Temă de casă - Rețele de calculatoare și Internet

SDN - Software-Defined Networking

Profesor Coordonator:

Conf.dr.ing Ștefan Stăncescu

Aflorei Victor

Master IISC

Cuprins

1. SDN - Prezentare generală 3
1.1. Descriere generală 3
1.2. Definire concisă a elementelor arhitecturale 6
2. Principii și componente arhitecturale 7
2.1 Principii 7
2.2. Data plane 9
2.3. Controller plane 10
2.3.1. Prezentare generală 10
2.3.2. Controller-ul SDN 11
2.3.4. Delegarea controlului 13
2.3.5 Resursele partajate 13
2.3.6. Domenii administrative multiple 13
2.3.7. Coordonarea controller-controller 15
2.4. Application plane 16
2.5. Virtualizare 17
2.6. Funcția de management 17
2.7. Modelul informațional 18
3. Concluzii 18
4. Bibliografie 19

1. SDN - Prezentare generală

1.1. Descriere generală

Scopul SDN este de a oferi interfețe deschise care permit dezvoltarea de software ce poate controla conectivitatea oferită de un set de resurse ale rețelei cât și fluxul de trafic din rețea, împreună cu posibilitatea de a analiza și de a modifica traficul din rețeaua respectivă. Aceste funcții primitive pot fi abstractizate în anumite servicii de rețea arbitrare, dintre care unele ar putea să nu fie prezente în mod evident.

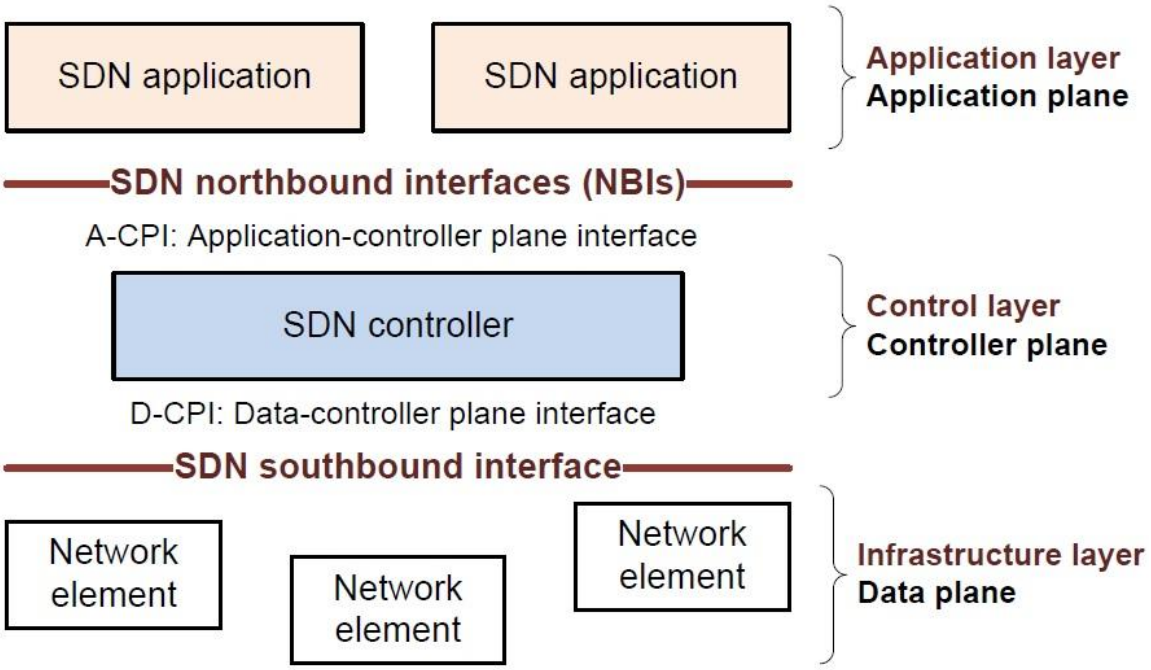

Figura 1.1. Componentele de bază ale SDN

Figura 1.1. prezintă componentele de bază SDN, cu terminologia similară cu cea prezentată de ONF - Open Networking Foundation - "Software-Defined Networking: The New Norm for Networks" [1]. Perspectiva inițială (cea definită de ONF) a cuprins straturile (layer) de infrastructură, control și de aplicație, care sunt desemnate în acestă figură arhitecturală ca fiind data plane, controller plane și application plane. Stratul de infrastructură (data plane) cuprinde elemente de rețea, care își expun capabilitățile spre stratul de control (controller plane) prin intermediul unor interfețe numite **SDN southbound interface**. Aplicațiile SDN se găsesc în stratul aplicație și transmit cerințele lor de rețea spre controller plane prin intermediul unor interfețe numite **SDN northbound interface** (adesea aceste interfețe se mai numesc NBI). În mijloc, controller-ul SDN traduce cerințele aplicațiilor și exercită un control de nivel scăzut asupra elementelor din rețea, în timp ce furnizează informații relevante spre aplicațiile SDN.

Modificările de terminologie reflecta faptul că unele aspecte cu privere la control se regăsesc în mod inevitabil în toate straturile, dar interfața de interes este aceea dintre controller-ul SDN și entitățile sale adiacente. Cele mai importante componente orizontale sunt numite planuri (plane) pentru a evita confuzia cu termenul strat (layer), care este folosit cu sensul de strat de rețea, de exemplu atunci când pachetele sunt mapate la nivel de MPLS, în continuare în Ethernet și mai departe în lungimi de undă.

Figura 1.2 adoptă terminologia revizuită și adaugă funcția de management, care este adesea omisă în diverese reprezentări SDN simplificate. Cu toate că multe dintre funcțiile de management tradiționale pot fi evitate cu ajutorul interfaței directe application-controller (A-CPI) anumite funcții de management sunt încă esențiale. În planul de date, funcția de management este cel puțin necesară pentru configurarea inițială a elementelor din rețea, pentru asignarea componentelor controlate SDN și pentru configurarea controller-ului SND. În planul de date, funcția de mnagementul trebuie să configureze politicile care definesc sfera de control oferită aplicației SDN și să monitorize performanța sistemului. În planul aplicație, funcția de managementul configurează în mod obișnuit contractele și acordurile la nivel de serviciu (service level agreements - SLA). În toate planurile, funcția de management configurează politicile de securitate care permit funcțiilor distribuite intercomunicarea în condiții de siguranță.

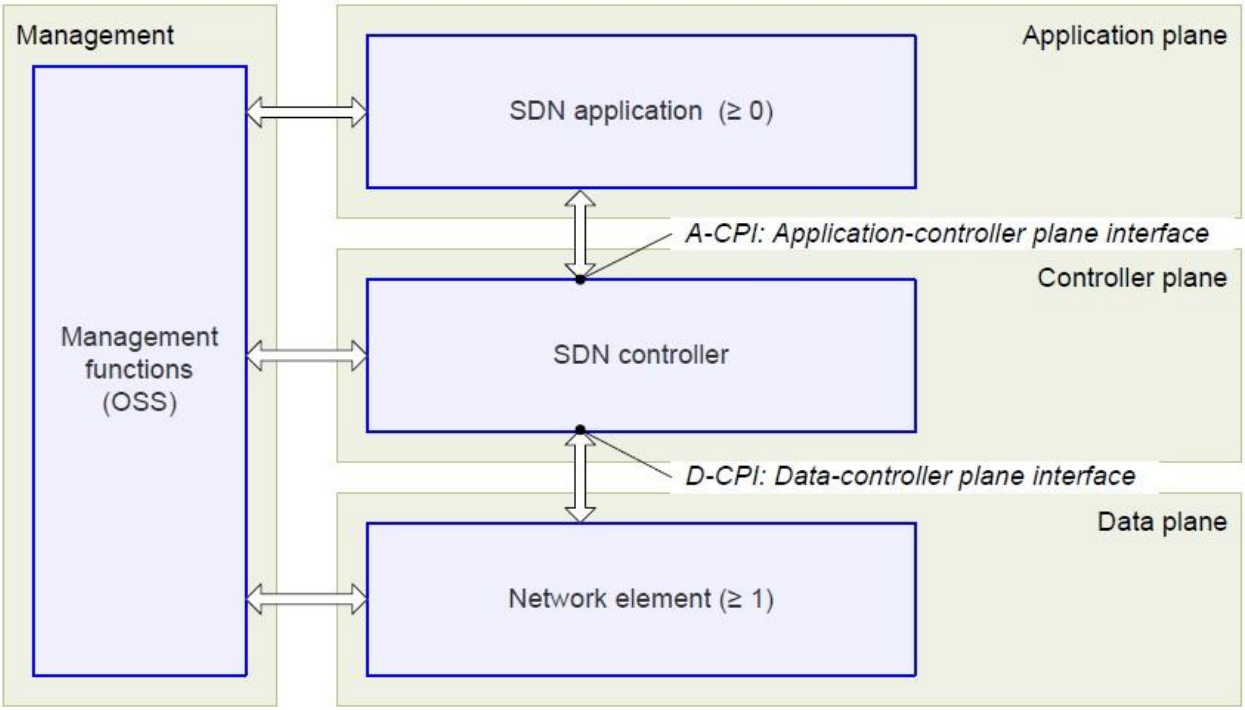

Figura 1.2. Componentele SDN și funcția de management

Figura 1.2. rezumă arhitectura SDN, cu punctele de terminologie și de referință utilizate în continuarea. Figura evidențiază planele distincte pentru aplicație, controller și date, împreună cu interfețele controller plane (CPI) desemnate ca puncte de referință între controller-ul SDN și planul aplicație (A-CPI) și între controller-ul SDN și planul de date (D-CPI). Informația schimbată prin intermediul acestor interfețe trebuie să fie modelată ca o instanță a unui model informațional de protocol neutru.

Figura 1.2 permite doar un singur domeniu de încredere. Figura 1.3. extinde ideea de a arăta mai multe domenii de încredere. Fiecare domeniu de încredere este înțeles ca având propria funție de management. Domeniile de încredere se poate extinde în mod logic în componente ale altor domenii de încredere, așa cum este exemplificat de către agenții verzi și roșii în controller-ul SDN albastru.

Figura 1.3, prezintă, de asemenea, agenții și coordonatorii din controlerul SDN și din elementele de rețea. Agenții sprijină conceptul de partajare sau virtualizare a resurselor de bază, de exemplu, care porturile ale elementului de rețea sunt controlate SDN (spre deosebire de porturi hibride sau moștenite) sau detaliile rețelei

virtuale care sunt expuse aplicațiilor SDN, în timp ce permite izolarea serviciilor unui de de serviciile altui client. În controller-ul SDN, diferiți agenți pot expune controlul asupra rețelei la diferite niveluri de abstractizare (latitudini) sau seturi de funcții (longitudini). Sarcina logică a controller-ului SDN este de a mapa și de a arbitra între cerințele de rețea de la toate aplicațiile SDN și traducerea lor în instrucțiuni pentru resursele elementului de rețea (NE) expuse prin intermediul agenților NE. Coordonatorii atât în elementul de rețea cât și în controller-ul SDN instalează resurse specifice clientului si politici primite de la funcția de management.

Mai mulți agenți se pot găsi în același timp, în orice element de rețea sau în controller-ul SDN, dar există o singură interfață de management logică și prin urmare, doar un singur coordonator pentru fiecare element de rețea sau pentru un controlor SDN.

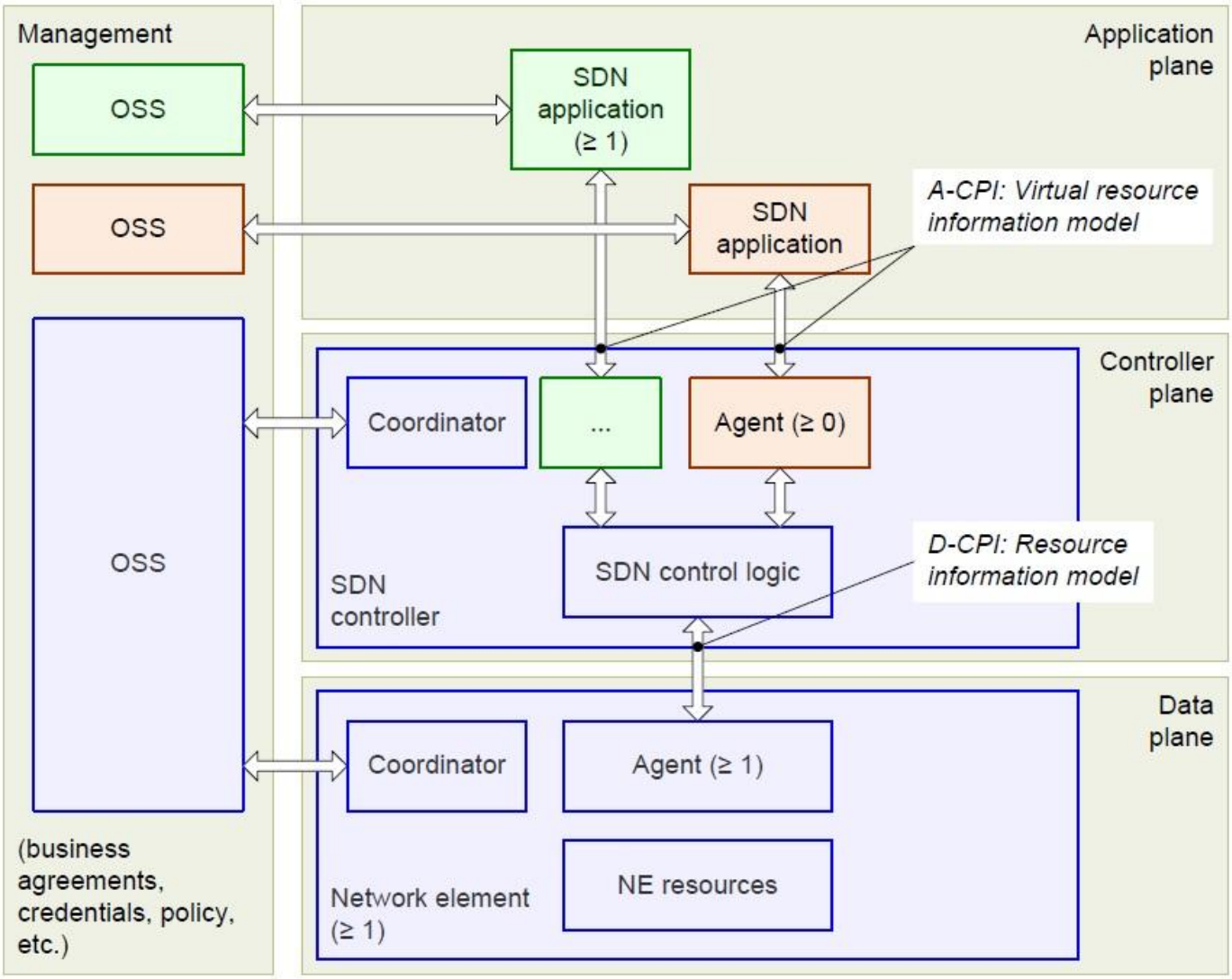

Figura 1.3. Privire de ansamblu a arhitecturii SDN

1.2. Definire concisă a elementelor arhitecturale

Figura 1.4. prezintă componentele principale și interfețele arhitecturii SDN.

Data plane

Planul de date cuprinde un set format din unul sau mai multe elemente de rețea, fiecare dintre ele conținând un set de resursele pentru forwardarea și pentru procesarea traficului. Resursele sunt întotdeauna abstractizări ale capabilităților elementelor sau entitățile de nivel inferior.

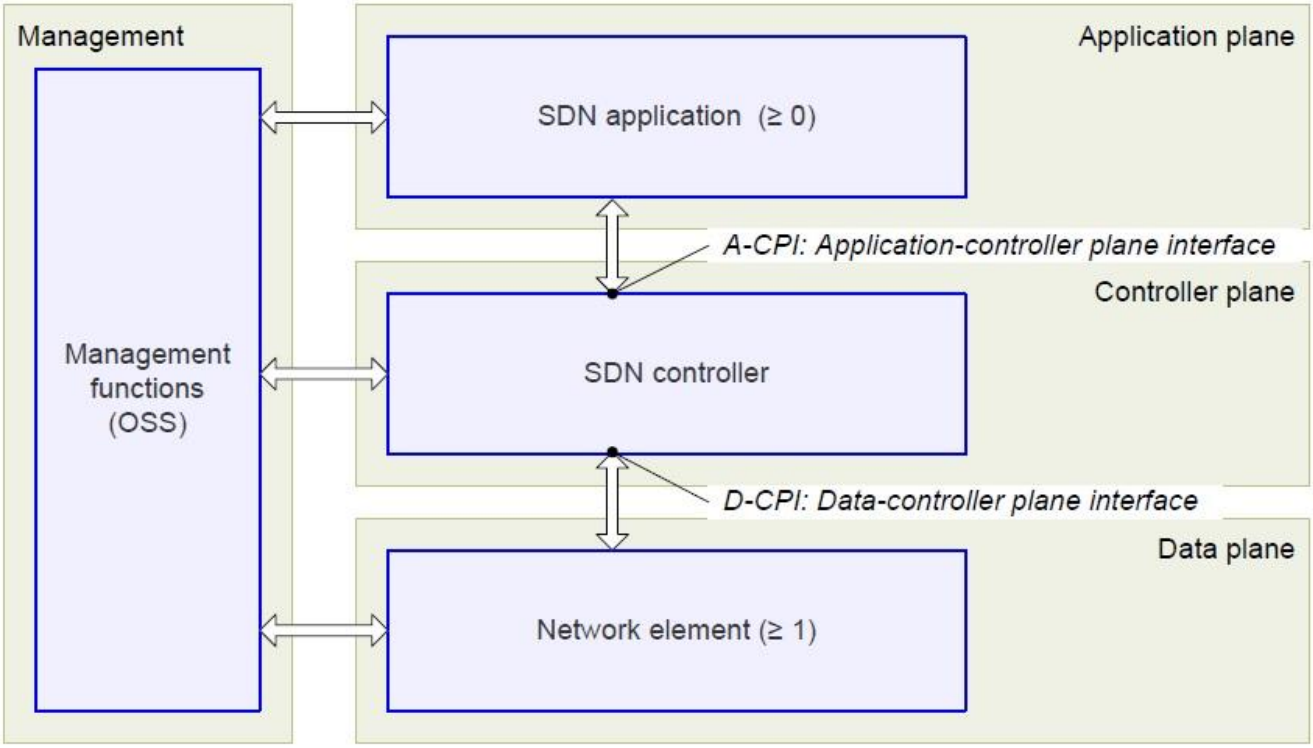

Figura 1.4. Arhitectura SDN

Controller plane

Controller plane cuprinde un set de controllere SDN, dintre care fiecare are controlul exclusiv asupra unui set de resurse expuse de către unul sau mai multe elemente de rețea din planul de date.

Funcționalitatea minimă a controller-ului SDN este de a executa cu fidelitate solicitările aplicațiilor pe care le suportă, în timp ce izolează fiecare aplicație de toate celelalte.

Pentru a efectua această funcție, un controller SDN poate comunica cu controllere SDN pereche, controllere SDN subordonate, sau medii non-SDN, după cum este necesar.

O funcție comună, dar neesențială a unui controler SDN este de a acționa ca element de control într-o buclă de feedback, răspunzând la evenimente din rețea pentru recuperarea de la eșec, reoptimizând alocarea de resurse sau printr-o altă modalitate.

Application plane

Application plane cuprinde una sau mai multe aplicații, dintre care fiecare are controlul exclusiv asupra unui set de resurse expuse de către unul sau mai multe controllere SDN.

O aplicație poate invoca sau colabora cu alte aplicații. O aplicație poate acționa ca un controller SDN în domeniul propriu.

Management-ul

Fiecare aplicație, fiecare controller SDN și fiecare element de rețea are o interfață funcțională către un manager.

Funcționalitatea minimă a managerului este de a aloca resurse dintr-un pool de resurse localizate într-un plan inferior la o anumită entitate client într-un plan superior, precum și de a stabili accesibilitatea informațiilor pentru a permite entităților de pe planele inferioare și superioare să comunice reciproc.

Administrarea

Fiecare entitate într-o tranziție nord-sud prin toate planele poate aparține unui domeniu administrativ diferit. Managerul trebuie să se afle în același domeniu administrativ ca și entitățile pe care le administrează.

Protocoalele ONF

Protocolul OF-config este utilizat pentru a efectua unele dintre funcțiile care sunt necesare interfaței de management.

2. Principii și componente arhitecturale

2.1 Principii

Principiile de bază ale SDN sunt:

- Decuplarea planului controller-ului și planului de date

Acest principiu solicită plane separate pentru controller și pentru date. Cu toate acestea, se înțelege că controlul trebuie neapărat exercitat în cadrul planului de date a sistemelor. Interfața D-CPI dintre SDN controller și elementul de rețea este definită în așa fel încât controlerul SDN poate delega funcționalitate semnificativă elementului de rețea (NE - network element) rămânând în același timp conștient de starea NE.

- Controlul centralizat din punct de vedere logic

În comparație cu controlul local, un controller centralizat are o perspectivă mai largă a resurselor aflate sub controlul său și poate lua decizii potential mai bune cu privire la modul lor de alocare. Scalabilitatea este imbunătățită atât prin decuplarea cât și prin centralizarea controlului, permițând priviri mai globale dar mai puțin detaliate a resurselor din rețea. Controllere SDN pot fi stivuite în mod recursiv pentru a oferi scalabilitate sau din motive de încredere cu privire frontierele rețelei.

- Expunerea abstractizată a resurselor rețelei și statusul aplicațiilor externe

Aplicațiile pot exista la orice nivel de abstractizare sau granularitate, atribute descrise adesea ca diferite latitudini, cu ideea că orientarea spre nord sugerează un grad mai mare de abstractizare. Pentru ca o interfață care expune resursele și status-ul unui anumit element poate fi considerată o interfață controller, distincția dintre aplicație și control nu este precisă. Aceeași interfață funcțională poate fi vizualizată din perspective diferite de anumite părți. La fel ca controllerele, aplicațiile se pot referi la alte aplicații ca fiind aplicații egale sau în calitate de clienți și servere.

Modelul de nivel înalt a tuturor interfețelor verticale din arhitectura SDN este expunerea unei instanțe a unui model informațional - de la server la client, pe care clientul poate efectua operații create-read-update-delete (CRUD) și operații specifice clasei. Din acest punct de vedere, funcția de management este responsabilă de instantierea modelelor de informații și a politicilor care definesc capacitățile expuse peste interfețele dintre plane, mai ales dincolo de granițele domeniului de încredere. Figura 2.1. ilustrează ideea că modelul client-server (sau controller-agent) este aplicabilă indiferent de câte nivele ierarhice ale controller-ului SDN există.

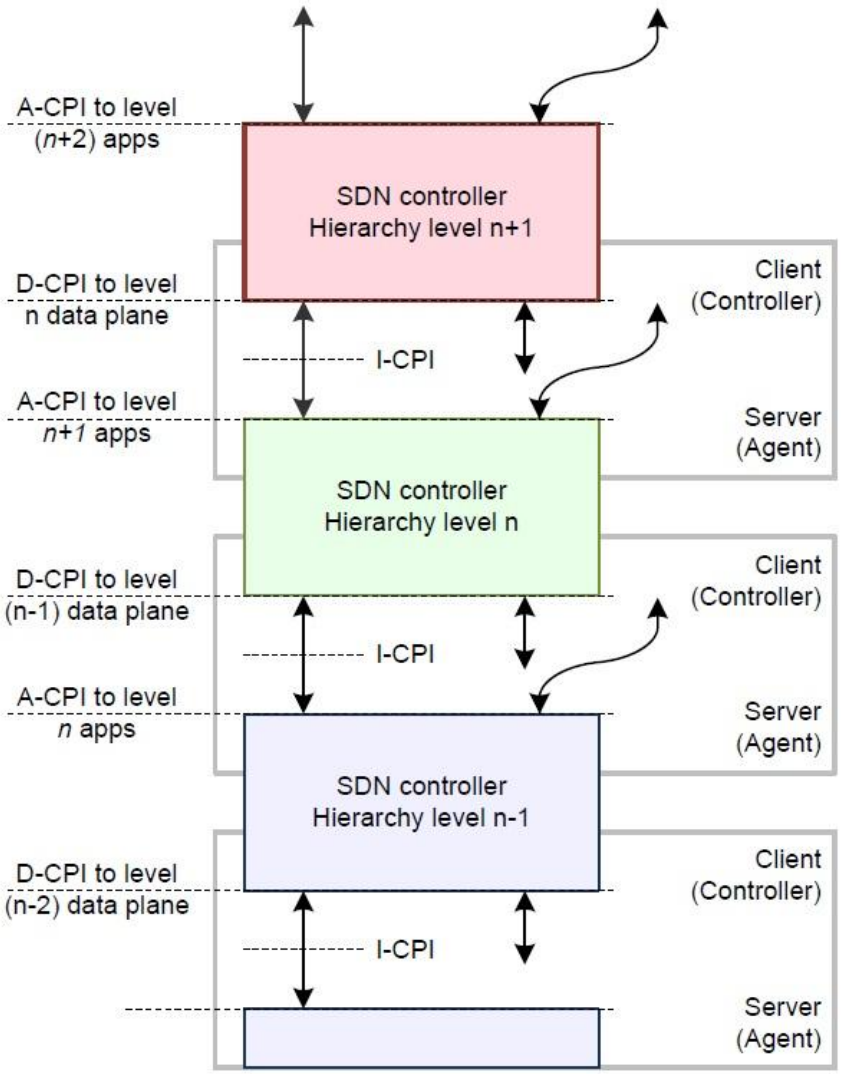

Figura 2.1. Rolurile ierarhice recursive

Nivelele ierarhice au două scopuri:

- Scalabilitate și modularitate: fiecare nivel succesiv superior are potențial pentru un nivel de abstractizare mai mare și un domeniu de aplicare mai larg.
- Securitate: fiecare nivel pot exista într-un domeniu de încredere diferit. Interfața de nivel reprezintă un punct de referință standard pentru securitatea inter-domeniu.

2.2. Data plane

Data plane include resursele care interacționează direct cu traficul clienților, împreună cu resursele de sprijin necesare pentru a asigura buna virtualizare, conectivitate, securitate, disponibilitate și calitate. Figura 2.1 extinde privirea asupra resurselor NE prezentate în figura 1.3.. Blocul de resurse NE cuprinde surse de date, motoare de forwardare și/sau de procesare a traficului, precum și un Virtualizer a cărui funcție este de a abstractiza resursele pentru controller-ul SDN și de a aplica politici. Această expansiune a detaliilor introduce o bază de date master de resurse (RDB) ce reprezintă repozitoriul conceptual a tuturor resurselor informaționale cunoscute la nivelul elementului de rețea.

Software-defined networking conține funcții de expediere și de prelucrare a traficului, cum ar fi QoS, filtrare, monitorizare. Traficul poate intra sau părăsi planul de date SDN prin porturile fizice sau logice, și poate fi dirijat în interior sau în exterior de funcțiilor de expediere sau de prelucrare. Prelucrare traficului poate fi efectuată de un motor OAM, de o funcție de criptare sau de o funcție virtualizată de rețea. Controlul funcțiilor de expediere sau de prelucrare a traficului poate fi efectuat de către un controller SDN sau prin mecanisme separate, eventual orchestrate în strânsă legătură cu controller-ul SDN.

Data plane implementează deciziile de expediere luate în controller palne. În principiu, nu poate lua decizii de expediere autonome. Cu toate acestea, controller plane poate configura data plane pentru a răspunde anonim la evenimente precum defecțiuni de rețea sau pentru susținerea funcțiilor livrate de exemplu de, LLDP, STP, BFD sau ICMP.

Interfața dintre data plane și controller plane (D-CPI) conține funcții cum ar fi:

- Controlul pragmatic al tuturor funcțiilor expuse de RBD
- Avertizarea capabilităților
- Notificarea evenimentelor

Agentul planului de date este entitatea care execută instrucțiunile controller-ului SDN în planul de date. Coordonatorul planului de date este entitatea prin care funcția de management alocă resursele planului de date la diverși agenți ale clienților și stabilește politica pentru utilizarea lor. Agenții și coordonatorii au același scop, în fiecare plan al arhitecturii.

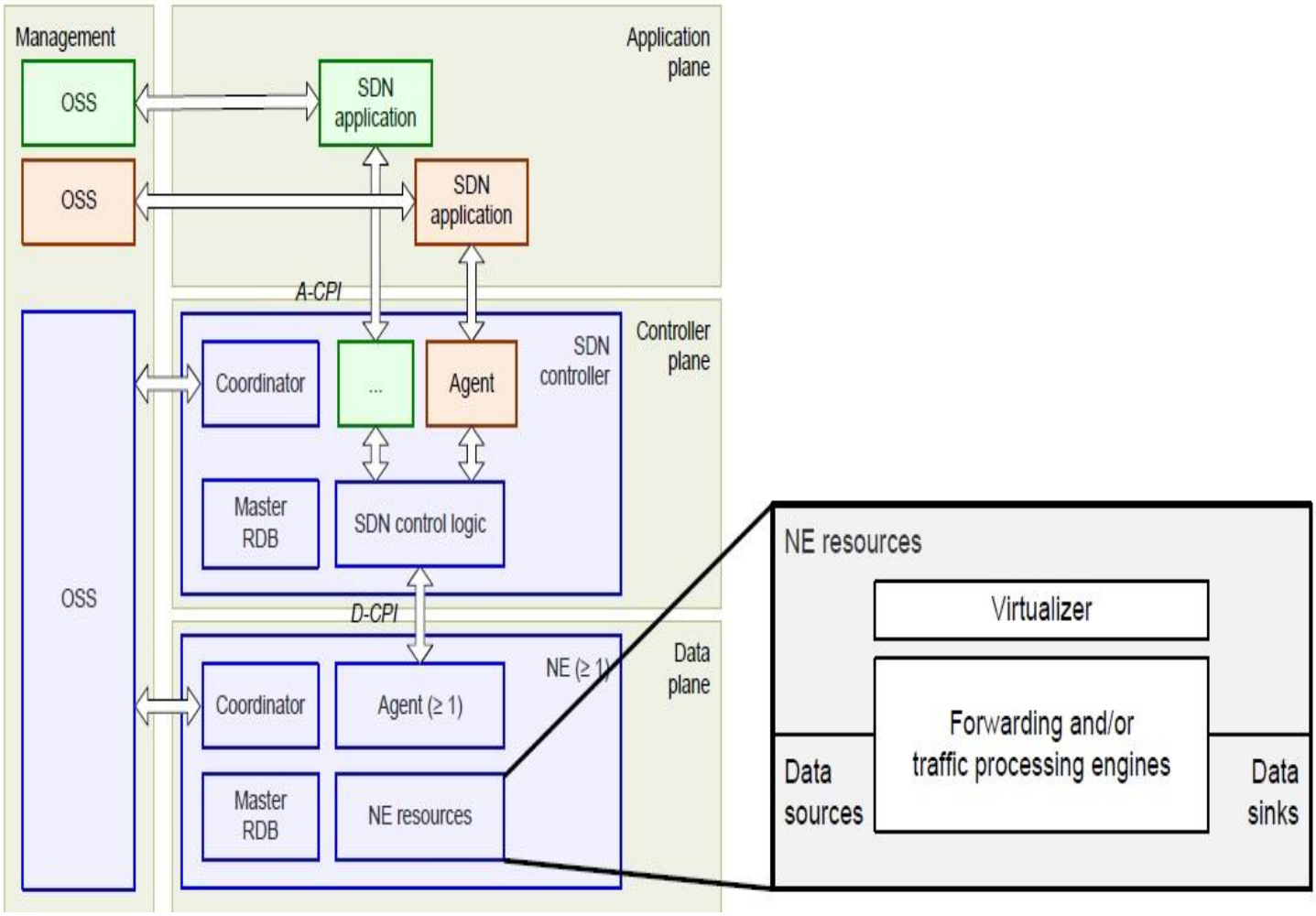

Figura 2.2. Resursele NE (Network elements)

2.3. Controller plane

2.3.1. Prezentare generală

Arhitectura nu specifică design-ul intern a unui controller SDN.

Controlul SDN este centralizat în mod logic.

- Un controller de obicei are domeniul de aplicabilitate o subrețea, care acoperă mai mult de un singur NE fizic.
- Nu există nici o dispută cu privire la resurse cu alte entități; controlerul SDN se consideră proprietarul resurselor virtuale care îi sunt alocate de către funcția de management.

Funcțiile și serviciile care fac parte din comportamentul vizibil din exterior a unui controller includ o vizibilitate completă a instanței modelului informațional aflat sub controlul său. Funcțiile suplimentare care pot fi necesare, depind de anumite circumstanțe:

- Învățarea topologiei și calculul de căi (path computation) (operatorul poate invoca, de asemenea, un serviciu extern pentru aceste funcții)
- Crearea și menținerea unui model suplimentar abstractizat de resurse pentru aplicațiile sale, cu resursele delimitate de politica aplicată. Virtualizarea și controlul resurselor sunt potențial recursive.

Un controller SDN este de așteptat să coordoneze o serie de resurse interdependente, de multe ori distribuite într-o serie de platforme subordonate și uneori să asigure integritatea tranzactionala, ca parte a procesului. Acest lucru se numește în mod obișnuit orchestrație. Un orchestrator este uneori considerat a fi un controler SDN în sine, dar domeniul de aplicare redus a unui controler de nivel inferior nu elimină necesitatea unui controller SDN de nivel inferior pentru a efectua orchestratie peste propriul domeniu de control.

2.3.2. Controller-ul SDN

Arhitectura SDN nu specifică design-ul interior sau implementarea unui controlor SDN. Ar putea fi un singur proces monolitic; ar putea fi o confederație de procese identice, dispuse pentru a partaja sarcina sau pentru a se proteja unul de celălalt de eșecuri; ar putea fi un set de componente funcționale distincte într-un aranjament de colaborare; s-ar putea abona la servicii externe pentru unele dintre funcțiile sale, de exemplu pentru calculul de căi. Orice combinație dintre aceste alternative este permisă: controlerul SDN este privit ca o cutie neagră, definit prin comportamentul său observabil extern. Componentele controller-ului sunt libere să ruleze pe platforme de calcul arbitrare, inclusiv cu resursele de calcul locale a unui NE fizic. Ele pot, de asemenea, rula pe resurse distribuite cum ar fi pe mașini virtuale (VMs) din centrele de date.

Multiple componente ale manager-ului sau controller-ului pot avea acces comun de scriere asupra resurselor rețelei, dar pentru a se conforma principiilor SDN, ele trebuie să fie:

a) configurate pentru a controla seturi disjuncte de resurse sau acțiuni;
b) sincronizate una cu cealaltă, pentru a nu emite comenzi inconsistente sau conflictuale.

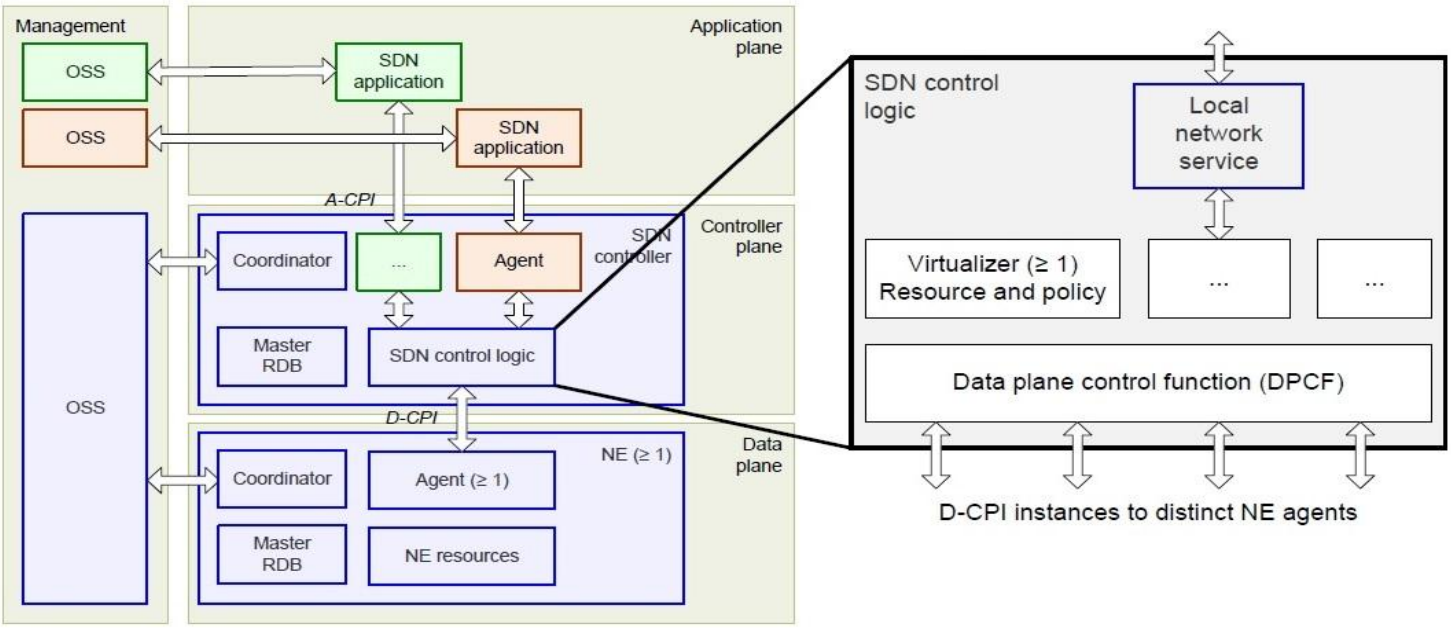

Figura 2.3. Logica de control a SDN

2.3.3. Elemente funcționale ale controller-ului SDN

- Funcția de control a planului de date (Data plane control function - DPCF)

Componenta DPCF deține resursele subordonate aflate la dispoziția sa și le folosește în conformitate cu instrucțiunile OSS/coordonatorului sau virtualizer-ul care le controlează. Aceste resurse iau forma unei instanțe a unui model informațional accesate prin intermediul agentului din nivelul de subordonare.

Deoarece domeniul de aplicare al unui controller SDN este de așteptat să se extindă peste mai multe NE (virtuale) sau chiar peste mai multe rețele virtuale (cu o instanță D-CPI distinctă pentru fiecare), DPCF trebuie să includă o funcție care operează peste aceste resurse agregate. Această funcție este denumit în mod obișnuit orchestrație. Această arhitectură nu specifica orchestrația ca fiind o componentă funcțională distinctă.

- Coordonatorul (Coordinator)

Pentru a configura ambele medii de client și de server, este nevoie de funcția de management. Coordonatorul este componenta functională a controlerului SDN care acționează în numele managerului. Clienții și serverele necesită o gestionare, din perspectiva planelor de date, de aplicație si de controller astfel încât blocurile funcționale coordonatoare sunt omniprezente.

- Virtualizer

Un controller SDN oferă servicii aplicațiilor prin instanțierea unui model informațional, care este derivat din resursele de bază, din politica de management instalată și din funcțiile de suport locale sau externe. Entitatea funcțională care suportă instanță modelului informațional și politica de la nivelului unei A-CPI (interfețe aplicație-controller) este numit Virtualizer. Acesta prezintă limita domeniului local de încredere agentului corespunzător, care reprezintă punctul de vedere al clientului cu privire la instanța modelului informațional.

Un virtualizer este instanțiat de OSS/coordonator pentru fiecare aplicație client sau organizație. OSS/ coordonatorul alocă resursele folosite de virtualizer pentru vizualizarea interfeței A-CPI pe care o expune clientului aplicație și instalează politica care trebuie aplicată de către virtualizer. Efectul acestor operațiuni este crearea unui agent pentru un client dat.

Virtualizer-ul poate fi gândit ca procesul care primește cereri specifice din parte clientului prin interfața A-CPI, validează cererile împotriva politicii și a resurselor alocate clientului, traduce cererea în ceea ce privește resursele de bază și trimite rezultatele către DPCF și către interfața D-IPC.

Virtualizer-ul și DPCF și eventual alte funcții ale controller-ului SDN trebuie să colaboreze pentru a asigura funcții, cum ar fi interpretarea notificărilor, partajarea resurselor, furnizarea serviciilor implicite și integritatea tranzacțională.

- Agentul

Orice protocol trebuie să se termine într-un anumit mod la entitatea funcțională. Un model controller-agent este adecvat pentru relația dintre entitatea controlată și entitatea de control și se aplică recursiv la arhitectura SDN. Entitatea controlată este desemnată de agent, o componentă funcțională care reflectă resursele și capabilitățile clientului în mediul serverului.

Un agent într-un anumit controller SDN, la nivelul N reprezintă resursele și acțiunile disponibile pentru un client sau pentru o aplicație a controlerului SDN, la nivelul N + 1. Un agent la nivelul N-1 în planul de date reprezintă resursele și acțiunile disponibile la nivelul N al controller-ului SDN. Chiar dacă locația fizică a agentului

este în interiorul domeniului de încredere a serverului agentul se află teoretic în domeniul de încrederea a clientului.

2.3.4. Delegarea controlului

O lectură mai nuanțată a principiului decuplării permite unui controller SDN de a delega funcții de control planului de date, sub rezerva unei cerințe cs aceste funcții să se comporte în moduri acceptabile pentru controller; ceea ce reprezintă că controller-ul nu ar trebui să fie surprins. Această interpretare este vitală pentru a obține o modalitate de a aplica principiile SDN în lumea reală.

Criteriile care încurajează controller-ul să delege o funcție planului de date includ:

- Reacție rapidă în timp real necesar pentru evenimentele de rețea;
- O cantitate mare de trafic ce trebuie procesată;
- Funcții orientate pe octet sau pe bit care nu se pretează cu ușurință la pachetizare;
- Cu valoare redusă, posibil repetive, predictibile, bine înțelese, cu comportament complet standardizat, de exemplu criptarea, BIP, inserția AIS, MAC learning, schimburi CCM;
- Supraviețuire sau continuitate în caz de defectare a controller-ului sau re-inițializare;
- Funcționalitate de obicei disponibilă în siliciul planului de date;
- Oportunitatea de a adăuga valoare prin separarea funcției.

2.3.5 Resursele partajate

Modelul planului de date presupune că resursele sunt dedicate clienților sau aplicațiilor. Intr-un model de resurse dedicate, există posibilitatea limitată de a spori eficiența utilizării resurselor subiacente, ceea ce reprezintă unul dintre beneficiile anticipate ale SDN. O modalitate de a rezolva această problemă este de a stabili prin contract partajarea best-effort a resurselor, posibil, cu contracte de prioritizare și de ponderare a traficului prioritare. O extindere a funcției de partajare a resurselor, este reprezentată posibilitatea ca furnizorul să poată menține un ansamblu de resurse neangajate pentru alocarea la cerere (și facturare) clienților pe principiul primul venit primul servit, sau print-un program de pre-negocire .

O altă formă de partajare a resurselor are loc atunci când o fracțiune a unei resurse se angajează la mai mulți clienți, de exemplu 20% din lățimea de bandă a unei legături. Acest lucru nu poate fi reprezentat în modelul simplu în care un client deține în totalitate instanțele unor obiecte. Unele astfel de resurse pot fi alocate static, mai bine decât la cerere și pote fi imune la suprasubscriere.

Controlerul SDN are responsabilitatea de a administra resursele care sunt partajate de mai mult de un client sau de o aplicație, fie ele statice sau dinamice, cu o viziune comună a angajamentelor asumate clienților în cauză. Schimbarea cererile și eliberarea resurselor partajate pot declanșa reoptimizarea întreagii rețele a furnizorului. Arhitectura nu alocă această responsabilitate către anumite componente funcționale ale controller-ului SDN.

2.3.6. Domenii administrative multiple

Figura 2.4. ilustrează un caz în care fiecare dintre administrații deține propria sa rețea, care sunt reciproc interconectate.

Figura 2.5. organizează aceste subrețele în domenii administrative și abstractizează vizualizarea pentru a afișa numai probleme de interes pentru Roșu. Pentru Roșu se pot observa detalii complete a propriilor NE și porturi, dar domeniile Albastru și Verde sunt abstractizate la cea mai simplă reprezentare posibilă.

Figura 2.6. ilustrează opțiunile pentru asocierile controller-ului SDN pentru Roșu.

Putea avea cazul arhitectural când un controler SDN oferă servicii prin intermediul interfeței A-CPI și în care clientul pentru astfel de servicii este, de fapt, un alt controller SDN. Așa cum se arată în figura 2.6., controlerul client (Alb) sau o aplicație de rețea, poate orchestra o serie de controllere de server.

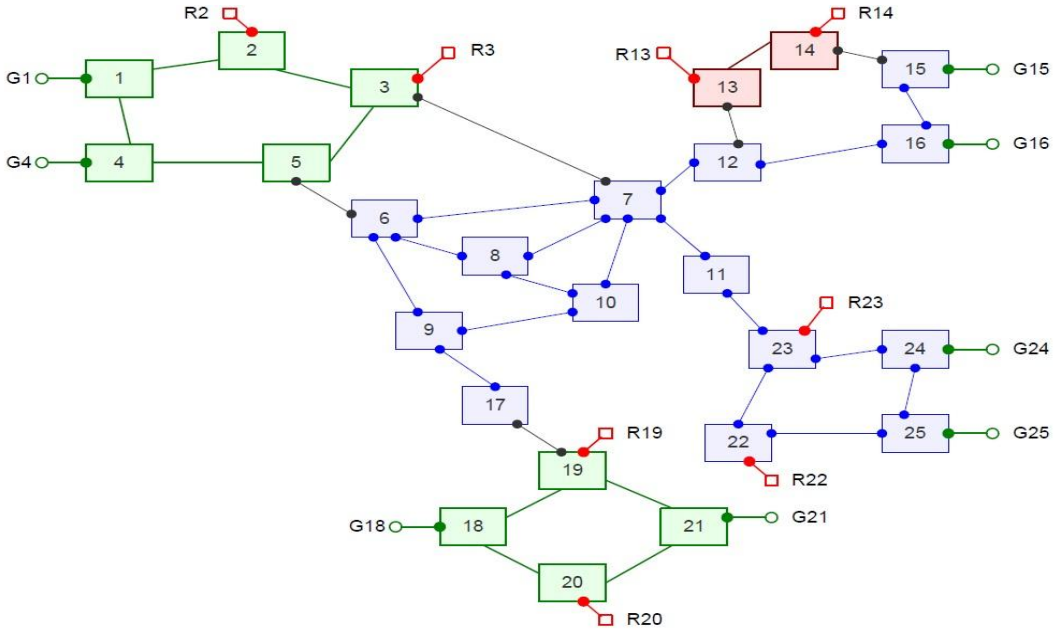

Figura 2.4. Exemplu de rețea cu mai mulți proprietari

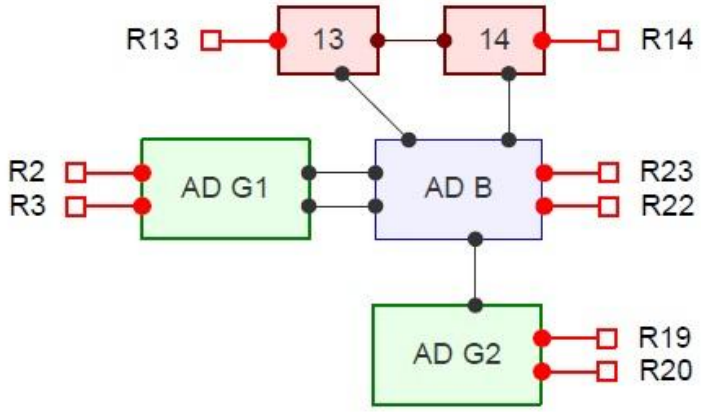

Figura 2.5. Domeniile administrative de interes pentru Roșu

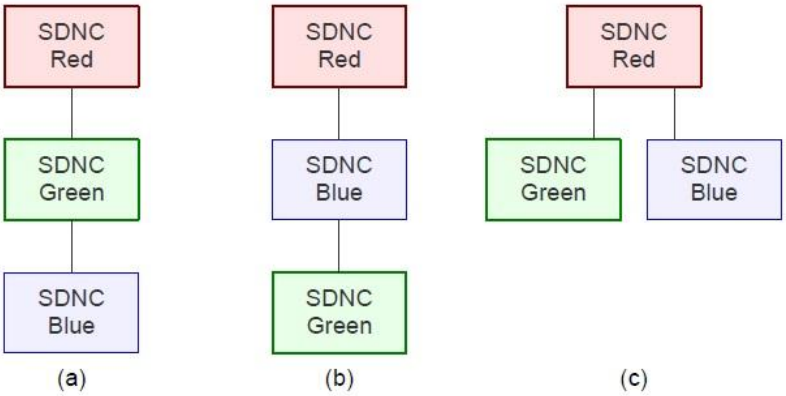

Figura 2.6. Opțiunile de organizare

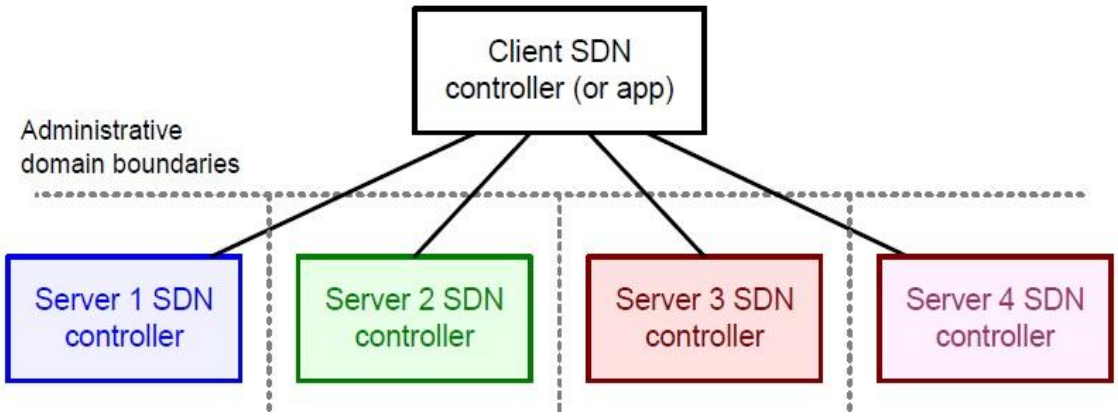

Figura 2.7. Exemplu comun de coordonare controller

2.3.7. Coordonarea controller-controller

Motivele pentru a separa controller-ele SDN în domenii egale distincte pot include orice combinații din următoarele considerente:

- Controller-ele pot fi de la diferiti furnizori care nu au obținut interoperabilitate completă;
- Controller-ele sau infrastructura de bază pot fi deținute sau operate de diferite organizaţiile administrative;
- Controller-ele pot avea diferite tehnologii;
- Scalabilitate a numărului de noduri din rețea sau întinderea geografică, inclusiv distincția dintre WAN și LAN.

Figura 2.8. ilustrează un simplu set de domenii de control a rețelei NCD1..NCD4, împreună cu controller-ele SDN asociate. Într-un hibrid a exemplelor anterioare, clientul Alb este prezentat ca având o relație directă cu clientul Albastru și clientul Verde, dar numai o relație indirectă cu Roșu. NCD3 este exemplificat ca un domeniu fără controller SDN; serviciile care se termină în NCD3 sau traversează NCD3 trebuie să utilizeze protocoale de semnalizare sau de rutare existente sau acorduri de pre-negociate.

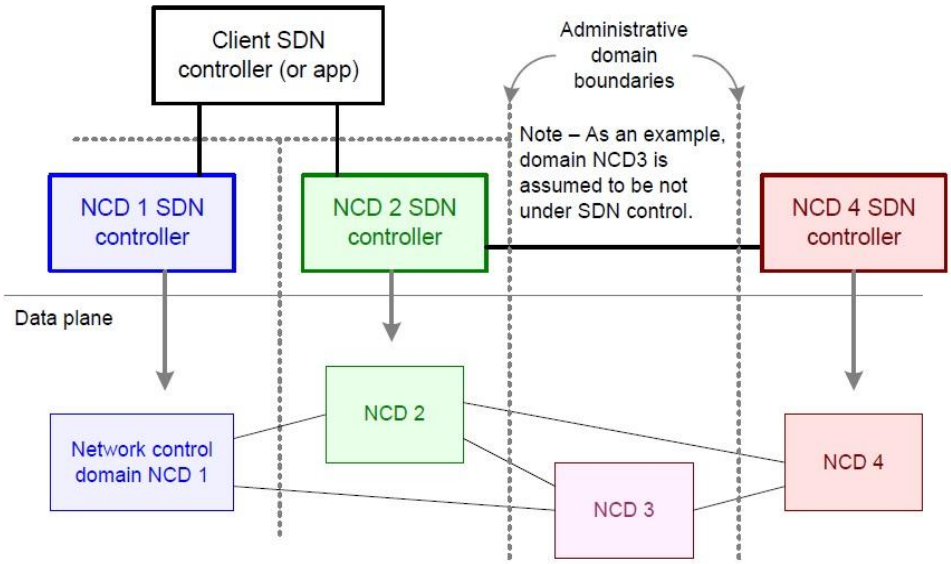

Figura 2.8. Exemplu de coordonare controller peer-peer

2.4. Application plane

Principiile SDN permit ca aplicațiile să precizeze resursele și comportamentul de care au nevoie de la rețea, în contextul unui acord de afaceri și de politică. Interfața dintre controller-ul SDN și planul aplicațiilor este numită interfața application-controller, A-CPI. Figura 2.9. ilustrază că o aplicație SDN poate suporta ea un agent A-CPI, care permite ierarhii recursive a aplicației. Diferitele niveluri de ierarhie a aplicației sunt descrise ca având diferite latitudini, în funcție de gradul lor de abstractizare.

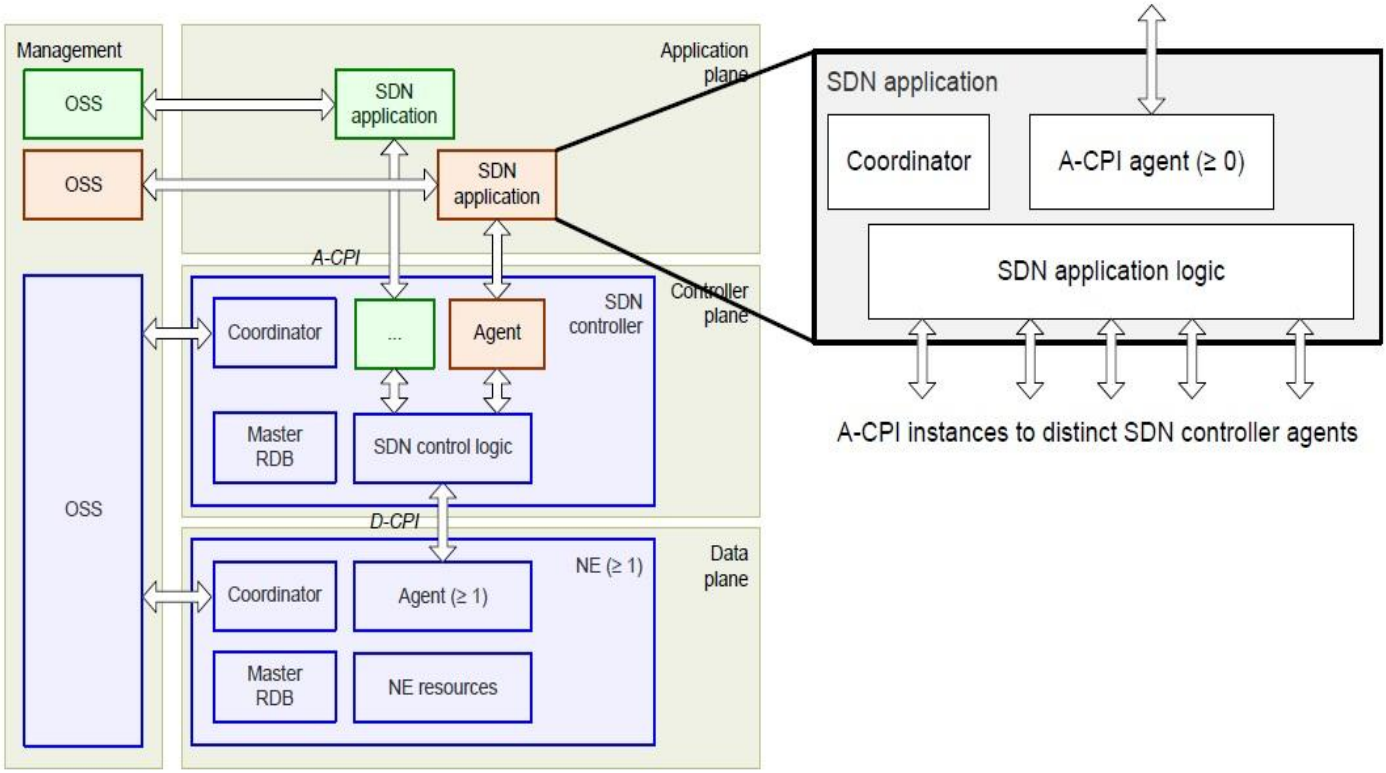

Figura 2.9. SDN Application plane

O aplicație SDN poate invoca alte servicii externe și poate orchestra orice număr de controllere SDN pentru atingerea obiectivelor sale. Legătura OSS și funcția de coordonator recunoasc faptul că, la fel ca celelalte blocuri majore ale arhitecturii, aplicațiile SDN necesită cel puțin o anumită cantitate de cunoaștere apriori a mediilor și rolurile lor.

Figura 2.10, ilustrează posibilitatea ca un sistem al utilizator să prezinte atât planul de date cât și planul aplicație. Un host sau un dispozitiv de rețea se pot potrivi acestui model. Un firewall sau detector DDOS ar exemplifica cazul dispozitivului de rețea, precum și un terminal client care a fost capabil de a semnala stările sale existente sau dorite este un exemplu de host.

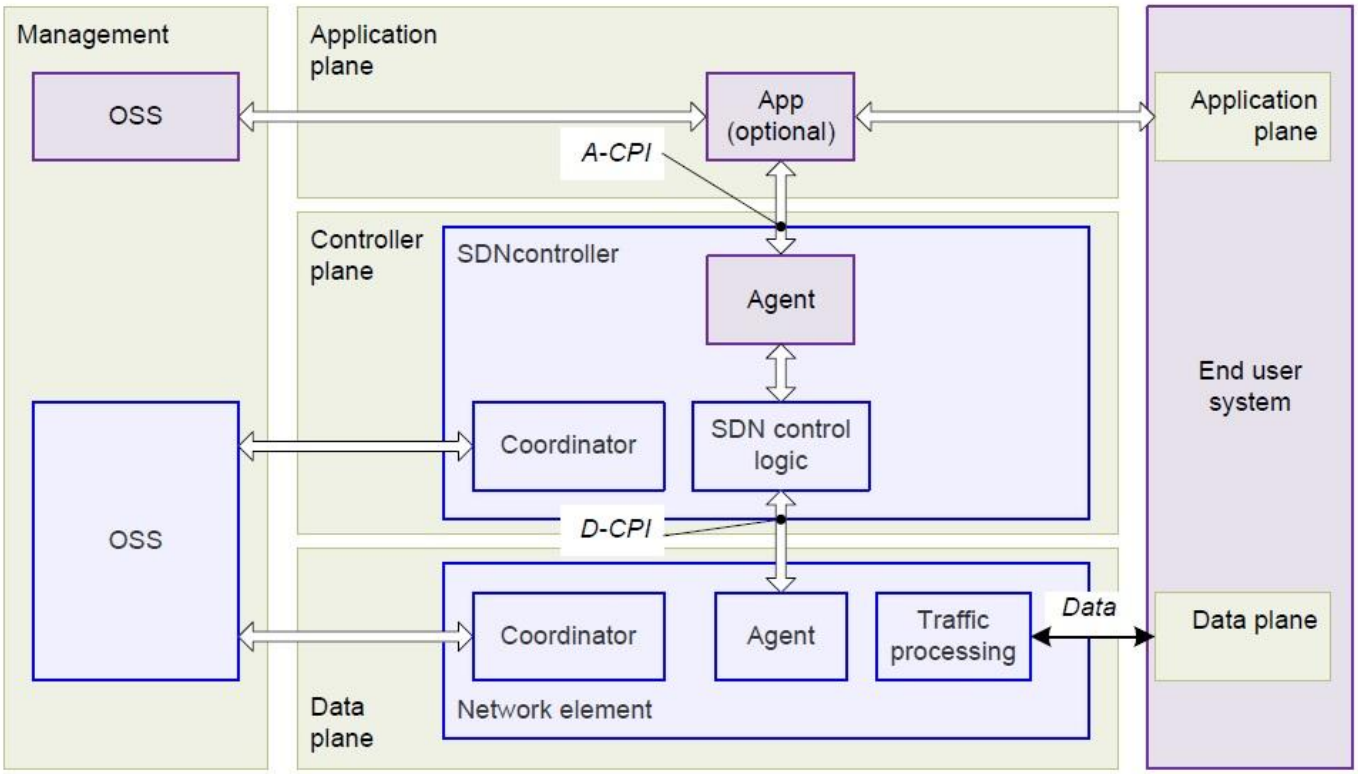

Figura 2.10. Exemple de terminal multi-plane

2.5. Virtualizare

Controlul și management SDN trebuie să fie proiectate pentru a funcționa pe resurse abstracte și virtualizate, care în cele din urmă cuprind resurse fizice de nivel inferior, eventual prin mai multe niveluri succesive de virtualizare. Acest lucru se face prin intermediul unui model de informații comun, care include o reprezentare a hardware-ului fizic ca un caz special.

2.6. Funcția de management

Funcția de management acoperă task-uri de suport a infrastructurii care nu trebuie să fie efectuate de către planele de aplicație, control si de date. Funcția de management poate efectua și operații pentru care planele aplicație, de control și de date sunt restricționate de politica sau din alte motive. Poate că motivul cel mai important pentru a preveni o sarcină de a fi executata de componentele SDN este faptul că controller-ul SDN poate fi amplasat într-un domeniu de încredere pentru clienți, în timp ce acordul business-ului spun că managementul de bază și funcțiile de sprijin să fie realizate din cadrul domeniului de încredere al furnizorului. Deşi o politică agent ar putea fi recomandată ca fiind de încredere completă pentru controller, politica de transparență și software-ul de aplicare a politicii ar trebui, totuși, să fie instalate de către managerul furnizorului. Din motive de securitate, comportamentul implicit este recomandat pentru a fi a expune nimic, mai degrabă decât orice.

Arhitectura SDN recunoaște funcțiile de management clasice, cum ar fi de inventarul echipamentelor, izolarea erorilor, upgrade de software și altele asemenea, dar le privește ca fiind inafara domeniul de aplicare al

SDN. Unul dintre beneficiile percepute ale SDN este de a permite clienților (în domenii de încredere străine) să realizeze multe dintre acțiunile care sunt în prezent efectuate de sisteme de management.

Două roluri de management sunt recunoscute: manager de server și manager de clienți. Responsabilitățile managerului de server nu sunt aceleași ca și cele ale managerului de clienți.

2.7. Modelul informațional

Se pot folosi o varietate de protocoale între diversele componente ale unei software-defined network, pentru a servi o varietate de scopuri. Deși, în mod evident este de dorit să se reducă numărul de protocoale, arhitectura nu menționează utilizarea unor protocoale speciale. Ceea ce este esențial este faptul că toate entitățile comunicante împart un model informațional comun. Acest lucru nu trebuie să fie confundat cu o reprezentare de date comune care este o problemă de protocol. Într-adevăr, atunci când contextul permite, elementele modelului informațional pot fi înțelese în mod implicit, decât de a fi transmise în mod explicit de un protocol.

Acestă arhitectură modelează operarea SDN ca manipularea unor instanțe de obiect gestionate (MO - managed objects), peste diferite interfețe. Operațiile efectuate peste instanțele MO includ CRUD: a crea, a citi, a actualiza, a șterge precum și invocarea metodelor definite peste clasele MO și abonarea la notificările lor. Ca atare, modelarea informațională este o funcție de bază a arhitecturii și a evoluției sale în timp.

Această arhitectură recomandă re-utilizarea modelelor informaționale existente. Nu se așteaptă în mod necesar că modelele din surse mai largi din industrie să fie importate direct și complet intr-un model informațional SDN. Adaptarea la contextul SDN poate lua în considerare atât caracteristicile speciale ale SDN și cele mai bune practici aflate în continuă dezvoltare. Cu toate acestea, modelul SDN rezultat trebuie aleas și documentat astfel încât să poată fi ușor înțeles și folosit în afara comunității SDN. Aceast lucru comprimă curba de învățare și încurajează migrarea, prin facilitarea integrării în infrastructura existentă.

3. Concluzii

Principalele avantaje ale Software-Defined Networking:

- Reducerea costurilor - În primul rând, SDN nu are nevoie de o investiție uriașă. Există chiar câteva produse SDN care sunt free. În timp ce costurile unor soluții SDN sunt destul de mari, cum ar fi NSX VMware, există unele soluții care sunt implementate în sistemul de operare în sine - Microsoft Hyper-V Network Virtualization.

- Reducerea overhead-ului - Într-un mediu fizic, izolarea traficului diferiților clienți necesită configurarea de VLAN-ne pe dispozitivele din rețea. Deoarece majoritatea deciziilor sunt efectuate în SDN, este ușor pentru furnizorii de servicii să izoleze mașinile virtuale apaținând clienților printr-o diversitate metode de izolare disponibile în arhitectura SDN.

- Fizic vs. Virtual Networking Management - Mediile fizice necesită colaborarea între diferite echipe pentru a efectua un task. De exemplu, dacă sunt necesare unele modificări la un dispozitiv din rețeaua fizică, va fi nevoie de o cantitate considerabilă de timp și colaborarea mai multor echipe pentru a realiza acestă sarcină. SDN oferă posibilitatea de a controla rețelele virtuale și cele fizice prin utilizarea unui instrument central de management. Un administrator virtual poate procesa modificările necesare fără a fi nevoie să colaboreze cu diferite echipe.

- Managing Virtual Packet Forwarding - SDN poate ajuta la transmiterea pachetelor virtuale unui software sau dispozitiv fizic din rețea. De exemplu, în cazul în care o mașină virtuală are nevoie de acces la internet, devine ușor pentru administratorii virtuali să realizeze configurația necesară pentru mașina virtuală cu un minim de efort.

- Downtime redus - SDN ajută la virtualizarea multora dintre dispozitivele de rețea, devenind ușor de a efectua un upgrade pentru o singură masină în comparație cu cazul în care upgrade-ul trebuie efectuat pe mai multe dispozitive. SDN sprijină, de asemenea snapshotting de configurație, ceea ce este avantajos pentru recuperarea rapidă în cazul avariilor cauzate de upgrade.

- Izolarea și controlul traficului - Furnizorii de servicii cloud pot beneficia de centralizarea controlului din rețea cu ajutorul unui instrument central de management. Totodată, SDN oferă mai multe mecanisme de izolare, cum ar fi configurarea ACL și firewall la nivelul NIC-ului mașinii virtuale. Deasmenea se pot defini reguli de trafic, folosind consola de administrare SDN, ceea ce ajută la asigurarea unui control complet asupra traficului din rețea.

- Extensibilitate: Deoarece SDN este bazat pe software, se pot utiliza ușor API SDN pentru ca furnizorii să-si extindă capacitatea unei soluții SDN prin dezvoltarea de aplicatii pentru a controla comportamentul traficului din rețea.

- Central Networking Management Tool: SDN poate furniza toate necesitățile rețelei într-un singur produs, permițând controlul fiecărei părți din rețeaua unei organizații, folosind un instrument de management centralizat.

4. Bibliografie

1. RFC - 7426: Software-Defined Networking (SDN): Layers and Architecture Terminology - https://tools.ietf.org/html/rfc7426

2. RFC - 7149: Software-Defined Networking: A Perspective from within a Service Provider Environment - https://tools.ietf.org/html/rfc7149

3. ONF, SDN architecture overview- https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/SDN-architecture-overview-1.0.pdf

4. ONF, Software-Defined Networking: The New Norm for Networks (2012)

5. ONF, OpenFlow switch specification (2013)

6. Software Defined Networking - Cengiz Alaettinoglu, Ph.D.