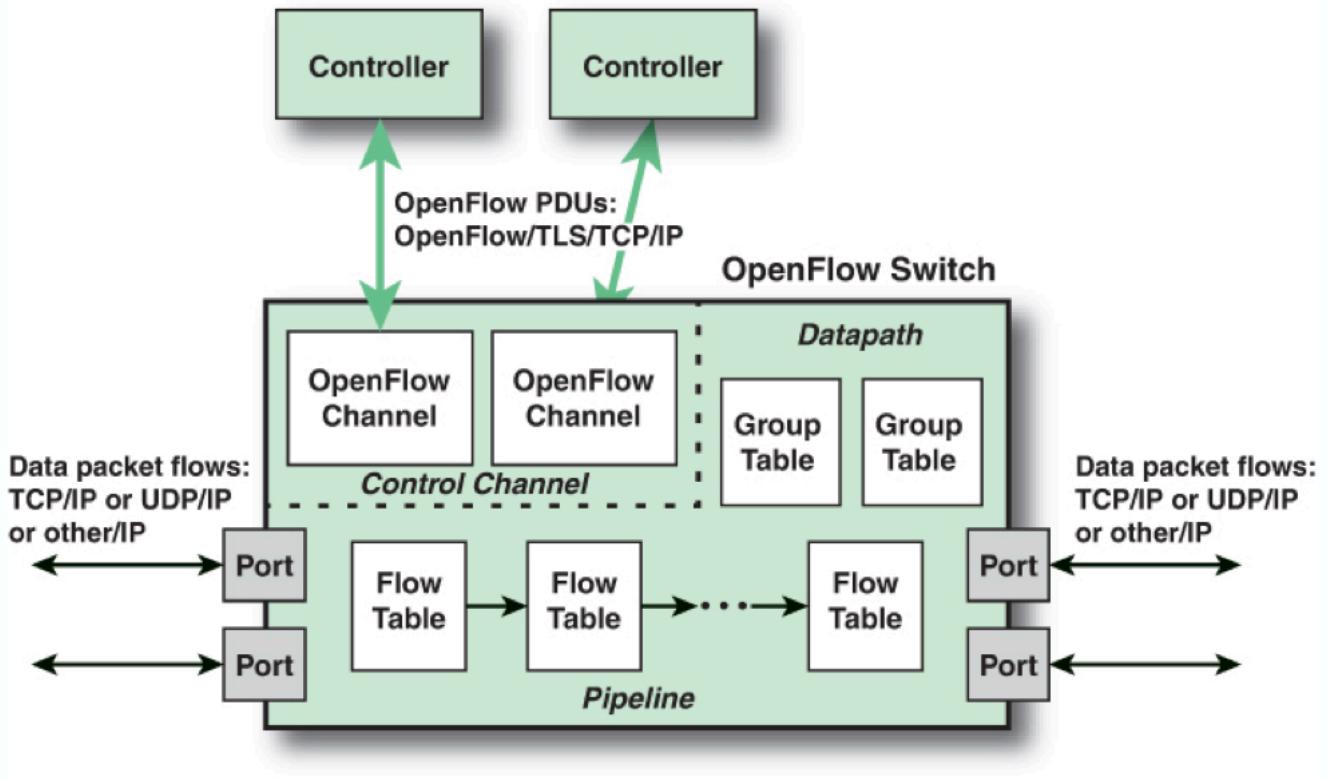


Esempio:

Un dispositivo ha tre porte:

- uno per la comunicazione con l'SDN controller.
- uno per i pacchetti in input.
- uno per i pacchetti in output.

Ma questi device possono avere anche multi porta sia per collegarsi con più controller che per il flusso di pacchetti.



L'SDN controller usa il protocollo OpenFlow sopra il Transport Layer Security per comunicare con OpenFlow-compatible switch.

Gli switch sono connessi tramite le porte OpenFlow.

Un esempio di flusso:

- tutti i pacchetti con la stessa sorgente e destinazione.
- tutti i pacchetti con la stessa identificazione VLAN.

Flow Tables

Ogni switch ha una tabella dove sono salvate tutte le regole di routing.

Le componenti sono le seguenti:

- Match Fields: identificano i pacchetti che matchano il flusso.
- Priorità: serve a valutare ordine di importanza in relazione con gli altri flussi.
- Contatore: numero di pacchetti matchati.
- Azione: istruzione per la gestione del pacchetto.
- Timeout:

- **Idle timeout:** se il flusso non riceve nessun pacchetto allora viene eliminato dalla tabella.
- **Hard timeout:** viene eliminato il flusso (o regola) dopo un certo tempo.
- Cookie: scelto dal controller per identificare un flusso.

Cosa fa il nostro switch?

Matcha i pacchetti usando una moltitudine di controlli nell'header del pacchetto.

Il processo è:

1. si controlla da che porta è arrivato il pacchetto. (Livello Fisico)
2. si controlla gli indirizzi MAC di sorgente e destinazione per operazioni di switch (switcha con l'indirizzo MAC del prossimo switch). (Livello MAC)
3. Controllare se il pacchetto è IPv4 e controllare indirizzo di destinazione. Valutare, prima di questo, se rispetta il frame Ethernet. Specificare il numero del protocollo IP.

OSI Layer	Protocol	Action	Header Field	Example Application
Layer 1	Output	Forward	Port ID	Drop, flood, or forward packet
	Queue	Set	Queue ID	Bandwidth shaping
Layer 2	Ethernet	Set	VLAN ID	Manipulate VLAN tags
Layer 3	IPv4	Set Src./Dst.	Network address translation	
Layer 3	IPv4	Decrement	TTL	Decrement Time-To-Live
Layer 4	TCP	Set	Port	Port address translation

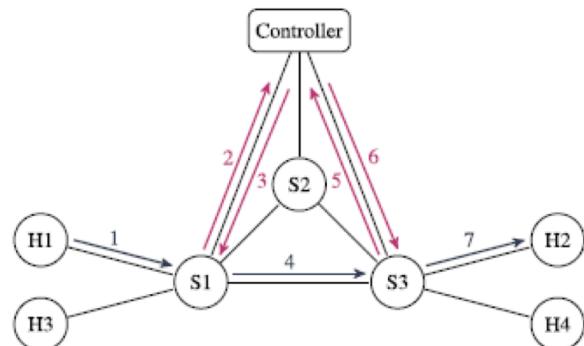
Il forwarding (flooding->tutte le porte).

Dropping: utilizzato dai firewall per permettere alcune connessioni e altre no.

Bandwidth Shaping: assicura un QoS inoltrando pacchetti in differenti code.

Inizializzazione:

- Step 1: Packet sent from host H1 to switch S1.
- Step 2: Switch S1 evaluates its flow table entries for a match.
 - If no match is found (first packet of the flow), the switch asks the controller for instructions.
- Step 3: Controller replies with a new rule (e.g., forward packets to switch S3).
- Step 4: Switch S1 stores the new rule in its flow table.
- Step 5: Rule is applied to all incoming packets from host H1 to switch S1.
- Step 6: Procedure is repeated for succeeding switches (e.g., switch S3).
- Step 7: Packets are sent to the desired destination (host H2).



Sembra che tutti i pacchetti vengano mandati al controller, causando inefficienza nel sistema ma in realtà una volta che la comunicazione iniziale configura l'insieme di regole e le salva nella tabella il controller non ha più molta utilità (a parte varie occasioni), gli switch cercano prima nella tabella (non è che fanno una richiesta in qualunque caso) e se la trovano utilizzano le azioni associate.

Piano di Controllo SDN

Serve a mappare le richieste di servizi a livello applicazione con specifici comandi da mandare agli switch del piano dati.

Da informazioni della topologia del piano dati e dettagli delle attività alle applicazioni.

Questo piano consiste in un insieme di server che cooperano(SDN Controllers).

Il piano di controllo traduce il servizio richiesto in comandi di rete, gestisce la topologia della rete e le informazioni riguardante essa. Direziona le operazioni del piano dati basato sui bisogni dell'applicazione.

NOS - Network Operative System

Provvede a servizi essenziali come API e elementi per gli sviluppatori, permettendo ad essi di definire delle politiche di rete e gestire le reti senza avere una conoscenza dettagliata delle caratteristiche dei dispositivi di rete.

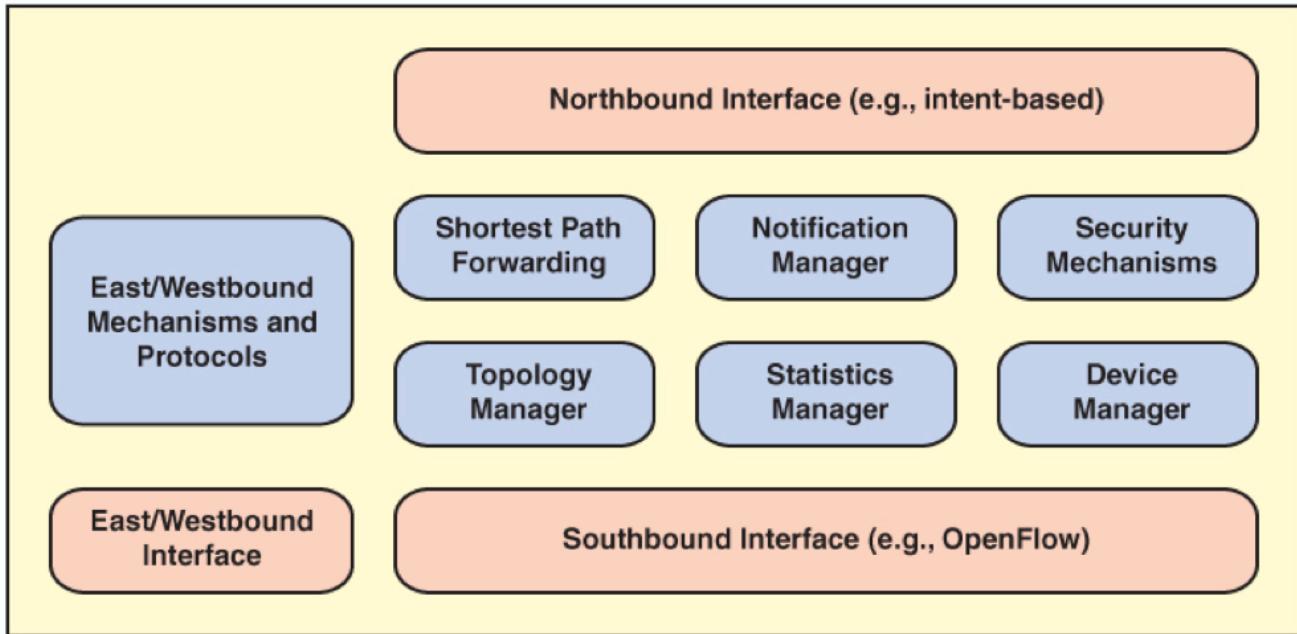
Northbound Interface

Offre un mezzo uniforme per gli sviluppatori di applicazioni e i gestori di rete per accedere ai servizi SDN ed eseguire attività di gestione della rete. Permette agli sviluppatori di creare software che sia in gran parte indipendente dai dettagli del piano dati e compatibile con vari server controller SDN. In questo modo, le applicazioni utilizzano una vista astratta della rete per prendere decisioni.

(Esempio -> RESTful API basate su HTTP)

Southbound Interface

Fornisce la connessione logica tra il controller SDN e gli switch del piano dati. Consente al controller di gestire e configurare gli switch in modo dinamico.



- Shortest Path Forwarding: usa informazioni di routing dagli switch per stabilire dei percorsi preferibili.
- Notification Manager: per gestire eventi come allarmi, notifiche, cambiamenti di stato e inoltrarli alle applicazioni.
- Security Mechanisms: per assicurare isolazione e rinforzo della sicurezza tra applicazioni e servizi.
- Topology Manager: mantiene e crea info su interconnessioni tra switch.
- Statistics Manager: colleziona dati sul traffico attraverso switch.
- Device Manager: configura parametri, attributi e tabelle di flusso degli switch.

Westbound Interface

Permette comunicazione tra il controller SDN e una rete non SDN.

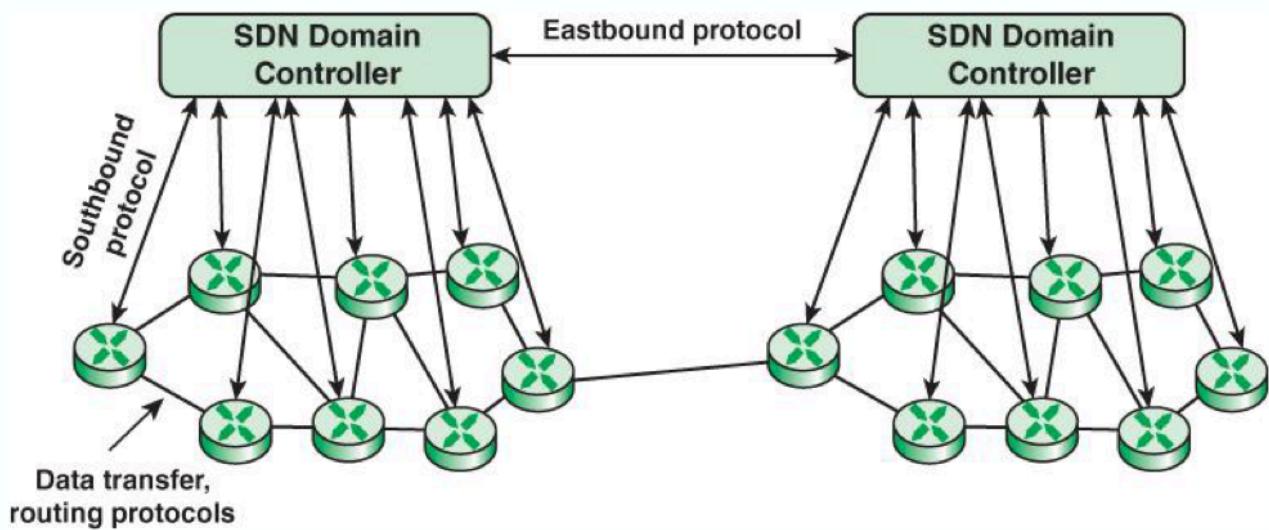
Viene usato il protocollo BGP(Border Gateway Protocol) per fare da ponte tra una rete tradizionale e una rete SDN.

Eastbound Interface

Permette lo scambio di informazioni tra controller distribuiti , supportando più SDN controller.

Rende il sistema scalabile.

Deployment incrementale: integra facilmente e molto flessibili con la possibilità di nuove infrastrutture.

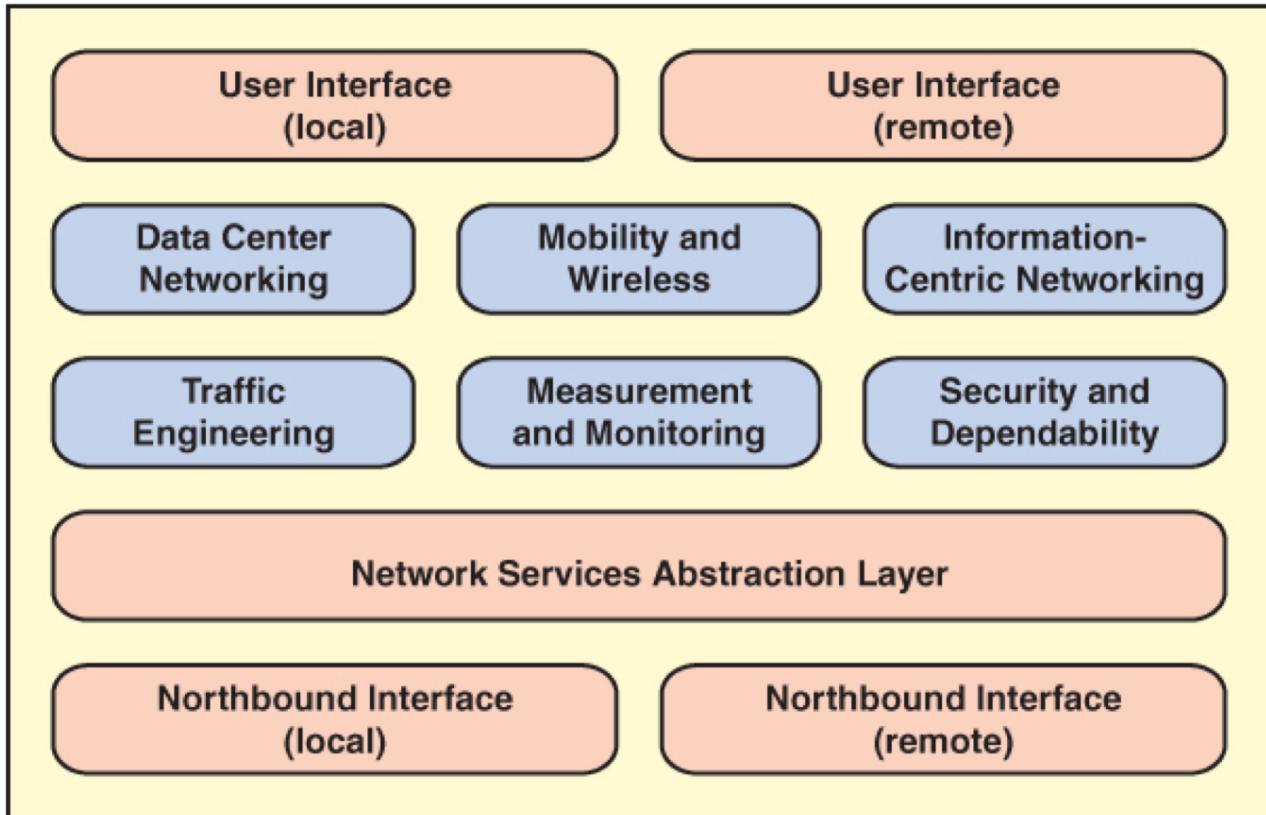


Piano Applicazione

Includere applicazioni per il controllo e per la gestione della rete, può includere dei tools e dei servizi per monitorare e controllare le risorse della rete.

Interagiscono con il piano di controllo SDN tramite interfaccia dell'applicazione.

Usa viste astratte di risorse di rete date da il livello di controllo SDN attraverso il modello di dati e le info.



La Northbound Interface serve ad accedere tramite app alle funzioni del piano controllo senza conoscere i dettagli sottostanti degli switch.(Viste Astratte).

Ci sono due tipi di interfacce (implementate similmente):

- Local: l'applicazione è eseguita nello stesso server del piano di controllo
- Remote: l'applicazione è eseguita su sistemi remoti , si utilizzano protocolli o API per connettere il controller NOS al server centrale. Vuol dire che l'applicazione **è eseguita su una macchina diversa** da quella dove gira il controller SDN, ma **comunica comunque con quel controller**, tramite API (es. REST, gRPC...).

Esempio:

Immagina di avere un **controller SDN** (es. OpenDaylight) che gira su un server a scuola. Se crei una app SDN che gira **sullo stesso server** del controller, usi una **local interface**:
 → ad esempio, un'app Java che usa direttamente librerie interne del controller.

Se invece crei un'app Python che gira **sul tuo portatile a casa** e interagisce con l'API REST del controller SDN via HTTP:
 → stai usando una **remote interface**.

Casi d'Uso SDN:

L'SDN abilita numerosi casi d'uso come:

- Pianificazione del traffico per la prevedibilità e la manutenzione (es. instradamento dinamico per aggirare guasti o manutenzione).
- Controllo dell'accesso alla rete con configurazione agile e controllo granulare per servizio.
- Service Function Chaining (SFC): reindirizzare il traffico attraverso funzioni di servizio virtualizzate (es. firewall, DPI).
- Handover utente dinamico (es. per spostare un servizio da un host all'altro man mano che un utente si muove tra stazioni base, riducendo latenza e carico di rete).

NFV (Network Functions Virtualization)

Il concetto di NFV è emerso nel 2012, spinto da operatori di rete che miravano a ridurre i CAPEX (spese in conto capitale) e gli OPEX (spese operative), aumentando la flessibilità della rete.

Principi e Vantaggi della NFV:

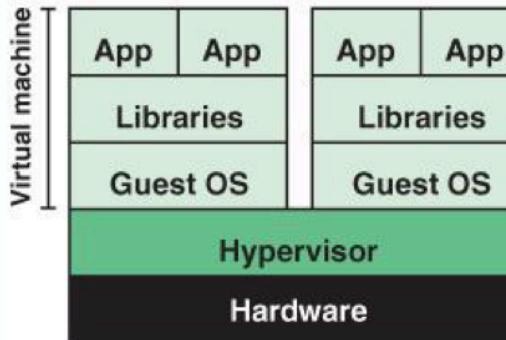
- Disaccoppiamento del software dall'hardware: Le funzioni di rete basate su software (Virtual Network Functions - VNF) sono separate dall'hardware fisico dedicato. Questo permette di farle girare su hardware commodity (general-purpose) e in ambienti cloud.
- Deployment flessibile e su richiesta: Le VNF possono essere distribuite dinamicamente, scalando le risorse in base alle esigenze attuali.
- Service Chaining: Le VNF, spesso modulari, possono essere concatenate per creare servizi di rete complessi.
- Virtualizzazione hardware: L'NFV si basa sulla virtualizzazione, utilizzando hypervisor (Type 1 o Type 2) per creare macchine virtuali (VM) o, più recentemente, container.... I container sono più leggeri e portabili rispetto alle VM, condividendo il kernel del sistema operativo host.

La virtualizzazione delle funzioni permette anche di avere, su una singola macchina, numerose applicazioni che possono essere eseguite su differenti sistemi operativi.

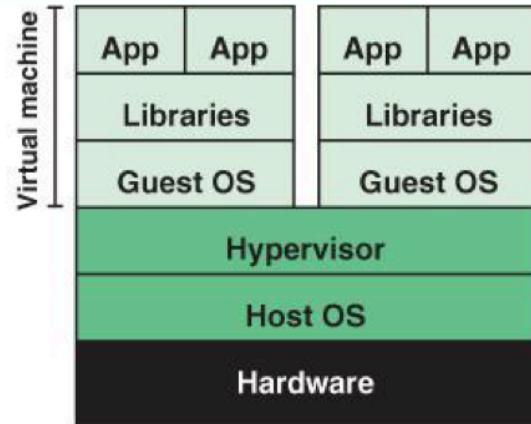
I sistemi operativi infatti possono supportare molte VM e ogni VM emula il suo OS. Alcune virtualizzazioni emulano le piattaforme hardware specifiche.

Tipologia di Hypervisor:

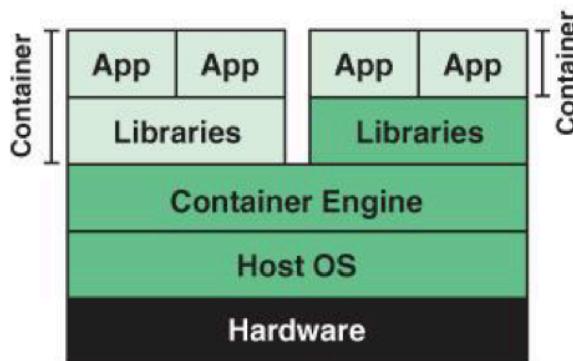
- Tipo 1: caricato dentro il server fisico come OS. Supporta VM come guest, ideale per insieme di host virtualizzati.
- Tipo 2: Esegue una OS, poi sopra ha un Hypervisor, meno performance poichè si ha competizione di risorse tra Hypervisor e OS



(a) Type 1 Hypervisor



(b) Type 2 Hypervisor



(c) Container

Poi si ha il Container, esso runna sul sistema operativo e provvede ad avere ambienti isolati per le applicazioni.

Il Container Engine è un piccolo livello di software che richiede all'OS le risorse per istanze isolate(singolo container).

I pro di questa tipologia di tecnica sono:

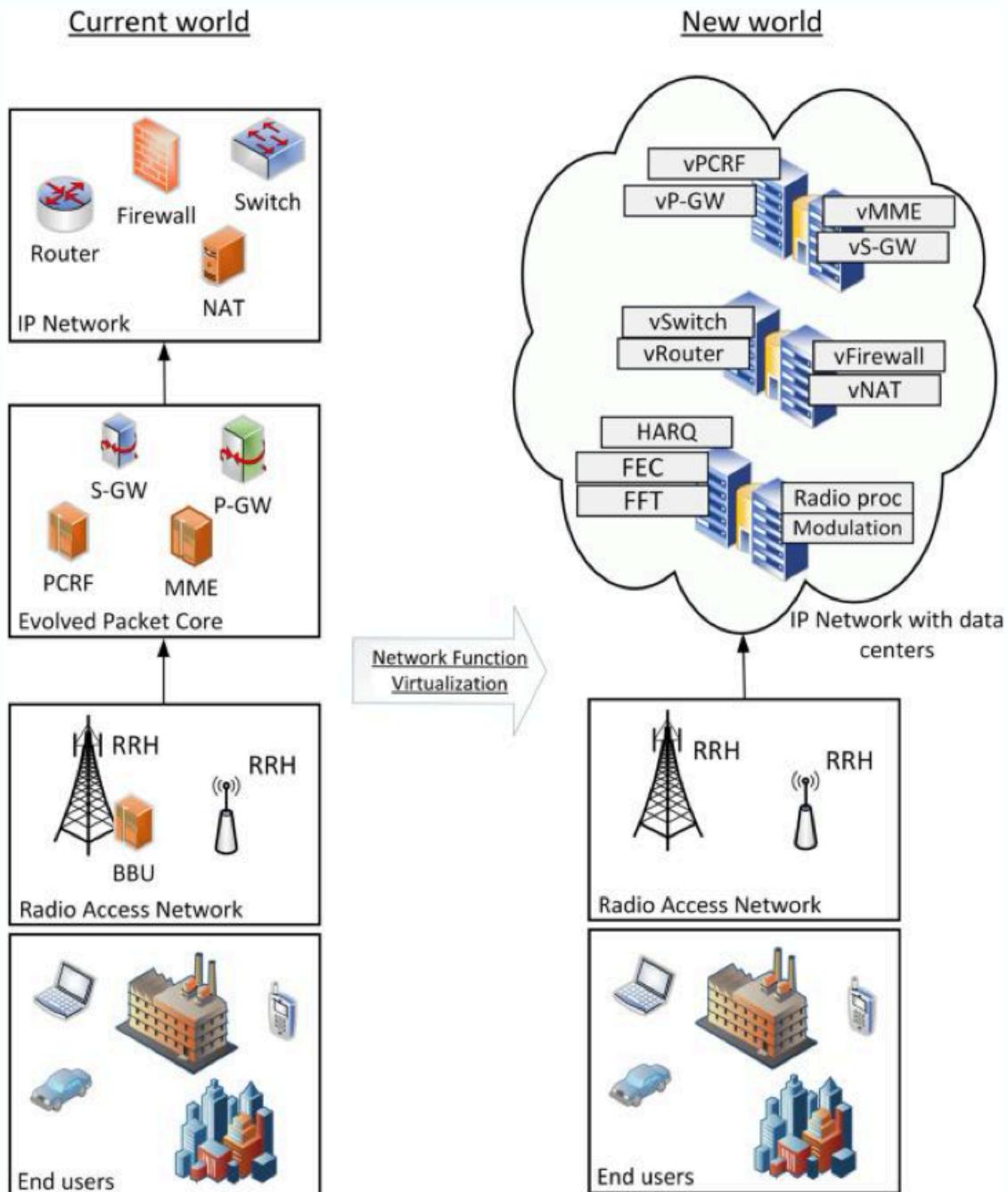
- molto più leggero, non si ha Host OS.
- La manutenzione e la creazione è molto più semplice.
- Si ha un movimento di applicazioni tra un sistema e l'altro molto più semplice.

NFV vs. Reti Cellulari 4G:

Nel contesto delle reti cellulari 4G, l'NFV consente che componenti chiave del core network (es. P-GW, S-GW, MME, PCRF) e parti della RAN (es. Baseband Unit - BBU) siano implementate come VNF.... Questo permette la softwarizzazione della BBU e la sua divisione in sottofunzioni eseguibili da remoto, come nella Cloud RAN (C-RAN), che centralizza l'elaborazione della banda base nei data center, trasformando le stazioni base in Remote Radio Heads (RRH).

Infatti, le reti cellulari 4G, sono fatte in questo modo:

- RAN (Radio Access Network): gli utenti si collegano alle BS (Base Station) che includono RRH (Remote Radio Head) che è un equipaggiamento radio e poi si ha il BBU (Baseband Unit) - hardware che processa il segnale di banda.
- Poi si entra nel CORE Network: ha dispositivi chiave come il P-GW, S-GW, MME, PCRF che sono hardwaree.
- IP Network: che ha Firewall, Router , NAT e Switch.

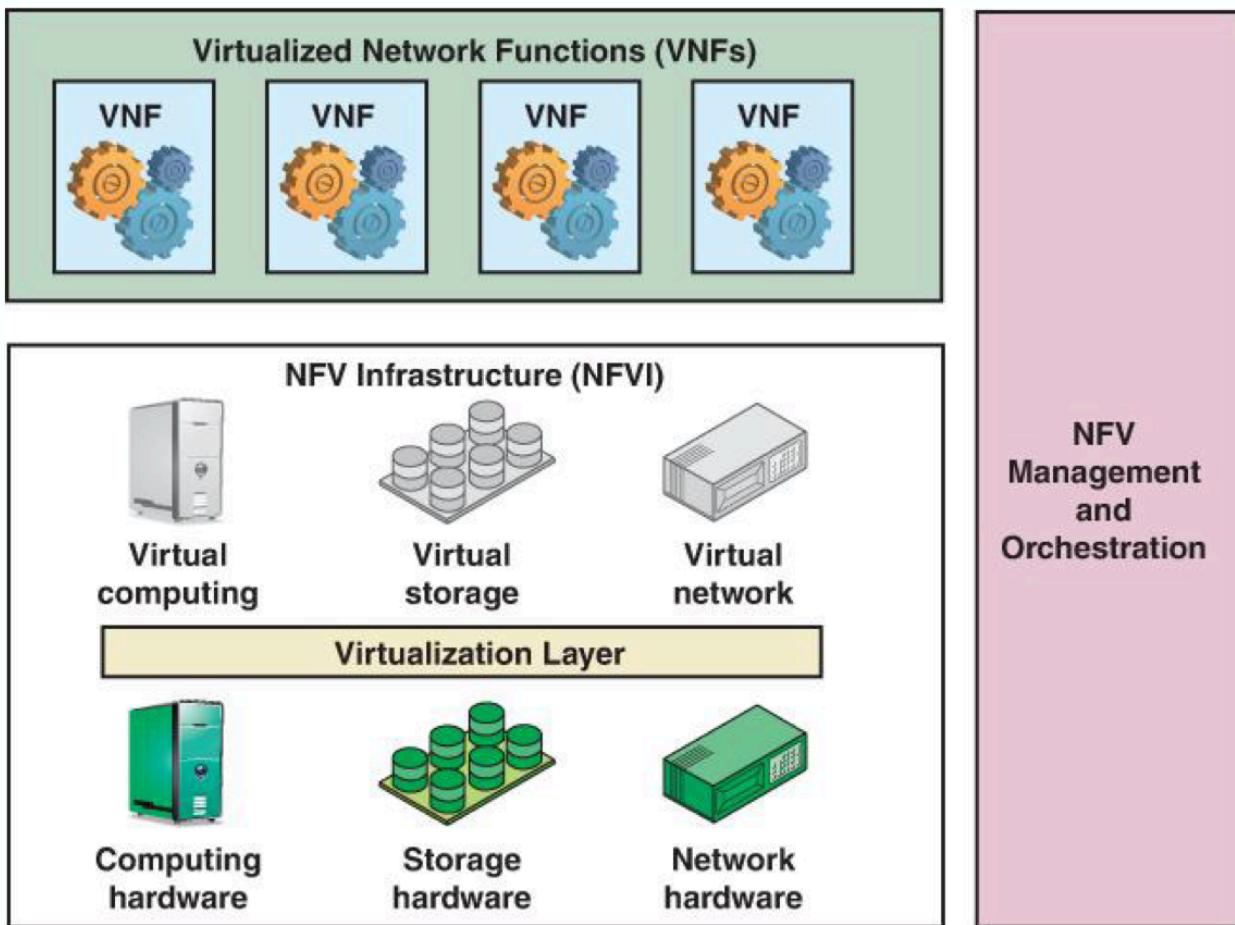


Quello che si vuole è softwarizzare tutte queste funzioni, come anche quella della BBU, rappresentandole come sotto funzioni che possono essere richiamate da remoto.

Framework NFV (NFV-MANO):

Il framework NFV, definito dall'ETSI ISG NFV, comprende:

- NFV Infrastructure (NFVI): L'ambiente hardware e software che ospita le VNF (risorse di calcolo, storage e rete virtualizzate).
- Virtualized Network Functions (VNFs) / Element Management Systems (EMS): Le VNF implementate come software e i sistemi per gestirle che sono eseguite su NFVI.
- NFV Management and Orchestration (NFV-MANO): Il cuore della gestione, responsabile dell'orchestrazione e del ciclo di vita delle risorse fisiche e software. Include l'NFV Orchestrator (installazione, configurazione, gestione ciclo di vita dei servizi di rete e VNF), il VNF Manager (gestione ciclo di vita delle istanze VNF) e il Virtualized Infrastructure Manager (VIM).



Descriviamo dettagliatamente i quattro maggiori blocchi:

- NFVI (Network Function Virtualisation Infrastructure): comprende risorse hardware e software per creare ambiente per VNF.
- VNF/EMS: sono una collezione di VNFs implementate nel software e eseguite su risorse virtuali. EMS (Element Management System) è un elemento che permette di gestire le VNF.

- NFV- MANO: è un framework che serve a maneggiare, orchestrare e inizializzare tutte le risorse nell'ambiente NFV. Si occupa anche di calcolare , networking, salvare e usare le risorse VM.
- OSS/BSS: sistema di supporto operazionale e business implementato dal provider VNF. Interfaccia da dove viene richiesto il servizio.

In generale, quindi, ci sono tre livelli:

- uno che si occupa di gestire e provvedere alle risorse virtuali per l'ambiente che stanno su delle risorse fisiche.(NFVI e Virtualize Infrastructure Manager - in rosa).
- uno si occupa della implementazione software delle funzioni di rete tramite VNF manager e EMS (in verde).
- E una parte che consiste da OSS/BSS e NFV Orchestrator, che trattano di orchestrare, gestire e controllare tutti gli altri piani precedenti.

NFV MANO

Questa parte importante dell'NFV controlla la gestione e l'implementazione delle funzioni che softwarizzano hardware della rete.

È formata da tre parti:

- NFV Orchestrator: è un livello che installa e configura la nuova rete di servizi e i pacchetti VNF. Gestisce la vita di un Network Service. Gestisce risorse globali. Valida e autorizza richieste di risorse nell'NFVI.
- VNF Manager: tiene d'occhio il ciclo di vita delle istanze VNF.
- Virtualized Infrastructure Manager: controlla, gestisce le interazioni tra VNF con calcolo, memoria e risorse network. Gestisce la virtualizzazione di queste risorse.

NFVI

È il cuore dell'architettura NFV, consiste in risorse e funzioni tra tre domini:

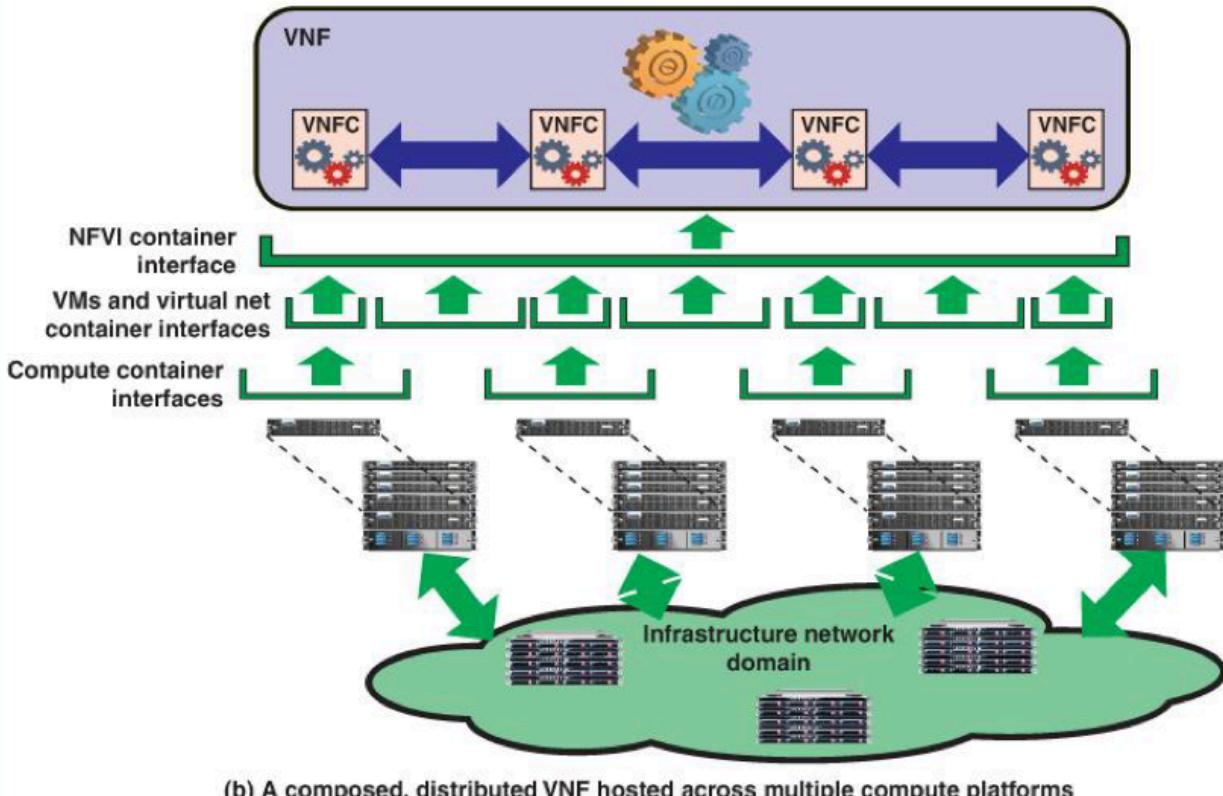
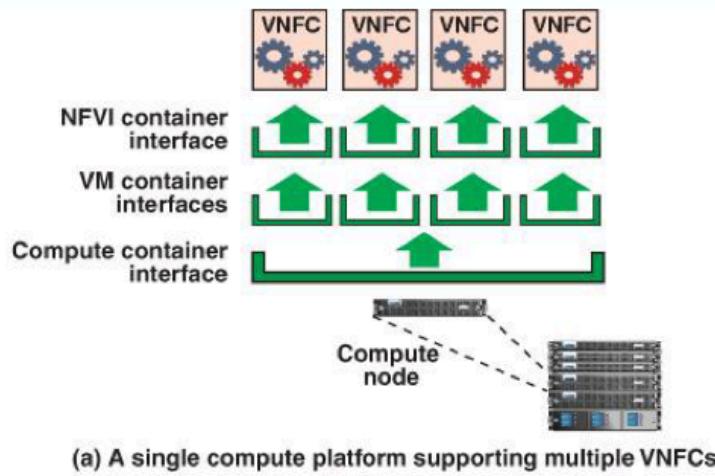
- Dominio del Calcolo(Compute Domain): provvede ad un alto volume di server e memoria.
- Dominio Hypervisor: media le risorse del Compute Domain al dominio delle virtual machine, provvede alla astrazione dell'hardware.
- Dominio Infrastructure Network: configurato per fornire servizi, comprende switch di grande volume.

Principi Chiave NFV:

- Service Chaining: le VNF sono modulari e hanno funzionalità limitata, quello che può fare i service provider è far passare il flusso di dati tra VNF multiple per fornire un altro servizio

desiderato.

- MANO: permette di gestire il ciclo di vita di queste istanze VNF, gestisce infrastrutture NFV e permette creazione di VNF, service chaining, monitoraggio, riallocazione, rimozione VNF.
- Architetture Distribuite: una VNF può essere formata da una o più componenti VNFC, ognuna che implementa un sottoinsieme di funzionalità VNF. Queste componenti possono essere hostati su host distribuiti. (Scalabilità e Ridondanza).



Si possono avere due tipi di sviluppi dei contenitori per NFVI:

- Un singolo Compute Node supporta più VNFC.
Hosta multiple VM e ognuno di esse forma una VNF. Una singola VNF può essere virtualizzata da una singola VNFC ma multiple VNFC possono essere combinate per fare una VNF.
- Si ha una infrastruttura network formata da più server che hanno diversi compute node. Alcuni di essi ospiteranno delle VM che contengono VNFC tale che tutte insieme formino una VNF. I nodi sono interconnessi dall'host network formando un dominio di infrastruttura di rete.

SDN e NFV: Tecnologie Complementari

Sebbene SDN e NFV siano sviluppati da enti di standardizzazione diversi, il loro uso coordinato offre vantaggi significativi, specialmente nelle reti 5G.

- SDN come abilitatore di NFV: L'SDN semplifica la configurazione e la gestione delle reti con VNF, assicurando che siano connesse correttamente e che soddisfino i requisiti di QoS. Permette una configurazione dinamica e riduce l'intervento manuale.
- Ruoli complementari: L'SDN si integra nel concetto più ampio di controllore di rete nell'NFVI, orchestrando risorse fisiche e virtuali. Un controllore SDN può anche essere eseguito come VNF stesso. Questa sinergia fornisce un approccio unificato basato su software per controllare l'attrezzatura e le risorse di rete, migliorando efficienza, programmabilità e flessibilità per soddisfare le esigenze dinamiche del 5G60.

Il documento conclude sottolineando l'evoluzione della virtualizzazione dello stack di protocollo e come Cloud RAN, SDN e NFV stiano trasformando l'architettura delle reti verso un modello più centralizzato e flessibile.

Cloud RAN vs Legacy 4G RAN

Legacy 4G RAN:

- Componenti: Una Legacy 4G RAN è composta principalmente da Base Station (BS), a cui gli utenti finali si connettono in modalità wireless. Ogni BS include una Remote Radio Head (RRH), che gestisce l'apparecchiatura radio/antenna, e una Baseband Unit (BBU).
- Posizionamento della BBU: Le BBUs sono tipicamente situate in ogni sito radio (vicino alle antenne). Sono connesse alle RRH tramite fibre che supportano lo standard CPRI.
- Elaborazione della banda base: L'elaborazione della banda base (come la conversione analogico-digitale, l'elaborazione del segnale fisico, la modulazione/demodulazione, la correzione degli errori, la pianificazione radio, la crittografia/decrittografia del PDPCP e la modulazione multi-carrier) è gestita da hardware dedicato (ASIC, DSP, FPGA) all'interno della BBU in loco.

- Flessibilità: L'architettura è più rigida e meno flessibile per la gestione e la configurazione della rete, poiché le funzionalità sono strettamente legate all'hardware fisico.

Cloud RAN:

- Virtualizzazione: Il Cloud RAN virtualizza le unità di banda base (v-BBUs), trasformando le tradizionali Base Station (BS) in sole Remote Radio Head (RRH). Questo significa che l'hardware dedicato viene sostituito da funzioni software.
- Centralizzazione dell'elaborazione: L'elaborazione della banda base viene centralizzata in data center o in strutture di elaborazione condivise.
- Flessibilità e Scalabilità: Offre una maggiore flessibilità nella gestione e configurazione della rete e una scalabilità migliorata per gestire scenari 5G ampi ed eterogenei. Poiché le funzioni BBU sono virtualizzate, possono essere facilmente allocate, ridimensionate o spostate.
- Splits della BBU: L'architettura Cloud RAN consente di suddividere le funzioni della BBU in diversi punti dello stack di protocollo (ad esempio, Split A o Split E). Split più "bassi" (più vicini al livello fisico) richiedono throughput più elevati e latenze inferiori.

Vantaggi:

- Utilizzo efficiente delle risorse: Centralizzando le risorse, il Cloud RAN può ottimizzare l'utilizzo dell'hardware e ridurre i costi operativi.
- Aggiornamenti semplificati: Gli aggiornamenti e le modifiche alle funzioni di rete possono essere implementati tramite software, riducendo la necessità di sostituire l'hardware fisico.
- Integrazione con NFV: Sfrutta la Network Function Virtualization (NFV) per disaccoppiare le funzioni di rete dal loro hardware sottostante, consentendo l'implementazione di queste funzioni come software su hardware standard.