

УНИВЕРЗИТЕТ У БЕОГРАДУ
ФАКУЛТЕТ ОРГАНИЗАЦИОНИХ НАУКА

Јелена Т. Шух

**ИНФРАСТРУКТУРА ЗА
Е-ОБРАЗОВАЊЕ ЗАСНОВАНА НА
СОФТВЕРСКИ ДЕФИНИСаниМ
МРЕЖАМА**

докторска дисертација

Београд, 2017.

UNIVERSITY OF BELGRADE
FACULTY OF ORGANIZATIONAL SCIENCES

Jelena T. Šuh

**INFRASTRUCTURE FOR E-EDUCATION
BASED ON SOFTWARE DEFINED
NETWORKING**

Doctoral Dissertation

Belgrade, 2017.

Ментор:

др Божидар Раденковић,

Редовни професор, Универзитет у Београду,

Факултет организационих наука

Чланови комисије:

др Маријана Деспотовић - Зракић

Редовни професор, Универзитет у Београду,

Факултет организационих наука

др Милорад Станојевић,

Редовни професор у пензији, Универзитет у Београду,

Саобраћајни факултет

Датум одбране: _____

Инфраструктура за е-образовање заснована на софтверски дефинисаним мрежама

Апстракт: Предмет истраживања докторске дисертације је развој модела инфраструктуре за е-образовање, који је заснован на концепту и технологијама софтверски дефинисаних мрежа, као и развој образовног модула за учење софтверски дефинисаних мрежа. Централни проблем је испитивање могућности примене концепта софтверски дефинисаних мрежа у оквиру *cloud computing* инфраструктуре образовних установа и креирања адекватног образовног модула за учење софтверски дефинисаних мрежа.

Главна хипотеза која је испитивана у оквиру дисертације јесте да се применом развијеног модела инфраструктуре и образовног модула остварује позитиван утицај на ефикасност образовног процеса. У раду је представљена структура предложеног модела који обухвата елементе инфраструктуре и архитектуре образовног система и могућности интеграције инфраструктуре са образовним сервисима.

У раду је описан процес пројектовања *cloud* инфраструктуре засноване на концепту софтверски дефинисаних мрежа, као и моделирања кључних индикатора перформанси образовне инфраструктуре. Имајући на уму важност едукације у области информационо-комуникационих технологија, у дисертацији је представљен и процес пројектовања образовног модула за учење софтверски дефинисаних мрежа.

У експерименталном делу докторске дисертације предложени образовни модул за учење софтверски дефинисаних мрежа је развијен и имплементиран у образовном процесу. Евалуација модела је извршена у Лабораторији за електронско пословање Факултета организационих наука Универзитета у

Београду. Реализовано је тестирање и мерење релевантних параметара који утичу на ефикасност предложеног модела. Резултати истраживања показују да предложени модел унапређује процес учења у области софтверски дефинисаних мрежа. Модел се може на једноставан начин прилагодити и применити у процесу едукације у различитим областима, а посебно је погодан за едукацију у области нових информационо-комуникационих технологија.

Кључне речи: инфраструктура за е-образовање, cloud computing, софтверски дефинисане мреже.

Научна област: Информациони системи и технологије

Ужа научна област: Електронско пословање

УДК број: 004:37

Infrastructure for e-education based on software defined networking

Abstract: *The subject of this dissertation is the development of an infrastructure model for e-education, based on the concept and technologies of software defined networks as well as development of an educational module for learning software defined networks. The main problem discussed in the thesis is to investigate the possibilities of applying the concept of software defined networks within cloud computing infrastructure of educational institutions and the creation of adequate educational module for software defined networks.*

The main hypothesis examined in the dissertation is that the application of the developed infrastructure model and the educational module has a positive impact on the efficiency of the educational process. The thesis presents the structure of the proposed model, which includes elements of educational system's infrastructure and architecture and integration possibilities with the educational services.

The thesis describes the process of design of cloud infrastructure based on the concept of software defined networks, and modeling of key performance indicators of educational infrastructure. Bearing in mind the importance of education in the field of information and communication technologies, the thesis also presents the process of design of educational module for learning software defined networks.

In the experimental part of the doctoral thesis, the proposed education module for learning software defined networks is developed and implemented within the educational process. The model evaluation was performed in the Laboratory for e-business of the Faculty of Organizational Sciences, University of Belgrade. The testing and measuring of relevant parameters that affect the efficiency of the proposed model is realized. The research results showed that the proposed model improves the learning process in the field of software defined networks. The model can be easily adapted and

applied in the process of education in various fields, and is particularly suitable for education in domain of new information and communication technologies.

Key Words: *infrastructure for e-education, cloud computing, software defined networking.*

Scientific field: Information Systems and Technology

Scientific subfield: E-business

UDK number: 004:37

Садржај

1.	Увод.....	1
1.1.	Дефинисање предмета истраживања.....	2
1.2.	Циљеви истраживања.....	5
1.3.	Полазне хипотезе.....	7
1.4.	Методе истраживања.....	9
2.	Анализа постојећих решења.....	11
2.1.	Основни концепти е-образовања.....	11
2.2.	Појам и дефиниција ИТ инфраструктуре.....	12
2.2.1.	Компоненте ИТ инфраструктуре.....	13
2.3.	<i>Cloud computing</i> инфраструктура.....	14
2.4.	Разлози за увођење софтверски дефинисаних мрежа.....	16
3.	Софтверски дефинисане мреже.....	19
3.1.	Појам и дефиниција софтверски дефинисаних мрежа.....	19
3.2.	Архитектура софтверски дефинисаних мрежа.....	21
3.3.	SDN контролери.....	23
3.4.	Протоколи у софтверски дефинисаним мрежама.....	25
3.4.1.	<i>OpenFlow</i> протокол.....	25
3.4.2.	OVSDB протокол.....	29
3.5.	Модел реализације софтверски дефинисаних мрежа.....	33
3.5.1.	Реализација софтверски дефинисаних мрежа у <i>Linux</i> окружењу....	33
3.5.2.	<i>OpenDaylight</i>	35
3.5.3.	Комерцијална решења.....	36
3.6.	<i>Application centric</i> инфраструктура.....	38
3.7.	Безбедност у софтверски дефинисаним мрежама.....	41
3.8.	Квалитет сервиса у софтверски дефинисаним мрежама.....	45
3.9.	Миграција на SDN архитектуру.....	47
3.10.	Емулација софтверски дефинисаних мрежа.....	49
3.11.	Примена софтверски дефинисаних мрежа.....	52
3.11.1.	Примена софтверски дефинисаних мрежа у предузећима.....	52

3.11.2.	Примена софтверски дефинисаних мрежа у провајдерским окружењима.....	53
3.11.3.	Примена софтверски дефинисаних мрежа у академским кампус окружењима.....	55
4.	Модел SDN инфраструктуре за е-образовање	58
4.1.	Развој модела инфраструктуре	58
4.2.	Анализа постојећих модела инфраструктуре.....	59
4.3.	Структура предложеног модела инфраструктуре.....	61
4.4.	Приказ главних компоненти структуре предложеног модела	64
4.4.1.	Логички модел архитектуре.....	64
4.4.2.	Модел <i>cloud</i> инфраструктуре.....	65
4.4.3.	Модел SDN инфраструктуре	67
4.4.4.	Приказ сервиса е-образовања	73
4.4.5.	Приказ интеграције компоненти инфраструктуре.....	75
4.4.6.	Модел управљања инфраструктуром.....	76
4.5.	Евалуација перформанси	78
5.	Образовни модул за учење софтверски дефинисаних мрежа	82
5.1.	Пројектни захтеви.....	82
5.2.	Процес креирања образовног модула за учење софтверски дефинисаних мрежа	83
5.3.	Пројектовање и имплементација образовног модула за учење софтверски дефинисаних мрежа.....	85
5.3.1.	Фаза припреме SDN образовног модула	85
5.3.2.	Фаза планирања SDN образовног модула	85
5.3.3.	Фаза развоја инфраструктуре за SDN образовни модул	87
5.3.4.	Фаза имплементације SDN образовног модула	89
5.3.5.	Фаза евалуације SDN образовног модула	92
5.4.	Анализа исхода образовног процеса.....	93
5.4.1.	Образовни модул 1	93
5.4.2.	Образовни модул 2	97
6.	Научни и стручни доприноси	105
7.	Будућа истраживања.....	110
8.	Закључак.....	112

9.	Литература.....	114
10.	Списак слика	127
11.	Списак табела.....	130
	Биографија аутора.....	131
	Изјава о ауторству	132
	Изјава о истоветности штампане и електронске верзије докторског рада	133
	Изјава о коришћењу	134

1. УВОД

Информациони системи високошколских институција морају да одговоре на строге захтеве у погледу доступности, скалабилности и безбедности. Поред тога, имају задатак да подрже велики број различитих образовних сервиса. Такође, велики број образовних институција реализује један део образовног процеса применом е-образовања, па је питање примене адекватне инфраструктуре од изузетног значаја.

Један од основних захтева е-образовања јесте прилагођавање образовних процеса и садржаја у складу са захтевима студената и наставног особља [1]. Комплексност ових захтева је утицала на развој и неопходност примене нових технологија. *Cloud computing* технологија представља ефикасно решење за реализацију инфраструктуре са виртуелизованим ресурсима [2] и из тог разлога налази примену у многим образовним институцијама.

Развој нових образовних сервиса и апликација поставља високе критеријуме које образовни системи морају да испуне. Да би њихова ефикасност била на високом нивоу, неопходна је примена динамичког окружења и нових технологија [3], које могу да подрже будући развој академског окружења, уз додатни захтев у погледу једноставне примене решења. Наведене захтеве може да испуни технологија софтверски дефинисаних мрежа (енгл. *Software Defined Networking – SDN*). Концепт софтверски дефинисаних мрежа уводи виши степен програмабилности мрежне инфраструктуре и омогућава динамичку доделу потребних ресурса [4].

Интеграцијом концепта софтверски дефинисаних мрежа и *cloud computing* технологија могуће је креирати образовно окружење високе доступности и скалабилности које омогућава прилагођавање образовног процеса учесницима у настави. Да би образовни процес био ефикасан, неопходно је паралелно са развојем инфраструктуре радити на развоју образовних модула за учење нових информационо-комуникационих технологија.

1.1. Дефинисање предмета истраживања

Предмет истраживања докторске дисертације је пројектовање и реализација модела инфраструктуре за е-образовање, заснованог на концепту софтверски дефинисаних мрежа као основи за ефикасну и ефективну реализацију пословних и образовних процеса у академском окружењу. Дисертација се бави проблемом едукације у области нових мрежних технологија, па предмет истраживања докторске дисертације обухвата методолошки приступ за креирање образовних модула из области софтверски дефинисаних мрежа и *cloud computing*-а.

Академске мреже треба да обезбеде услове за квалитетну реализацију образовног процеса и да повежу факултете, истраживачке центре, наставно особље и студенте који су ангажовани на различитим пројектима. Промене у образовном процесу и њихова динамика, као и појава нових информационо-комуникационих технологија представљају изазов и постављају нове захтеве пред систем за е-образовање [5]. Систем за е-образовање мора да пружи могућност брзог и једноставног увођења нових образовних садржаја и студијских програма и квалитетну реализацију образовног процеса прилагођеног потребама и захтевима студената и наставног особља. Предуслов за то је постојање адаптивне ИТ инфраструктуре, која треба да обезбеди подршку за различите апликације и сервисе у академским мрежама. Да би се реализовала оваква инфраструктура, потребно је у академске мреже унети висок степен програмабилности како би се повећала ефикасност реализације процеса у академском окружењу.

Постојеће академске мреже су засноване на традиционалном, хијерархијском моделу и могу да подрже сервисе и апликације када је реч о стандардној клијент-сервер комуникацији [6]. Оне захтевају познавање великог броја мрежних технологија и конфигурацију већег броја уређаја да би се имплементирао један сервис. Карактерише их постојање бројних нестандардних решења, што систем за управљање мрежом чини комплексним и са израженим проблемом интероперабилности опреме различитих произвођача. Велики број функционалности дефинисаних у хардверу за последицу има високу цену

реализације мрежне архитектуре, дуготрајан процес стандардизације протокола и недовољан степен флексибилности, што успорава процес увођења нових апликација и сервиса. Пројектовање и имплементација инфраструктуре за е-образовање у академском окружењу постали су комплексни узимајући у обзир пораст количине података у модерним мрежама и постојање великог броја мрежних технологија. Наведени недостаци традиционалних рачунарских мрежа представљају разлог за увођење динамичке архитектуре у пословању образовних институција [1].

Динамичка архитектура подразумева примену техника виртуелизације мрежних ресурса и софтверски дефинисаних мрежа за централизовано управљање виртуелном и физичком инфраструктуром [7]. Аутоматизацијом конфигурације у целој мрежи процес имплементације нових сервиса треба да буде знатно убрзан и поједностављен, а оперативни трошкови смањени. Да би се обезбедио континуалан рад у академској мрежи, неопходно је да постоји ефикасан механизам за брзу детекцију инцидентних ситуација у мрежи и преусмеравање саобраћаја, а без утицаја на сервисе корисника. Да би постојећи ресурси били искоришћени на ефикасан начин, потребно је да постоји могућност реализације оптималног рутирања и балансирања саобраћаја на апликативном нивоу. Логичка централизација и примена софтверски дефинисаних мрежа треба да омогући ефективно праћење перформанси система, уочавање нерегуларности и инцидентних ситуација, грануларну примену безбедносних полиса на нивоу сервиса и апликација уз флексибилно прилагођавање пословним и образовним потребама корисника академских мрежа.

Cloud computing се показао као ефикасно решење за реализацију модела скалабилне и апстраховане инфраструктуре са виртуелизованим ресурсима [8], па универзитети широм света развијају приватну *cloud* инфраструктуру, коју користе за изградњу система за е-образовање. Међутим, комплексни захтеви, као што је виши степен сарадње, размена великих количина података и развој нових технологија, намећу потребу да се у академске мреже унесе виши степен програмабилности применом концепта софтверски дефинисаних мрежа.

Софтверски дефинисане мреже омогућавају управљање мрежама и сервисима апстраховањем нижих слојева мрежне инфраструктуре [9]. Концепт је заснован на раздвајању процеса управљања мрежом од прослеђивања података, чиме се смањује комплексност архитектуре рачунарских мрежа. Захваљујући принципима централизације, апстракције и програмабилности, софтверски дефинисане мреже се примењују у рачунарским мрежама предузећа, у *data* центрима, у мрежама сервис провајдера, као и у академском окружењу.

Карактеристика софтверски дефинисаних мрежа је динамичка додела мрежних ресурса у складу са захтевима корисника, а без потребе да корисници познају мреже технологије и физичку топологију мреже [10]. Креирање виртуелног мрежног окружења независно од физичке топологије омогућава лаку имплементацију и тестирање нових сервиса за подршку образовним и научно-истраживачким процесима. На овај начин се креира флексибилно мрежно окружење које се једноставно прилагођава различитим инфраструктурним захтевима за сваку од области која се изучава на студијским програмима. Једноставно увођење нових предмета који су засновани на новим технологијама, као што су мобилно пословање, интернет интелигентних уређаја, *enterprise networking* и *big data*, олакшава образовни процес и студентима омогућава да на брз и ефикасан начин усвоје нова практична знања и искажу креативност у развоју нових апликација и сервиса.

У процесу креирања образовних апликација и сервиса је значајна примена интерфејса отвореног стандарда и софтвера отвореног кода [11] пошто доприноси ефикасности и једноставности имплементације, а утиче и на смањење трошкова реализације мрежне инфраструктуре. Академско окружење карактеришу онлајн сарадња и размена велике количине података. Појава велике количине података, односно *big data*, захтева виши степен флексибилности мрежне инфраструктуре у односу на традиционални приступ и могућност примене напредних апликација за обраду ових података. Зато је потребно обезбедити ефикасну инфраструктуру, која се динамички прилагођава захтевима корисника. ИТ инфраструктура, која је заснована на концептима *cloud computing*-а и софтверски дефинисаних мрежа,

може у великој мери одговорити на постављене комплексне захтеве уз поједностављење процеса имплементације и интеграције у оквиру система е-образовања.

Систем за е-образовање чине инфраструктура, односно одређене хардверске и софтверске компоненте, али и кадрови. Недостатак кадрова у овој области представља мотивацију за развој образовног модула за учење софтверски дефинисаних мрежа. Образовни модул мора на једноставан, али ефикасан начин да пружи студентима неопходна знања, која ће моћи да примене у реалним мрежних окружењима. Зато образовни модул мора да има интердисциплинарни карактер и да поред основних знања о традиционалним рачунарским мрежама, студентима омогући усвајање знања и из области програмирања, виртуелизације, *cloud computing*-а и софтверски дефинисаних мрежа.

У дисертацији је креиран образовни модул за учење софтверски дефинисаних мрежа на пројектованом моделу инфраструктуре за е-образовање који је заснован на софтверски дефинисаним мрежама и *cloud computing*-у. Наведено решење је евалуирано у оквиру рачунарске инфраструктуре Катедре за електронско пословање Факултета организационих наука Универзитета у Београду.

1.2. Циљеви истраживања

Примарни циљ истраживања је развој модела ИТ инфраструктуре за е-образовање високошколске установе, који је заснован на софтверски дефинисаним мрежама и развој образовног модула за учење софтверски дефинисаних мрежа. Циљ рада се реализује кроз имплементацију образовног модула у инфраструктури за е-образовање базираној на *cloud computing* технологији и софтверски дефинисаним мрежама. Циљеви који се постижу у оквиру истраживања у овом раду су:

- Пројектовање ИТ инфраструктуре засноване на софтверски дефинисаним мрежама неопходне за ефикасну реализацију пословних и образовних процеса у академском окружењу.
- Повећање ефикасности, поузданости, скалабилности и безбедности инфраструктуре за е-образовање коришћењем софтверски дефинисаних мрежа.
- Повећање степена доступности система е-образовања изградњом поуздане и скалабилне информационо-комуникационе инфраструктуре.
- Смањење комплексности реализације инфраструктурних и образовних сервиса применом ИТ инфраструктуре засноване на софтверски дефинисаним мрежама.
- Пројектовање ИТ инфраструктуре за е-образовање која захтева мање капиталне и оперативне трошкове.
- Развој модела евалуације инфраструктуре за е-образовање засноване на софтверски дефинисаним мрежама.
- Развој модела инфраструктуре за е-образовање који се једноставно може мигрирати у друга академска окружења.
- Обезбеђивање инфраструктуре за е-образовање која доводи до повећања мотивисаности студената за учење и постизања бољих резултата.
- Развој модела и пројектовање образовног модула за учење софтверски дефинисаних мрежа.
- Интензивно укључивање студента у процесе везане за истраживања и примену нових техничких решења заснованих на концептима напредних информационо-комуникационих технологија коришћењем модерне *cloud* инфраструктуре за е-образовање.

За остварење постављених циљева потребно је извршити следећа истраживања:

- Анализа постојећих модела мрежне инфраструктуре и утврђивање могућности примене софтверски дефинисаних мрежа у академским мрежама.
- Моделирање инфраструктуре засноване на софтверски дефинисаним мрежама у оквиру образовне институције.

- Моделирање кључних индикатора перформанси инфраструктуре засноване на софтверски дефинисаним мрежама.
- Моделирање образовног модула за учење софтверски дефинисаних мрежа у инфраструктури за е-образовање.
- Примена развијеног модела у образовном процесу и мерење резултата примене.

Научни циљ рада се огледа у дефинисању модела и метода за имплементацију концепта софтверски дефинисаних мрежа у *cloud* инфраструктуру и за реализацију образовног модула за учење софтверски дефинисаних мрежа. Коначни резултати треба да дају допринос формализацији и стандардизацији процеса пројектовања и имплементације ИТ инфраструктуре за е-образовање и процеса развоја образовног модула за учење софтверски дефинисаних мрежа.

1.3. Полазне хипотезе

Главна хипотеза која ће бити тестирана у раду гласи:

Развојем и имплементацијом модела инфраструктуре за е-образовање, који је заснован на примени софтверски дефинисаних мрежа, квалитативно се повећава ефикасност, поузданост, скалабилност и безбедност инфраструктуре система е-образовања. Флексибилно мрежно окружење представља основу за ефикасно увођење нових образовних садржаја и студентима омогућава да на брз и ефикасан начин усвоје нова знања чиме се побољшавају коначни резултати образовног процеса.

На основу дефинисаног предмета истраживања може се издвојити неколико посебних хипотеза:

X0.1. Могуће је развити систем ефикасне, поуздане, скалабилне и безбедне инфраструктуре за е-образовање, који је заснован на примени софтверски дефинисаних мрежа у *cloud* инфраструктури.

X0.2. Систем инфраструктуре за е-образовање реализован на примени концепта софтверски дефинисаних мрежа у *cloud* инфраструктури доприноси побољшању квалитета образовног процеса.

X0.3. Могуће је развити образовни модул за учење софтверски дефинисаних мрежа.

Даљим прецизирањем наведених посебних хипотеза, формулишу се поједини елементи који су предмет истраживања:

X0.1.1. Применом концепта софтверски дефинисаних мрежа и *cloud computing*-а могуће је реализовати инфраструктуру за е-образовање која је прилагођена потребама студената и наставног особља.

X0.1.2. Применом концепта софтверски дефинисаних мрежа у *cloud* окружењу могуће је обезбедити већи квалитет образовних сервиса него у случају традиционалне инфраструктуре.

X0.1.3. Перформансе инфраструктуре засноване на софтверски дефинисаним мрежама задовољавају захтеве у погледу поузданости, скалабилности и безбедности.

X0.2.1. Могуће је реализовати евалуацију модела *cloud* инфраструктуре заснованог на концепту софтверски дефинисаних мрежа.

X0.2.2. Студенти остварују добре резултате приликом евалуације савладаног градива када користе *cloud* инфраструктуру за е-образовање засновану на концепту софтверски дефинисаних мрежа.

X0.3.1. Могуће је утврдити квалитет имплементације образовног модула за учење софтверски дефинисаних мрежа на основу постигнутих резултата студената.

X0.3.2. Образовни модул за учење софтверски дефинисаних мрежа позитивно утиче на мотивацију и интересовање студената за учење нових информационо-комуникационих технологија.

1.4. Методе истраживања

Приликом израде ове дисертације, од општих научних метода користе се методе прикупљања и анализе научних резултата, моделирање, аналитичко-дедуктивна и статистичка метода. Моделирање се користи приликом израде модела *cloud* инфраструктуре за е-образовање заснованог на софтверски дефинисаним мрежама и образовног модула за учење софтверски дефинисаних мрежа. Аналитичко-дедуктивне методе користе се за вршење анализе података о постојећим решењима, о технологијама за развој и имплементацију предложеног модела, као и о учесницима у процесу е-образовања током експеримента. Мерење релевантних параметара и анализа добијених резултата се врши помоћу стандардних статистичких метода.

У експерименталном делу је извршена евалуација развијеног образовног модула за учење софтверски дефинисаних мрежа у оквиру инфраструктуре за е-образовање у Лабораторији за електронско пословање на Факултету организационих наука у Београду. У *cloud* инфраструктуру, реализовану применом *OpenStack*-а, интегрисан је концепт софтверски дефинисаних мрежа коришћењем *OpenDaylight* решења. Евалуација је извршена са образовног аспекта. Добијени резултати експеримента треба да потврде главну хипотезу о побољшању квалитета и ефикасности система за е-образовање.

Резултати истраживања су презентовани текстуално, описивањем и приказани кроз више табела, слика и дијаграма са упоредном анализом. Истраживање је

интердисциплинарно, јер укључује научне дисциплине: методологију, рачунарство, информатику, статистику, педагогију и друге научне дисциплине.

2. АНАЛИЗА ПОСТОЈЕЋИХ РЕШЕЊА

2.1. Основни концепти е-образовања

Развој информационо-комуникационих технологија имао је велики утицај на процес образовања. Појава интернета и развој интернет технологија омогућили су значајан напредак у области учења на даљину, као и е-образовање.

Учење на даљину представља наставни процес код кога је планирано учење просторно одвојено од процеса предавања и који захтева посебне технике планирања образовног процеса, специјалне методе предавања, посебне технике комуникације применом електронских и других технологија, као и организациона и административна прилагођавања [12]. На почетку, учење на даљину се одвијало путем поште, радија, телевизије и аудио-трака [13]. Данас, са развојем интернет технологија примена е-образовања постаје све значајнија. Е-образовање (енг. *electronic education, e-education* или *e-learning*) представља вид реализације наставног процеса коришћењем електронских медија за испоруку различитих наставних материјала, активности и програма учења [14]. Друга дефиниција наводи да е-образовање представља испоруку различитих лекција, инструкција и курсева применом интернет технологија [15]. Е-образовање, у општем случају, описује учење уз помоћ рачунара, али са развојем нових технологија овај вид учења подразумева и коришћење различитих преносивих уређаја, као што су мобилни телефони, таблети итд. [16]. Е-образовање се у пракси често користи у комбинацији са традиционалним образовањем, при чему се добија „*blended learning*“ концепт [17], [18] који омогућава ефикасније окружење за учење и предавање.

Основна компонента система е-образовања је систем за управљање учењем (енг. *Learning Management Systems – LMS*). Системи за управљање учењем представљају софтверско решење за реализацију е-образовања и имају задатак да обезбеде скуп алата за реализацију и управљање процесом образовања [19].

Универзитет у Лондону је 1858. године увео учење на даљину [20], а данас велики број универзитета реализује наставни процес применом овог концепта. Један део светских универзитета реализује и бесплатне онлајн курсеве применом концепта е-образовања [21].

Крајњи циљ образовног процеса је повећати ниво знања и мотивисаности студената и олакшати припрему и реализацију наставног процеса наставном особљу, а примена концепта учења на даљину и е-образовања утиче на побољшање образовног процеса у поређењу са традиционалним приступом [22].

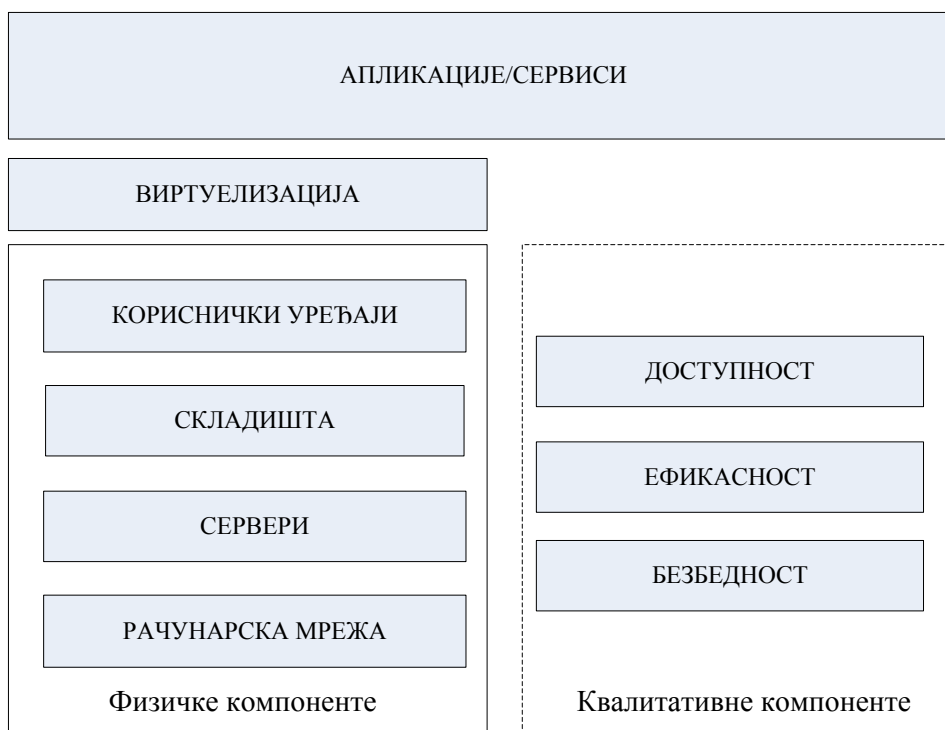
2.2. Појам и дефиниција ИТ инфраструктуре

ИТ инфраструктура представља основу сваког система, а у зависности од угла посматрања може се дефинисати на различите начине. Инфраструктура се може дефинисати као јединствена платформа за развој и извршавање апликација, која обухвата све мрежне уређаје и њихове конекције [23]. Мрежни администратори посматрају ИТ инфраструктуру као скуп мрежних уређаја, док је за програмере ИТ инфраструктура платформа за развој апликација [24]. ИТ инфраструктуру чине следећи елементи: ИТ компоненте, људски ресурси, ИТ сервиси и апликације [25].

ИТ инфраструктура образовних институција има задатак да подржи велики број образовних сервиса и апликација и омогући реализацију наставног процеса и сарадњу студената и наставног особља. Поред подршке за традиционални наставни процес, неопходно је постојање инфраструктуре која може да подржи и учење на даљину и е-образовање. Зато се посебна пажња посвећује развоју и изградњи академских мрежа, а пре свега одговарајуће инфраструктуре. Разлог за то је позитиван утицај на велики број академских грађана, али и на друштво у целини [26].

2.2.1. Компоненте ИТ инфраструктуре

ИТ инфраструктура представља скуп хардвера и софтвера, који има задатак да обезбеди различите сервисе крајњим корисницима. На слици 1 су приказане основне компоненте архитектуре ИТ инфраструктуре.



Слика 1: Компоненте ИТ инфраструктуре

Разликују се две групе компоненти: физичке и квалитативне. Физичке компоненте представљају основу за функционисање једног система, а у ову групу спадају: рачунарска мрежа, сервери, складишта и крајњи кориснички уређаји.

Између слоја хардвера и софтвера се налази слој виртуелизације, који врши симулацију одређеног мрежног ресурса. Овај слој има задатак да повећа степен ефикасности као квалитет ИТ инфраструктуре [27].

Квалитативне компоненте описују квалитет инфраструктуре, а међу њима се издвајају: доступност, ефикасност и безбедност. Доступност даје одговор на

питање да ли ИТ инфраструктура има одговарајући ниво перформанси у дефинисаном временском интервалу. Однос нивоа перформанси инфраструктуре и неопходних ресурса представља ефикасност. Посебно је важна безбедност ИТ инфраструктуре, пошто има задатак да заштити комуникацију, податке и уређаје. Поред наведених компоненти потребно је нагласити и значај примене одговарајућег система за управљање ИТ инфраструктуром, како би били испуњени захтеви у погледу доступности, ефикасности и безбедности једног система [28].

2.3. *Cloud computing* инфраструктура

ИТ инфраструктура образовних институција има комплексан задатак да одговори на различите захтеве студената и наставног особља. Једна од важних карактеристика оваквог окружења јесте скалабилност. Традиционалне рачунарске мреже, реализоване применом хијерархијског модела, могу само у извесној мери да испуне овај захтев [29]. Међутим, са развојем нових технологија појавило се боље решење у виду *cloud computing* технологије [30]. Ово решење може да превазиђе проблеме који настају као последица повећања броја корисника образовног система, имплементације нових сервиса, повећања количине података и различитих процеса, као што су моделирање, симулације, интерактивни видео и виртуелни светови [2]. Управо из ових разлога *cloud computing* налази примену у великом броју академских мрежа [8].

Cloud computing се може дефинисати на различите начине, али оно што је суштина ове технологије јесте апстракција мрежних ресурса доступних „на захтев“ коришћењем интернета [31]. Друга дефиниција каже да је *cloud* група дистрибуираних рачунара који пружају потребне мрежне ресурсе, као и сервисе према захтеву путем мрежног медија [8]. Важна карактеристика ове технологије јесте чињеница да крајњи корисници не морају бити упознати са локацијом мрежних уређаја и ресурса којима приступају. Амерички Национални институт за стандарде и технологију дефинише *cloud* као „модел који омогућава свеprisутни

и одговарајући приступ мрежи „на захтев“ дељеном скупу рачунарских ресурса (као што су мрежа, сервери, складишта, апликације и сервиси) који могу да се омогуће крајњим корисницима уз минималан напор и минималну интеракцију са провајдером сервиса“ [32]. Основу функционисања *cloud computing*-а представља технологија виртуелизације, захваљујући којој се остварује апстракција мрежних ресурса [33].

Cloud computing инфраструктура према начину коришћења мрежних ресурса разликује три приступа [34]:

- Инфраструктурни (енг. *Infrastructure as a Service – IaaS*)
- Платформски (енг. *Platform as a Service – PaaS*)
- Апликациони (енг. *Software as a Service – SaaS*)

Инфраструктурни приступ обезбеђује виртуелизовану инфраструктуру одређеног капацитета и перформанси на захтев корисника. Платформски приступ омогућава креирање апликације применом различитих развојних алата на платформи провајдера. Апликациони приступ омогућава коришћење готових апликационих решења, при чему је комплетна хардверска и софтверска инфраструктура у надлежности провајдера. Иако сви наведени типови *cloud*-а могу наћи примену у академском окружењу, PaaS има посебан значај [35] када је у питању образовање у области рачунарства и информационо-комуникационих технологија.

Према власништву *cloud computing* инфраструктура се може поделити на четири групе [36]:

- Јавни облак (енг. *Public Cloud*)
- Приватни облак (енг. *Private Cloud*)
- Хибридни облак (енг. *Hybrid Cloud*)
- Заједнички облак (енг. *Community Cloud*)

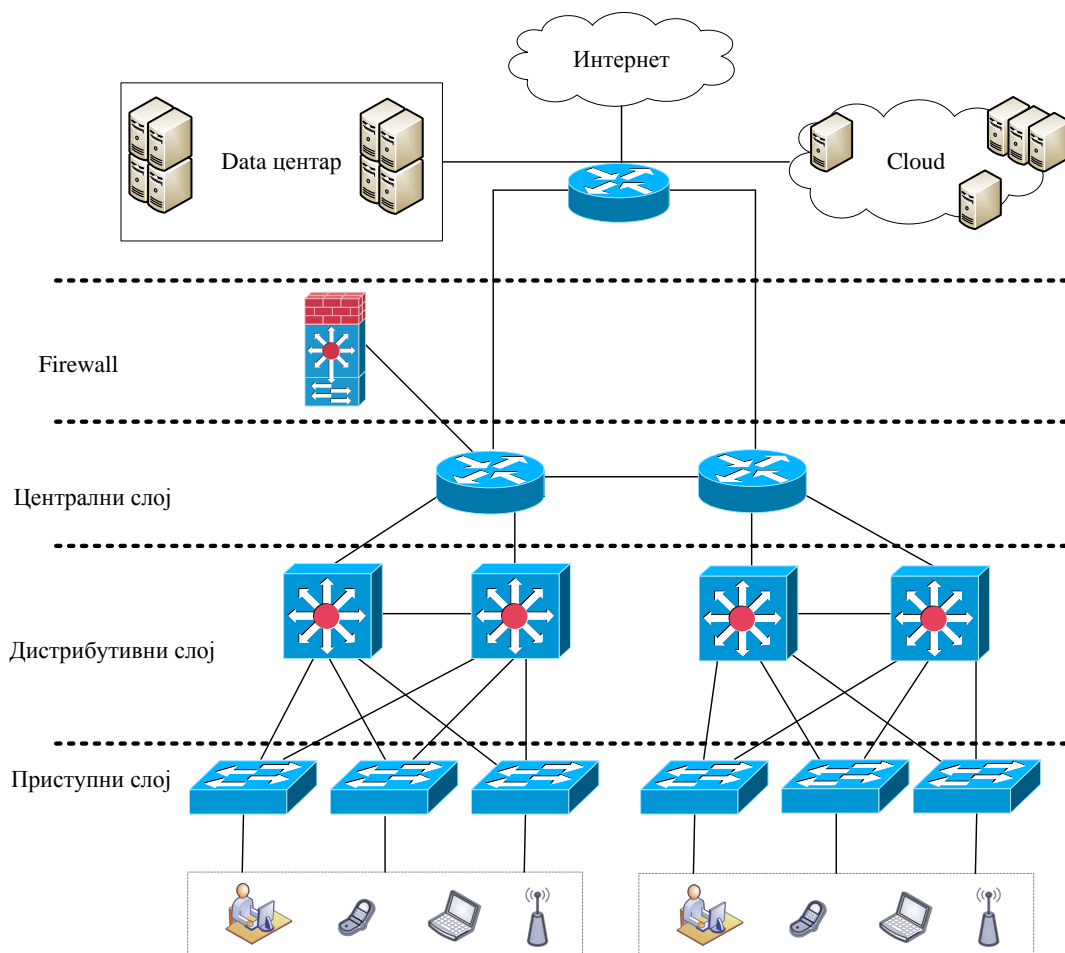
Образовне институције могу користити било који од ових модела, па велики број академских институција поседује приватни облак [37].

Предности примене инфраструктуре за *cloud computing* су вишеструке. Ова технологија омогућава висок степен скалабилности, могућност приступа ресурсима када за то постоји потреба, еластичност ресурса, мерљивост услуга [38], као и смањење трошкова одржавања ИТ инфраструктуре [39]. *Cloud computing* представља релативно нову технологију, али захваљујући својим позитивним карактеристикама налази примену у великом броју предузећа, у јавној управи [40], али и у многим академским институцијама [41], [42]. Као и све технологије и *cloud computing* има одређене недостатке. Основни недостаци се односе на питање безбедности података и комуникацију између крајњих корисника и провајдера сервиса [43], што је у директној вези са недостатком правне регулативе. Праћење перформанси *cloud computing* инфраструктуре у циљу правовремене детекције проблема и обезбеђивања одговарајућег нивоа сервиса (енгл. *Service Level Agreement – SLA*) представља велики изазов [44], [45].

2.4. Разлози за увођење софтверски дефинисаних мрежа

Захваљујући развоју информационо-комуникационих технологија, захтеви који се постављају пред рачунарске мреже су постали комплексни. Поред стандардних захтева у погледу скалабилности, доступности и безбедности, мрежна инфраструктура мора да обезбеди и висок степен мобилности корисника и да подржи BYOD (енг. *Bring Your Own Device*) принцип, односно приступ одговарајућим мрежним ресурсима у сваком тренутку и без ограничења у смислу приступних уређаја. Пораст количине података у мрежама представља још један фактор који утиче на процес планирања и имплементације данашњих мрежа. Посебан изазов представља и реализација управљања таквом мрежном инфраструктуром [46], [47].

Традиционалне рачунарске мреже су засноване на хијерархијском моделу, који целокупну мрежу посматра као низ слојева, са прецизно дефинисаним функционалностима на сваком слоју. На слици 2 је приказана традиционална кампус мрежна инфраструктура.



Слика 2: Традиционална кампус мрежна инфраструктура

Оваква архитектура обезбеђује извешан степен скалабилности и модуларности, пре свега захваљујући примењеном концепту редувантности [48]. Међутим, традиционалне мреже карактерише статичност, као и чињеница да су хетерогене, односно примењује се опрема различитих произвођача, чиме се процес одржавања и управљања мрежом усложњава. Процес увођења нових сервиса додатно успорава комплексност реализације, односно потреба познавања великог броја мрежних технологија и конфигурација великог броја уређаја. Важно је нагласити и постојање многих нестандардизованих решења, као и затвореност мрежне опреме [7]. Многе мрежне функционалности су дефинисане у хардверу, што за последицу има високу цену реализације мрежне архитектуре, као и недовољан степен флексибилности. Наведени недостаци традиционалних рачунарских мрежа

представљају разлог за увођење динамичке архитектуре у модерне рачунарске мреже.

Динамичка архитектура подразумева примену техника виртуелизације мрежних ресурса и софтверски дефинисаних мрежа за централизовано управљање виртуелном и физичком инфраструктуром. Виши степен програмабилности у мрежама значи побољшање флексибилности и могућности прилагођавања потребама корисника кроз примену грануларних полиса управљања, а све то коришћењем јединствене мрежне инфраструктуре. Централизовано управљање омогућава увид у стање у целокупној мрежи чиме се олакшава процес оптимизације потребних ресурса, а тиме смањују и оперативни трошкови. Могућност логичке доделе мрежних ресурса на захтев представља важну карактеристику у кампус окружењу, а посебно када је реч о академским мрежама. *Cloud* технологија је присутна у многим кампус мрежама, па је у том случају процес аутоматизације битан. SDN архитектура поједностављује процес аутоматизације захваљујући апстракцији мрежних ресурса, чиме се знатно смањује време имплементације нових сервиса [49].

Примена динамичке архитектуре, односно програмабилности и централизованог управљања у мрежама, има задатак да обезбеди следеће [50]:

- Брза примена нових сервиса без утицаја на друге логичке мреже захваљујући виртуелизацији.
- Повећање доступности сервиса због могућности постојања предефинисаних алтернативних путања, што знатно смањује време конвергенције у поређењу са традиционалним мрежама.
- Изолација саобраћаја логичких мрежа на другом и трећем слоју OSI модела.
- Оптимално коришћење ресурса пошто су управљање, као и сервис и апликације, виртуелизовани како би се оптимално користили простор и напајање.

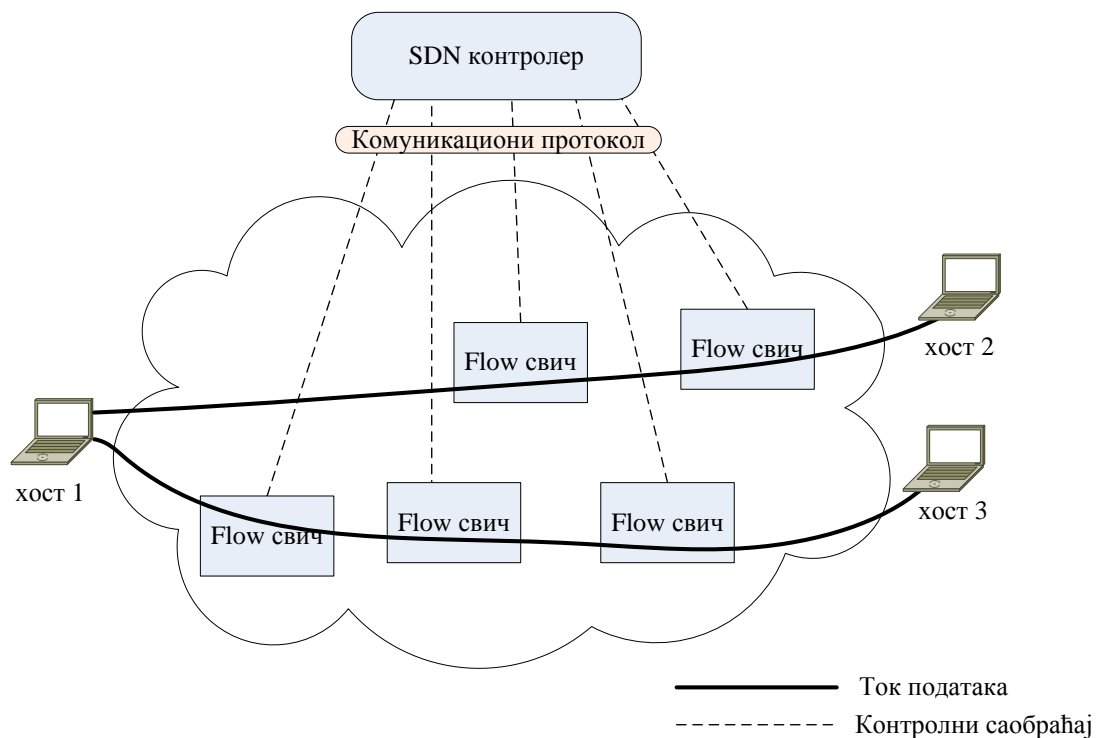
3. СОФТВЕРСКИ ДЕФИНИСАНЕ МРЕЖЕ

3.1. Појам и дефиниција софтверски дефинисаних мрежа

Први корак ка развоју софтверски дефинисаних мрежа представља *Ethane* пројекат на Универзитету Станфорд [51], који је увео централизацију управљања процесом прослеђивања мрежног саобраћаја. На развој и примену програмабилности у рачунарским мрежама је утицао и *OpenFlow* протокол, као први протокол за комуникацију између функције управљања мрежом и функције прослеђивања саобраћаја [11]. Програмабилност као концепт у рачунарским мрежама постоји преко 20 година [52].

Софтверски дефинисане мреже представљају парадигму у рачунарским мрежама, која има задатак да превазиђе недостатке традиционалног умрежавања [10]. Традиционалне мреже подразумевају вертикалну интеграцију функционалности, што значи да се функција управљања (енг. *control plane*) и прослеђивања података (енг. *data plane*) налазе у оквиру самих мрежних уређаја, што може бити комплексно са становишта управљања у великим мрежама.

Концепт софтверски дефинисаних мрежа користи хоризонталну интеграцију, која подразумева раздвајање процеса управљања и прослеђивања података [53], што је приказано на слици 3. Уместо дистрибуираног управљања, које постоји у традиционалним мрежама, прелази се на концепт централизованог управљања мрежом кроз примену контролера. Поред овога, важна карактеристика софтверски дефинисаних мрежа је и примена интерфејса отвореног стандарда за комуникацију. На овај начин се управљање мрежом реализује коришћењем контролера уместо већег броја уређаја различитих произвођача опреме и различитих протокола. Захваљујући наведеним особинама значајно је поједностављен процес креирања и управљања мрежном инфраструктуром, као и процес локализације и решавања уочених проблема [54], примена нових протокола и апликација и виртуелизација мреже [55].



Слика 3: Концепт софтверски дефинисаних мрежа

Постоји више дефиниција концепта софтверски дефинисаних мрежа, али оно што је заједничко за све јесте апстракција нижих слојева архитектуре [9]. *Open Network Foundation* наводи да софтверски дефинисане мреже омогућавају директно управљање и програмабилност рачунарских мрежа раздвајањем функција управљања и прослеђивања података [4]. Друга дефиниција наводи да SDN приступ омогућава интеракцију и манипулацију апликација и управљачких софтвера мрежних уређаја и ресурса [56].

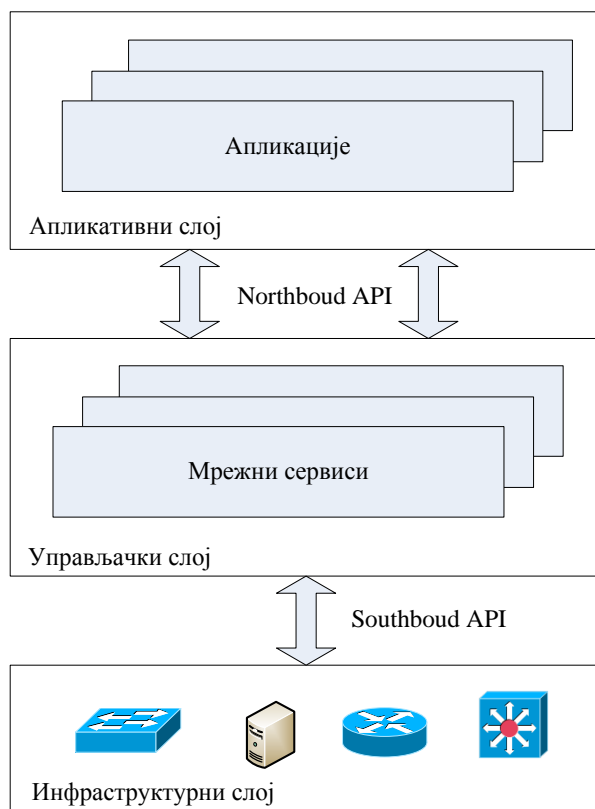
Свака мрежна инфраструктура мора да испуни захтеве у погледу скалабилности, интероперабилности и безбедности, а софтверски дефинисане мреже могу значајно да допринесу испуњењу ових захтева [57]. Централизација, као кључна карактеристика софтверски дефинисаних мрежа, поједностављује управљање мрежом и омогућава примену мање комплексних мрежних уређаја, али са друге стране може да доведе до појаве да контролер постаје *single-point-of-failure* [57]. Међутим, ова ситуација се може избећи применом дистрибуираног контролера и одговарајућег протокола за комуникацију између контролера, као што је SDNi

протокол [58], [59]. У великим мрежама може бити изражен проблем скалабилности, који примена дистрибуираних контролера [60] може да реши. Иако уводи виши степен флексибилности и омогућава нове функционалности, SDN концепт има и извешан утицај на перформансе [61], па је неопходно применити одговарајући систем за праћење перформанси.

Примена новог концепта, као што је SDN, захтева извесне финансијске трошкове, па се у многим мрежама примењује хибридни приступ, односно комбинација традиционалног и софтверски дефинисаног окружења.

3.2. Архитектура софтверски дефинисаних мрежа

Архитектура софтверски дефинисаних мрежа разликује три слоја (Слика 4): инфраструктурни, управљачки и апликативни [62].



Слика 4: Архитектура софтверски дефинисаних мрежа [4]

Инфраструктурни слој чине мрежни елементи, који су задужени за прослеђивање саобраћаја. Они могу бити реализовани у хардверу или софтверу, односно могу бити физички или виртуелни. Мрежни елементи доносе одлуку о прослеђивању саобраћаја на основу инструкција добијених од управљачког слоја.

Комуникација између инфраструктурног и управљачког слоја се остварује преко *southbound* API интерфејса. Његов задатак је да пренесе инструкције о прослеђивању пакета од управљачког слоја до мрежних елемената који ће извршити одређену акцију. То може бити прослеђивање пакета, одбацивање пакета или измена пакета. Пример протокола за комуникацију између инфраструктурног и управљачког слоја је *OpenFlow* [63].

Контрола и надзор прослеђивања саобраћаја у мрежи је функционалност управљачког слоја. На овом слоју се налазе SDN контролери задужени за контролу мрежних ресурса. Иако је функција управљања мрежом логички централизована, у мрежи може постојати више SDN контролера. SDN контролер има функцију оркестрације, односно координације мрежним ресурсима и његов задатак је да реализују захтеве дефинисане на апликативном слоју.

Комуникација између управљачког и апликативног слоја се остварује преко *northbound* API интерфејса. Примери сервисних интерфејса су *RESTful* API-ји, протоколи отвореног кода као што је NETCONF, CORBA интерфејси итд. Два најзначајнија приступа су *RESTful* и RPC (енг. *Remote Procedure Call*) интерфејси, који су засновани на клијент-сервер комуникацији и за пренос порука користе XML или JSON (енг. *JavaScript Object Notation*) [64].

Највиши слој је апликативни и њега чине апликације и сервиси који дефинишу понашање мреже коришћењем сервиса управљачког слоја. Примери апликација су откривање топологије мреже, конфигурација мрежних елемената, резервација путања кроз мрежу итд.

Комплетно управљање мрежном инфраструктуром је у надлежности SDN контролера, тако да се са становишта апликативног слоја цела мрежа посматра као логички свич [65]. На овај начин процес пројектовања рачунарских мрежа постаје мање комплексан. Задатак мрежних уређаја је да остваре комуникацију са SDN контролером, што значи да не постоји потреба да они буду задужени за комплетан процес прослеђивања саобраћаја, односно доношења одлуке о прослеђивању пакета и само прослеђивање. Архитектура мрежних уређаја може бити једноставна, пошто треба да подржи мањи број мрежних протокола [66], а то утиче и на смањење цене реализације мрежних уређаја. Централизација на управљачком слоју пружа флексибилност и приликом конфигурације и управљања применом динамичких SDN програма [67].

3.3. SDN контролери

Мрежна интелигенција је централизована у софтверским SDN контролерима или NOS (енгл. *Network Operating System*), тако да се целокупна мрежа, са становишта апликативног слоја, посматра као један логички свич. Овај концепт поједностављује процес пројектовања рачунарских мрежа и омогућава примену мање комплексних мрежних уређаја. Постоји изванредан број различитих SDN контролера, а у наставку ће бити описани најзначајнији.

NOX је први *OpenFlow* контролер развијен у оквиру *Nicira Networks* [68]. Написан је у *C++* и *Python*-у и омогућава подршку за мреже великих предузећа са више стотина свичева, али и за мале мреже са неколико хостова. Више се не ради на развоју овог контролера, већ је креиран *POX*.

POX контролер је настао на основу *NOX* контролера и написан је у *Python*-у [69]. Има подршку за *Linux*, *MAC OS* и *Windows* оперативне системе. У поређењу са *NOX* апликацијама у *Python*-у има боље перформансе. Активно се ради на развоју овог контролера и има значајну улогу у области истраживања и едукацији.

Beacon је *OpenFlow* контролер развијен 2010. године [70]. У питању је решење отвореног кода доступно под комбинацијом GPLv2 лиценце и *Stanford University FOSS License Exception v1.0*. Написан је у *Java* програмском језику и има примену у области образовања и истраживања. Овај контролер представља основу за развој *Floodlight* контролера.

Floodlight је *OpenFlow* контролер отвореног кода написан у *Java* програмском језику [71]. Може се применити у мрежама у којима постоји комбинација *OpenFlow* и *non-OpenFlow* мрежа. Велики број комерцијалних свичева (*Brocade*, *Dell*, *HP* итд.) је компатибилан са овим контролером.

Наведени контролери спадају у групу централизованих контролера што значи да представљају централно место са ког се врши комплетно управљање мрежом. У овом случају контролер представља *single-point-of-failure*, а у већим мрежама до изражаја може доћи и проблем скалабилности. Зато постоји и друга група SDN контролера, а то су дистрибуирани контролери. У овом случају функција управљања мрежом је дистрибуирана, чиме се решава проблем скалабилности, а у случају отказа контролера спречава велики утицај на целу мрежу. Међутим, проблем конзистентности конфигурације постаје изражен и одражава се на перформансе мреже [60]. Примери дистрибуираних контролера су: *Onix* [72], *HiperFlow* [73], *HP VAN SDN* [74].

OpenDaylight [75] је SDN платформа настала као резултат сарадње више произвођача опреме: *Brocade*, *Cisco Systems*, *IBM*, *HP*, *Juniper*, *Microsoft*, *Citrix* итд. У питању је *Linux* решење за SDN контролер које садржи интегрисану *OpenStack Neutron plug-in* архитектуру [76]. Ово решење има значај у области развоја софтверски дефинисаних мрежа и активно се ради на развоју и унапређењу ове платформе.

3.4. Протоколи у софтверски дефинисаним мрежама

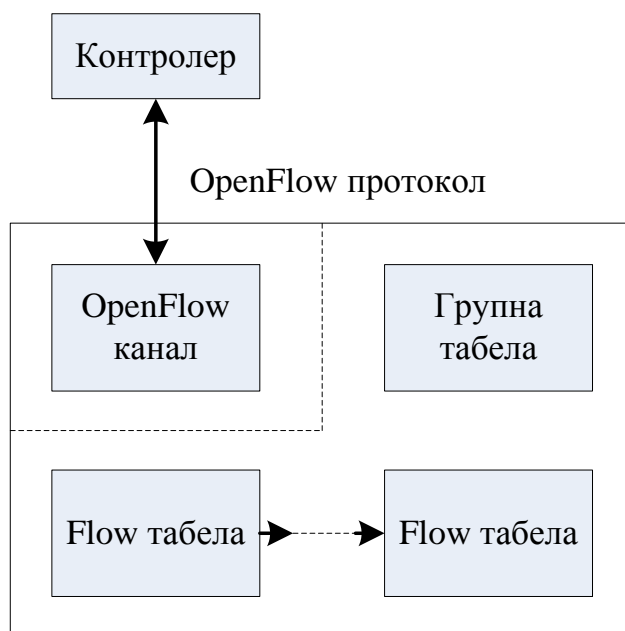
3.4.1. *OpenFlow* протокол

Први стандардизовани *southbound* API интерфејс у SDN мрежама је *OpenFlow* [77]. Овај протокол врши препознавање саобраћаја на основу статички или динамички предефинисаних правила у SDN управљачком софтверу. Захваљујући концепту токова података (енг. *flow*) омогућен је висок степен грануларности [78] на нивоу апликације, корисника и сесије, што није карактеристика традиционалних мрежа у којима сви подаци у оквиру једног тока података морају имати исту путању кроз мрежу.

OpenFlow протокол се може сматрати кључним елементом SDN мрежа [79], пошто је тренутно једини стандардизован протокол који омогућава манипулацију подацима у равни прослеђивања. Такође, може се применити и у мрежама са опремом различитих произвођача, што је од изузетног значаја, пошто све већи број компанија прихвата концепт софтверски дефинисаних мрежа [80].

OpenFlow логички свич чини једна или више *flow* табела и групна табела, на основу којих се врши прослеђивање саобраћаја и један или више *OpenFlow* канала ка екстерном контролеру (Слика 5).

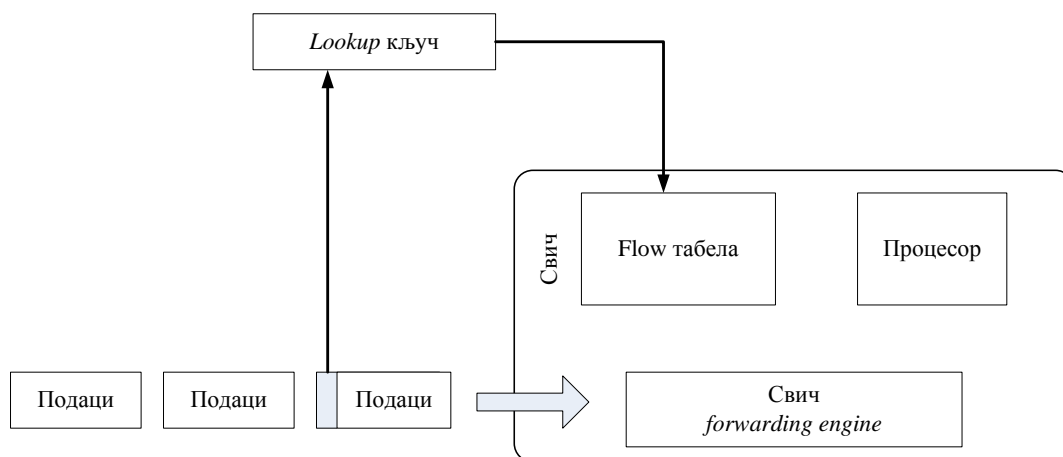
Коришћењем *OpenFlow* протокола контролер врши измене записа о токовима података у *flow* табелама, при чему свака табела садржи и низ инструкција на основу којих се врши прослеђивање саобраћаја [78].



Слика 5: Компоненте *OpenFlow* свича [78]

Прослеђивање саобраћаја се врши у неколико корака (Слика 6):

- Подаци долазе до *OpenFlow* свича
- Као *lookup* кључ се користе поља из заглавља пакета
- Ако не постоји запис у *flow* табели, контролер програмира *flow* табелу свича
- *Forwarding engine* врши прослеђивање пакета.



Слика 6: Прослеђивање саобраћаја код *OpenFlow* свича

Сваки запис у *flow* табели садржи следећа поља:

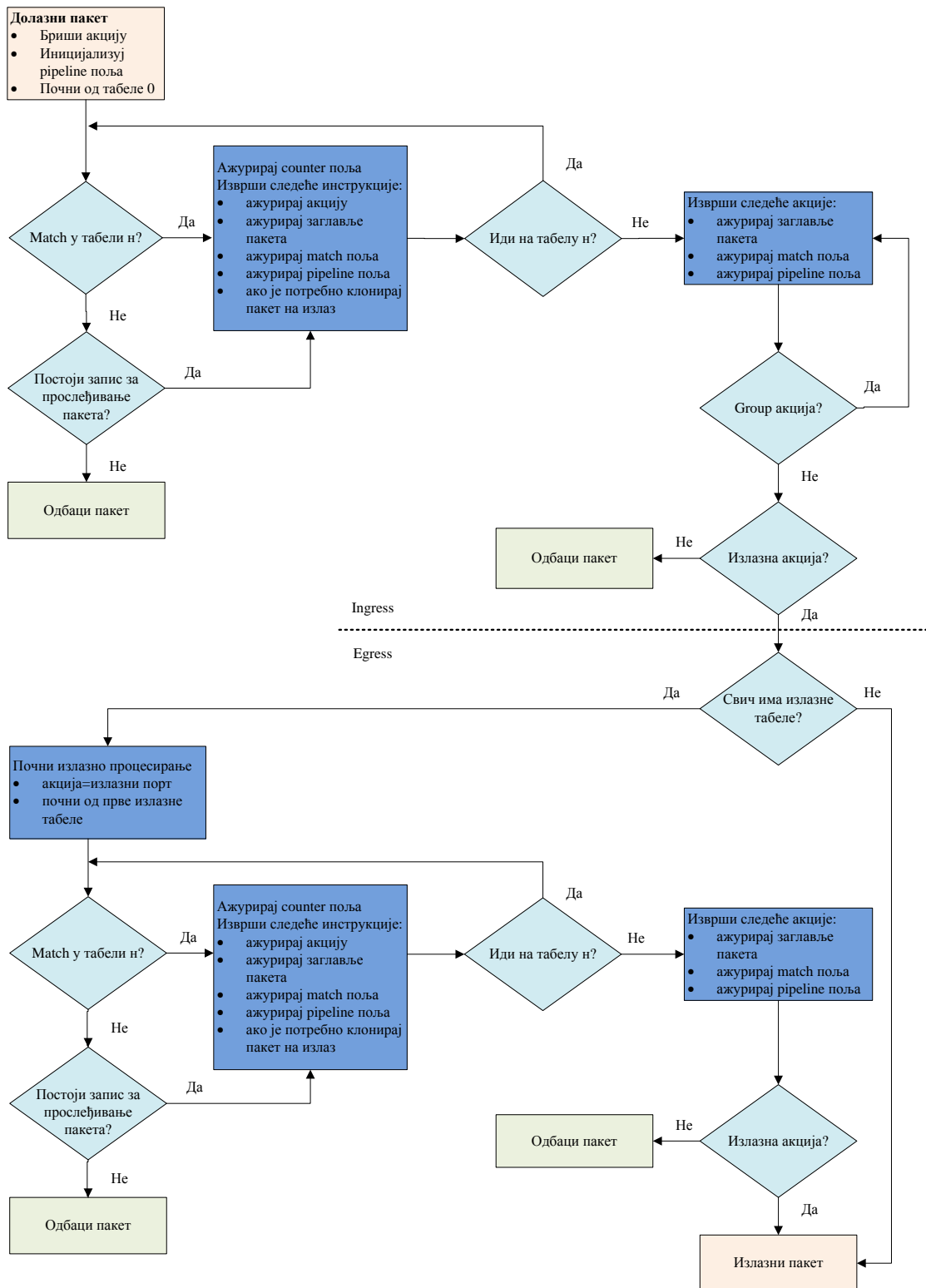
- *match fields* – поља на основу којих се врши *lookup*; чине их улазни порт и заглавља пакета, а опционо може да садржи и друга поља.
- *priority* – дефинише приоритет записа у *flow* табели.
- *counters* – бројачи који се ажурирају када постоји поклапање (*match*).
- *instructions* – акције за процесирање саобраћаја.
- *timeouts* – максималан временски период важења записа у *flow* табели.
- *cookie* – подаци који се не користе приликом процесирања саобраћаја, већ их контролер може користити за филтрирање записа у *flow* табели на основу захтева са статистичким подацима и сл.
- *flags* – флегови који мењају начин управљања записима у *flow* табели.

Сваки запис у *flow* табели се једнозначно идентификује на основу *match fields* и *priority* вредности.

Слика 7 приказује ток података кроз *OpenFlow* свич.

Свичеви који имају подршку за *OpenFlow* протокол могу бити *OpenFlow-only* и *OpenFlow-hybrid*. *OpenFlow-only* свичеви подржавају искључиво *OpenFlow* протокол, односно саобраћај могу да процесирају само на горе описани начин применом *OpenFlow* протокола. За разлику од њих, постоје и *OpenFlow-hybrid* свичеви, који саобраћај могу да процесирају коришћењем *OpenFlow* протокола или на стандардан начин применом *Ethernet switching*-а, што подразумева класичан *Ethernet switching* на другом слоју OSI модела, примену VLAN-ова (енг. *Virtual Local Area Network*), рутирање на трећем слоју итд. У овом случају неопходно је да постоји механизам за класификацију саобраћаја како би се процесирање извршило на традиционалан или *OpenFlow* начин.

OpenFlow v1.0 представља иницијални стандард и он подржава један *lookup*, а од верзије *OpenFlow* протокола 1.1 постоји подршка за вишеструки *lookup*. Последња верзија *Open Flow* протокола је 1.5.1.



Слика 7: Ток података кроз *OpenFlow* свич [78]

3.4.2. OVSDB протокол

OVSDB (енг. *Open vSwitch Database Management Protocol*) представља други протокол за програмабилан приступ мрежним уређајима [81]. Модел прослеђивања података код *Open vSwitch*-а је сличан ономе по коме функционише *OpenFlow*. Поред могућности да се утиче на карактеристике тока података, OVSDB има и додатне могућности, као што су дефинисање QoS (енг. *Quality of Service*) полиса, повезивање интерфејса са свичевима или прикупљање статистичких података, па се из тог разлога OVSDB и *OpenFlow* протоколи могу сматрати комплементарним [82].

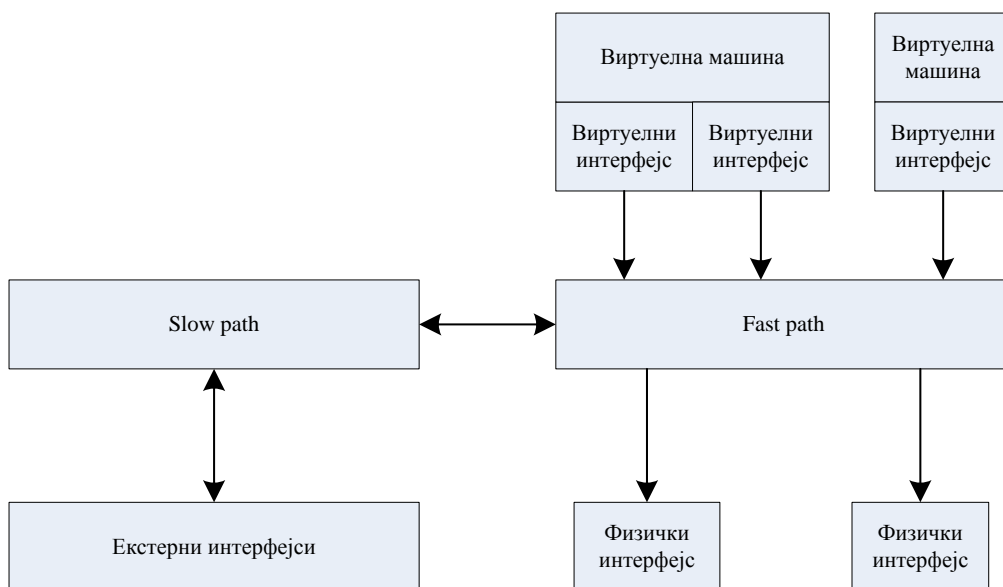
Open vSwitch је софтверски свич за хипервизоре, који је креиран са циљем да обезбеди комуникацију између виртуелних и физичких ресурса [27] и да подржи стандардне интерфејсе за управљање уз додатну могућност програмабилног управљања функцијом прослеђивања података [83]. Представља пројекат отвореног кода и доступан је под *Apache 2* лиценцом. *Open vSwitch* подржава различите технологије виртуелизације као што су *Xen*, *KVM* и *VirtualBox*.

Open vSwitch има задатак да обезбеди комуникацију између виртуелних машина и физичких интерфејса. *Open vSwitch* подржава читав низ технологија [84]:

- Мониторинг саобраћаја виртуелних машина (sFlow, NetFlow, SPAN, RSPAN)
- LACP (енг. *Link Aggregation Control Protocol*)
- подршку за 802.1Q VLAN-ове и *trunking*
- BFD (енг. *Bidirectional Forwarding Detection*)
- STP (енг. *Spanning Tree Protocol*) и RSTP (енг. *Rapid Spanning Tree Protocol*)
- QoS
- *OpenFlow* протокол
- IPv6
- Протоколе за креирање тунела (GRE, VXLAN)
- *Kernel* и *userspace* имплементацију свича.

Open vSwitch подржава *Ethernet* технологију и примену VLAN-ова, али даје и могућност интеграције у виртуелна окружења и управљања прослеђивањем података преко различитих интерфејса [27]:

- *Configuration* – преко овог интерфејса се врше измене конфигурације. Тренутно су подржани *port mirroring*, QoS полисе, *NetFlow* логовање и примена *bond* интерфејса. Поред могућности удаљене конфигурације, преко овог интерфејса се остварује и веза између мрежних портова и виртуелног окружења.
- *Forwarding path* – преко овог интерфејса се врши дефинисање путање прослеђивања података. На овај начин екстерни процеси могу директно да мењају табелу прослеђивања и тако дефинишу путању на основу L2, L3 или L4 заглавља пакета.
- *Management* – локални интерфејс за управљање преко кога се може мењати конфигурација топологије (креирање свичева, управљање виртуелним интерфејсима).

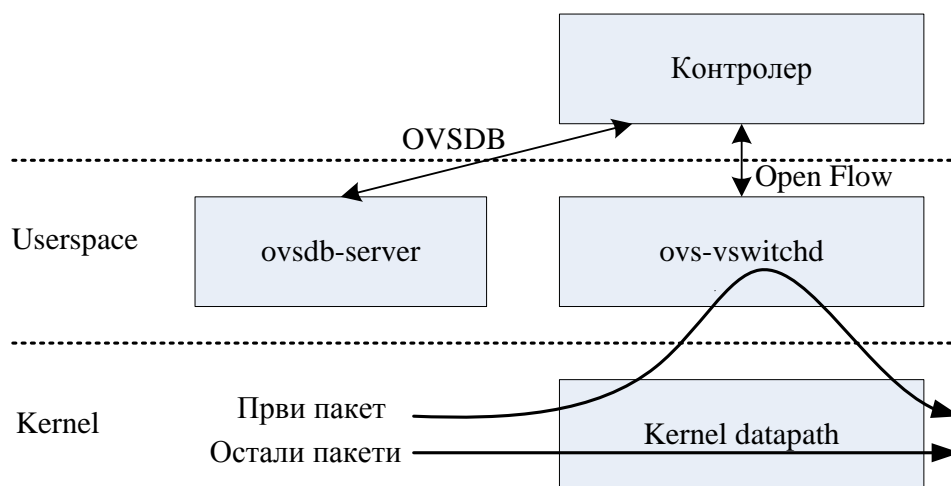


Слика 8: *Open vSwitch* архитектура [27]

На слици 8 је приказана архитектура *Open vSwitch*-а. *Open vSwitch* имплементација има две компоненте: *fast path* у кернелу и кориснички *slow path*.

Fast path је задужен за *lookup* и прослеђивање саобраћаја, док је већина функционалности имплементирана у оквиру *slow path* дела, који се покреће у оквиру виртуелне машине. *Slow path* имплементира логику за прослеђивање и интерфејсе за конфигурацију, као што су *OpenFlow*, *NetFlow* итд.

Модел прослеђивања података код *Open vSwitch*-а је сличан ономе по коме функционише *OpenFlow*. На слици 9 је приказан модел прослеђивања саобраћаја *Open vSwitch*-а.



Слика 9: *Open vSwitch* модел прослеђивања саобраћаја [85]

Први пакет из *flow*-а се шаље *ovs-vswitchd* кернел модулу, који га затим прослеђује *datapath* модулу, који чува правила за прослеђивање наредних пакета [85]. Из тог разлога је време прослеђивања првог пакета из *flow*-а нешто дуже у односу на остале пакете из *flow*-а. Пакет долази преко мрежне карте до *datapath* модула, који извршава инструкције добијене од *ovs-vswitchd* модула. Инструкцијама су дефинисане акције за прослеђивање пакета, као што су прослеђивање на одређени порт, модификација пакета или одбацивање пакета. Уколико *datapath* модул нема инструкције за прослеђивање, пакет се прослеђује *ovs-vswitchd* кернел модулу, који одлучује о начину процесирања пакета и те инструкције заједно са пакетом враћа *datapath* модулу.

На овај начин је дозвољено дефинисање функција прослеђивања саобраћаја који потиче од једног виртуелног интерфејса, једне или више виртуелних машина. Увођење интерфејса за глобално управљање конфигурацијом и прослеђивањем саобраћаја долази до одвајања логичке топологије од физичке.

Основне компоненте *Open vSwitch* дистрибуције су [83]:

- *ovs-vswitchd* – *daemon* који имплементира свич и кернел модул за *flow-based* прослеђивање саобраћаја.
- *ovsdb-server* – база података у којој се чува конфигурација свича.
- *ovs-dpctl* – алат за конфигурацију кернел модула.
- *ovs-vsctl* – алат за ажурирање конфигурације *ovs-vswitchd daemon*-а.
- *ovs-appctl* – алат за слање команди *Open vSwitch daemon*-у.

Поред наведених постоје и додатни алати:

- *ovs-ofctl* – за управљање *OpenFlow* свичевима и контролерима
- *ovs-pki* – за креирање и управљање *public-key* инфраструктуром за *OpenFlow* свичеве
- *patch* за *tcpdump* који омогућава парсирање *OpenFlow* порука.

Захваљујући принципима виртуелизације, може се рећи да у модерним информационим системима долази до проширења саме мреже до нивоа хоста, где се налази *Open vSwitch*. Уколико је у питању потпуно виртуелизовано окружење, сваки мрежни чвор је у ствари виртуелни свич. На овај начин је могуће направити потпуно раздвајање *core* и *edge* дела мрежне инфраструктуре уз постојање комуникације између ових делова. Виртуелни свичеви могу имплементирати велики број мрежних функционалности, чиме је омогућено да *core* инфраструктура буде једноставна и да има основни задатак да обезбеди конективност између хостова [27].

3.5. Модели реализације софтверски дефинисаних мрежа

Концепт софтверски дефинисаних мрежа у пракси се примењује у форми једног од следећих модела [86]:

- Реактивни
- Проактивни
- Предиктивни.

Реактивни модели су модели који се обично повезују са софтверски дефинисаним мрежама и *OpenFlow* протоколом и карактерише их висок степен прилагођавања и тренутне реакције на стање у мрежи. Могу наћи примену у окружењима где је честа промена локације крајњих уређаја (виртуелних машина), а недостатак је недовољан степен скалабилности. Проактивни модели имају задатак да предвиде проблеме који се могу јавити у мрежи и да покушају да их реше пре ескалације. Предиктивни модели су засновани на историјским подацима о перформансама мреже и на основу њих врше измене токова података.

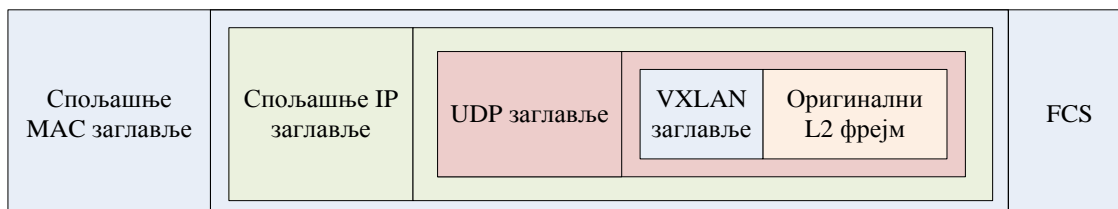
SDN архитектура се може реализовати применом комерцијалних или решења отвореног кода.

3.5.1. Реализација софтверски дефинисаних мрежа у *Linux* окружењу

Linux оперативни систем омогућава примену различитих технологија и техника за креирање виртуелног мрежног окружења.

Виртуелна локална рачунарска мрежа VLAN (енг. *Virtual Local Area Network*) представља једноставну технологију за сегментацију рачунарске мреже. Постојећа мрежна инфраструктура се на логичком, софтверском нивоу дели на *broadcast* домене независно од физичке топологије чиме се позитивно утиче на безбедност и перформансе мреже.

Недостатак VLAN концепта, који се огледа у недовољном степену скалабилности у случају већих мрежа, успешно се може решити применом VXLAN (енг. *Virtual Extensible LAN*) технологије (Слика 10), која подржава креирање до 16 милиона VXLAN мрежних сегмената [87].



Слика 10: Формат VXLAN фрејма

Значајну примену у мрежама у којима постоји потреба за имплементацијом виртуелизације имају и софтверски реализовани виртуелни свичеви. Они се реализују у оквиру хипервизора сервера и задужени су за управљање и комуникацију између виртуелних машина и физичког окружења. Пример софтверског свича који налази примену у великом броју мрежних окружења је *Open vSwitch*.

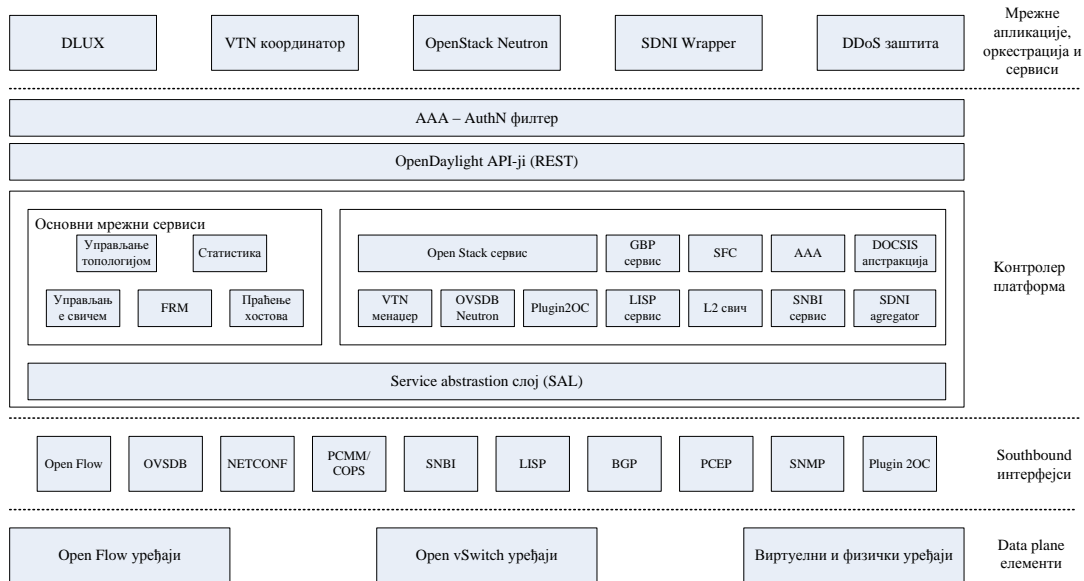
Поред могућности реализације функције *switching*-а на софтверском нивоу, постоје решења која омогућавају реализацију функције рутирања на софтверском нивоу. *Quagga* је софтверски пакет који омогућава реализацију рутирања (протоколи: OSPFv2, OSPFv3, RIP v1 и v2, RIPng и BGP-4) у *Unix* базираним окружењима [88]. Произвођач мрежне опреме *Mikrotik* је развио софтверски рутер заснован на *Linux* оперативном систему, који подржава протоколе рутирања (OSPF, RIP, BGP), MPLS, VPN, *firewall*, бежичну комуникацију итд. [89].

Када се говори о виртуелизацији мрежног окружења заснованог на *OpenFlow* протоколу, треба нагласити да постоји алат за виртуелизацију *FlowVisor*. *FlowVisor* представља *OpenFlow* алат који се понаша као прокси између *OpenFlow* свичева и више *OpenFlow* контролера, дели мрежно окружење и креира изоловане, програмабилне логичке мреже у оквиру исте топологије [90].

3.5.2. OpenDaylight

OpenDaylight је оквир отвореног кода за креирање софтверски дефинисаних мрежа настао као резултат сарадње више предузећа: *Brocade, Cisco Systems, IBM, HP, Juniper, Microsoft, Citrix* итд. Представља комбинацију контролера, интерфејса, апликација и *plug-in*-ова за различите протоколе. У *OpenDaylight* архитектури (Слика 11) се разликују следећи слојеви [75]:

- Мрежне апликације и оркестрација
- Контролер
- Мрежни уређаји



Слика 11: *OpenDaylight* архитектура [75]

Највиши слој чине мрежне апликације и он има задатак да контролише и прати понашање мреже. Поред тога на овом слоју се реализује и оркестрација, што је посебно важно у случају *cloud* сервиса. Слој контролера обезбеђује API интерфејсе за апликативни слој (*northbound interface*) и протоколе за управљање хардвером (*southbound interface*). Контролер је у потпуности реализован у софтверу, а преко различитих протокола (*OpenFlow 1.0, OpenFlow 1.3, BGP-LS* итд.) остварује везу са мрежним уређајима. Најнижи слој чине физички и

виртуелни мрежни уређаји (рутери, свичеви итд.), који имају задатак да обезбеде конективност у мрежи.

Важна карактеристика *OpenDaylight* пројекта је модуларна архитектура, која обезбеђује једноставну интеграцију са другим системима. Посебно је значајна могућност интеграције са *OpenStack*-ом, која се остварује преко *northbound ML2* (енгл. *Modular Layer 2*) *plug-in*-а [91].

3.5.3. Комерцијална решења

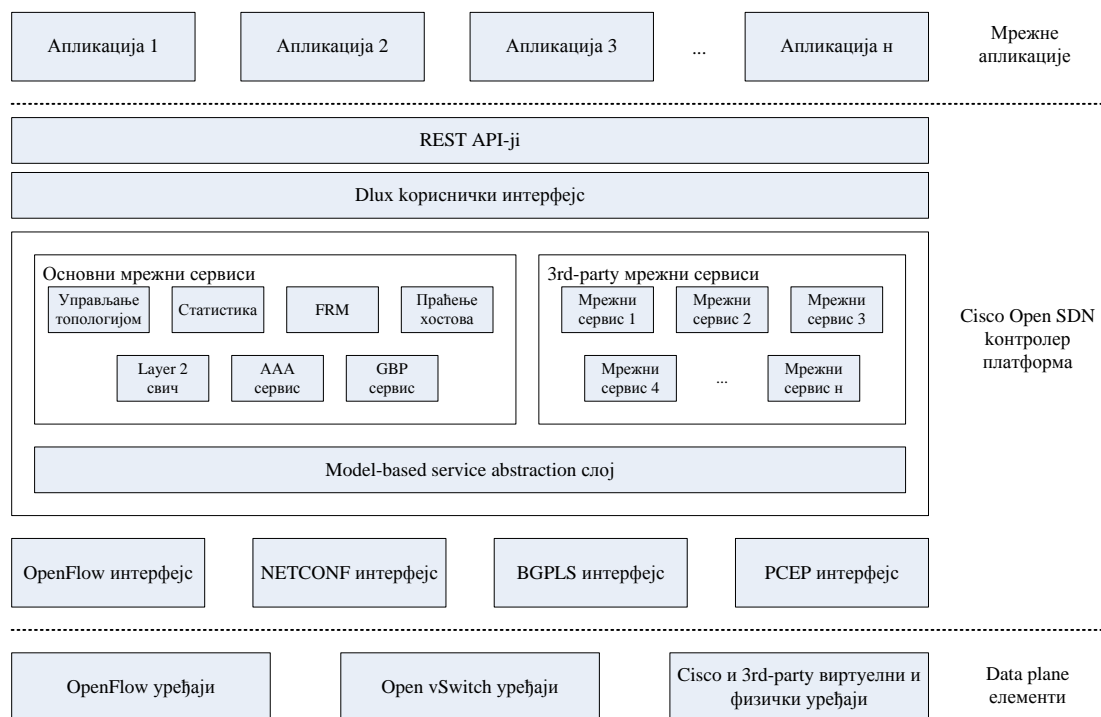
Многи произвођачи опреме су развили различита решења софтверски дефинисаних мрежа са жељом да омогуће виши степен флексибилности и програмабилности мрежне инфраструктуре.

Предузеће *Cisco Systems* је имплементирало SDN концепт у оквиру *Cisco Open Network Environment (ONE)* пројекта [92]. Основне компоненте овог решења су:

- Виртуелни свичеви
- Контролер
- Програмабилни API

Cisco Open SDN контролер представља комерцијалну дистрибуцију *OpenDaylight* пројекта, која уводи висок степен аутоматизације у мрежну инфраструктуру [93]. На слици 12 је приказана архитектура овог решења.

Примена програмабилних мрежа доводи до убрзања рада веб-апликација као резултат комбиновања SDN мрежа, функције виртуелизације мреже и софтверских технологија отвореног кода. Такође, долази до унапређења управљања мрежом кроз аутоматизацију, оркестрирање захтева, поједностављивање дизајна мреже и бржу испоруку сервиса. Важно је нагласити и интеграцију свих врста ресурса у приступном слоју мреже, као и бољу искоришћеност мрежне и опреме у *data* центрима.

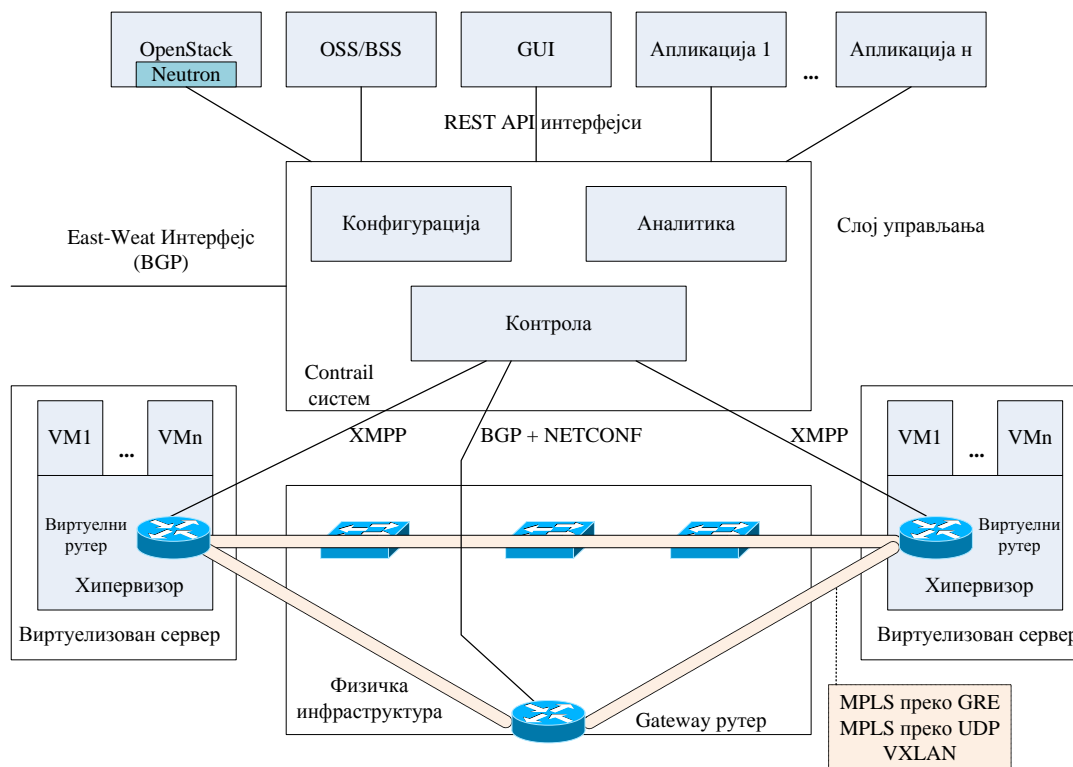


Слика 12: *Cisco Open SDN* платформа [93]

Предузеће *Juniper Networks* имплементирало је SDN концепт у оквиру *OpenContrail* решења [94]. Ово решење је доступно под Apache 2.0 лиценцом и обезбеђује компоненте за виртуелизацију мреже:

- *OpenContrail* контролер
- *OpenContrail* виртуелни рутер

На слици 13 је приказана архитектура *OpenContrail* решења. *OpenContrail* контролер је физички дистрибуиран SDN контролер задужен за управљање, контролу и анализу виртуелизованог мрежног окружења. *OpenContrail* виртуелни рутер је задужен за прослеђивање саобраћаја и реализован је у хипервизору виртуелизованих сервера чиме се креира виртуелна *overlay* мрежа. Начин рада *OpenContrail* виртуелног рутера је концептуално сличан постојећим виртуелним свичевима, али он поседује подршку за функцију рутирања саобраћаја и друге сервисе реализоване на вишим слојевима, па се зато и назива виртуелни рутер.



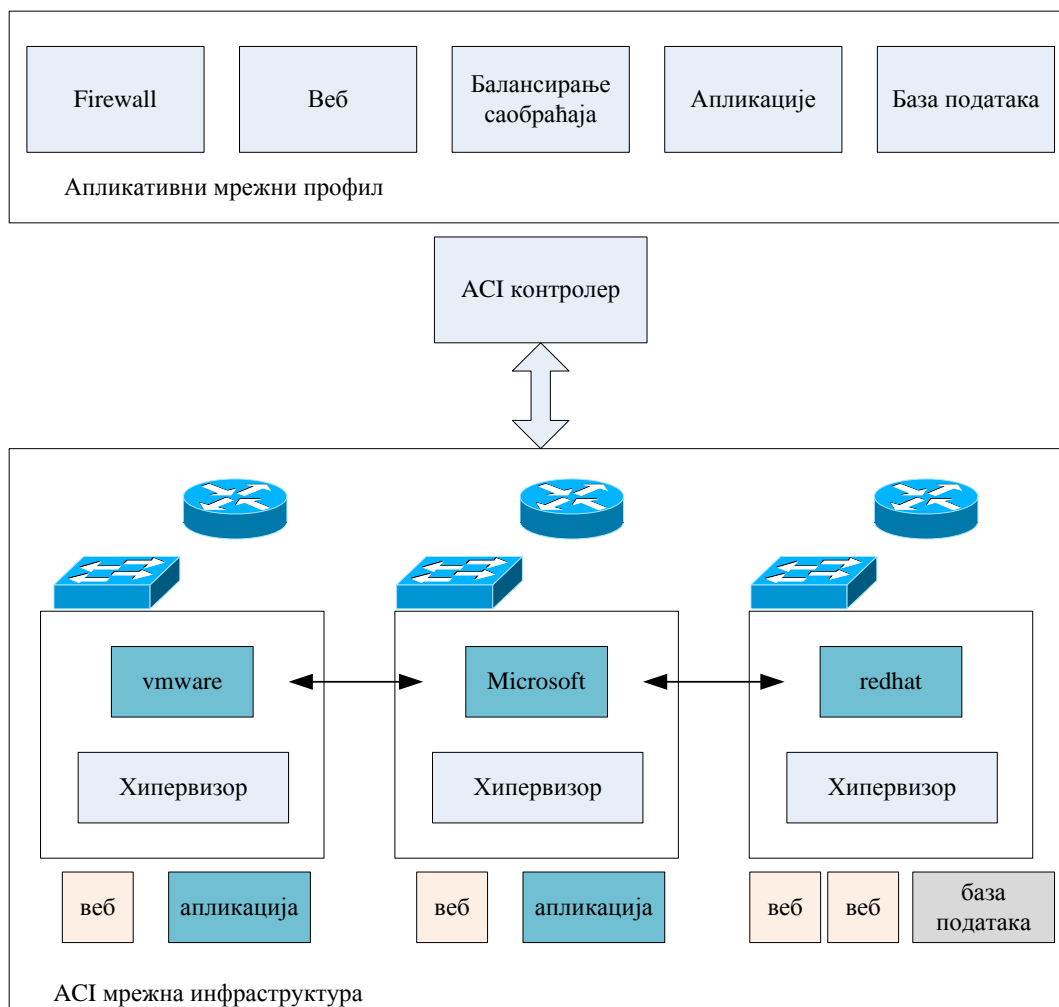
Слика 13: *OpenContrail* архитектура [94]

3.6. *Application centric* инфраструктура

Application centric инфраструктура (енг. *Application Centric Infrastructure – ACI*) представља модел мрежне инфраструктуре који је развила компанија *Cisco Systems* како би се процес примене апликација поједноставио, оптимизовао и убрзао [95]. Ова инфраструктура комбинује традиционалне мрежне елементе са централизованим управљањем, концептом програмабилности и аутоматизације. У питању је отворен екосистем који интегрише физичке и виртуелне мрежне елементе уз примену *policy-based* модела.

Application centric инфраструктуру чине следеће компоненте (Слика 14):

- *ACI* контролер (енг. *Application Policy Infrastructure Controller – APIC*)
- *ACI* мрежни уређаји (енг. *ACI Fabric*)
- Апликативни мрежни профили (енг. *Application Network Profile*)



Слика 14: АСИ архитектура [95]

АСИ контролер представља централну компоненту АСИ инфраструктуре, која је задужена за централизовано, програмабилно, аутоматизовано и једноставно управљање мрежним елементима. АСИ је потпуно одвојен од процеса прослеђивања података, што значи да и у случају губитка комуникације са АСИ контролером саобраћај може да се прослеђује. Као *northbound* интерфејс АСИ користи XML или JSON, а управљање мрежним елементима може да се реализује преко командне линије (CLI) или графичког интерфејса (GUI). За комуникацију са мрежним елементима користи *southbound* протокол OpFlex. OpFlex протокол омогућава размену апликативних полиса дефинисаних у XML или JSON формату између мрежног контролера и уређаја [96]. *Cisco Systems* у сарадњи са другим произвођачима опреме активно ради на стандардизацији овог протокола.

ACI мрежни уређаји могу бити физички уређаји или виртуелни ACI свичеви. Када су у питању физички уређаји реч је о *Cisco Nexus 9000* серији свичева, који могу да раде на класичан начин (*Cisco NX-OS* мод рада) или у *Cisco ACI* моду. Виртуелни свичеви могу бити реализовани у хипервизору и нису везани за одређеног произвођача опреме, а могу бити и AVS (енг. *Cisco Application Virtual Switch*) свичеви. AVS свичеви су посебно креирани за ACI инфраструктуру и засновани су на *Cisco Nexus 1000V* виртуелном свичу [97].

Апликативни мрежни профили представљају логичку репрезентацију свих компоненти апликације. У питању је логички скуп сервиса који захтевају сличну полису, њихове конекције и полисе које дефинишу наведене конекције. Дефинисање и конфигурација полиса је у надлежности ACI контролера.

Предности примене ACI инфраструктуре су вишеструке [98]:

- ACI инфраструктура представља дистрибуиран систем заснован на иновативним решењима који може да подржи трендове и динамичан развој апликација.
- ACI иновације су објављене као решења отвореног кода, а омогућена је и интеракција са системом.
- У питању је решење које комбинује опрему различитих произвођача опреме, што значи да решење није ограничено на групу произвођача опреме.
- ACI инфраструктура омогућава повезивање физичких и виртуелних уређаја, чиме се обезбеђује висок степен скалабилности у складу са захтевима.
- ACI инфраструктура представља једноставно решење које врши апстракцију управљања и тиме не зависи од имплементираних хардверских решења.
- Примена ACI инфраструктуре смањује капиталне и оперативне трошкове.

3.7. Безбедност у софтверски дефинисаним мрежама

Софтверски дефинисане мреже доносе значајна побољшања у домену умрежавања у погледу поједностављења процеса управљања мрежом и имплементације нових сервиса, али са друге стране питање безбедности постаје све значајније. Задатак мрежне инфраструктуре је да обезбеди ефикасан и безбедан пренос саобраћаја, а традиционална решења за заштиту мрежа више нису довољно ефикасна.

Применом централизованог управљања SDN контролер постаје *single-point-of-failure*. Зато је неопходно посебну пажњу посветити заштити контролера [99]. Друга критична тачка у SDN мрежама је *southbound* интерфејс преко кога се одвија комуникација између контролера и мрежних уређаја.

Основни безбедносни проблеми у софтверски дефинисаним мрежама су [57]:

- Фалсификовани или лажни токови саобраћаја који се могу искористити за напад на свичеве или контролер.
- Напади који користе рањивост свичева.
- Напади на комуникацију између свичева и контролера.
- Напади на контролере.
- Недостатак механизма како би се осигурало поверење између контролера и апликација за управљање.
- Напади на радне станице преко којих се остварује приступ SDN контролеру.
- Недостатак поузданих извора информација на основу којих би се утврдио узрок уоченог проблема и брз опоравак.

Open Networking Foundation дефинише следеће принципе, на којима треба да почивају софтверски дефинисане мреже, како би се обезбедио висок степен безбедности преноса саобраћаја [100]:

- Јасно дефинисати безбедносне зависности и границе поверења (енг. *Trust boundary*).

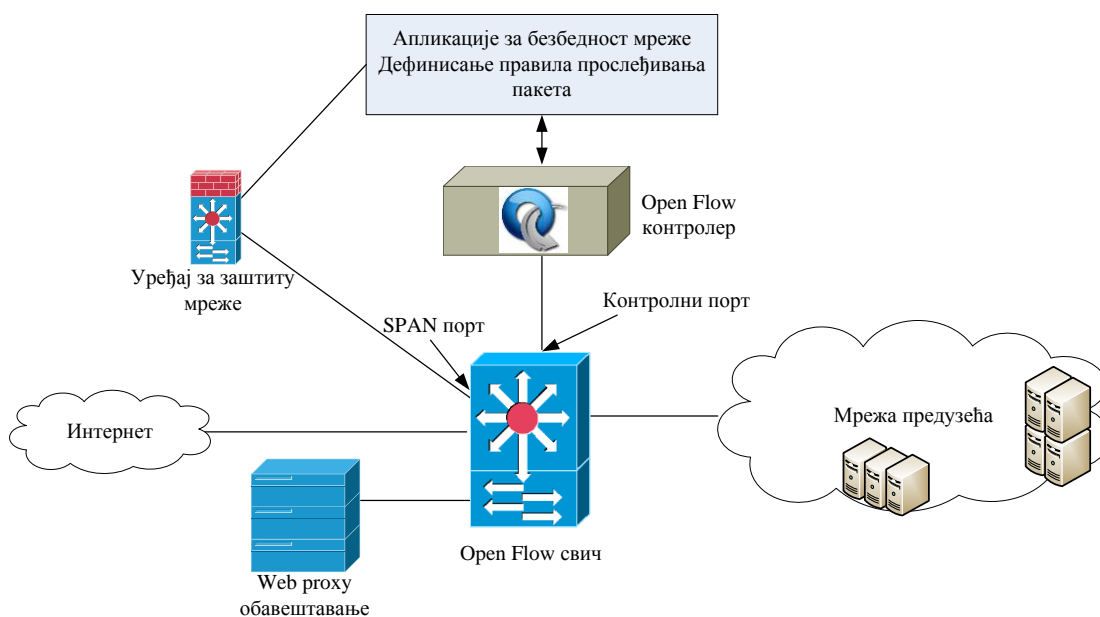
- Обезбедити јединствену идентификацију свих уређаја и корисника.
- Применити стандарде отвореног кода.
- Обезбедити поверљивост, интегритет и доступност безбедносних информација (енг. *confidentiality, integrity, availability* – CIA).
- Заштитити оперативне референтне податке (нпр. криптографски кључеви, кориснички креденцијали итд.).
- Дефинисати минималан ниво безбедности како би систем по *default*-у био довољно безбедан.
- Омогућити *accounting* и праћење активности.

Поред наведених принципа, приликом увођења безбедносне контроле потребно је обезбедити и следеће [100]:

- Безбедносне контроле треба да буду скалабилне и да подрже системе без обзира на њихову величину, а без повећања комплексности.
- Приликом увођења нових безбедносних контрола треба размотрити утицај на систем у целини. Нове безбедносне функције треба да уведу минималан степен комплексности.
- Безбедносне контроле треба да буду једноставне за имплементацију и одржавање.
- Треба омогућити компатибилност са претходним верзијама или обезбедити одговарајући начин надоградње система.
- Обезбедити адекватну документацију.
- Треба обезбедити могућност измене безбедносних креденцијала у сваком тренутку.
- Треба обезбедити аутоматизацију како би примена безбедносних контрола била ефикаснија и извршена на одговарајући начин.
- Могућност надгледања и решавања проблема за било који систем је од посебног значаја за успешно усвајање безбедносне контроле.

Пример примене SDN концепта у реализацији функције заштите је приказан на слици 15. У питању је реализација аутоматског карантина за *malware* (енг. *Automated Malware Quarantine* – AMQ). Задатак овог система је да детектује и

изолије небезбедне мрежне уређаје пре него што изврше негативан утицај на мрежу. Након тога аутоматски примењује неопходне *patch* програме како би се проблем решио и дозвољава поновно укључење уређаја у мрежу. У традиционалним мрежама оваква функционалност се примењује у виду нестандардизованих решења која су неефикасна, а посебно у случају великих мрежа са високом степеном виртуелизације и динамичких промена. Применом SDN концепта AMQ функционалност се реализује у оквиру SDN контролера [99], чиме се обезбеђује грануларно полисирање и ефикасна безбедносна заштита у мрежи.



Слика 15: Реализација аутоматског карантина за *malware* применом софтверски дефинисаних мрежа [99]

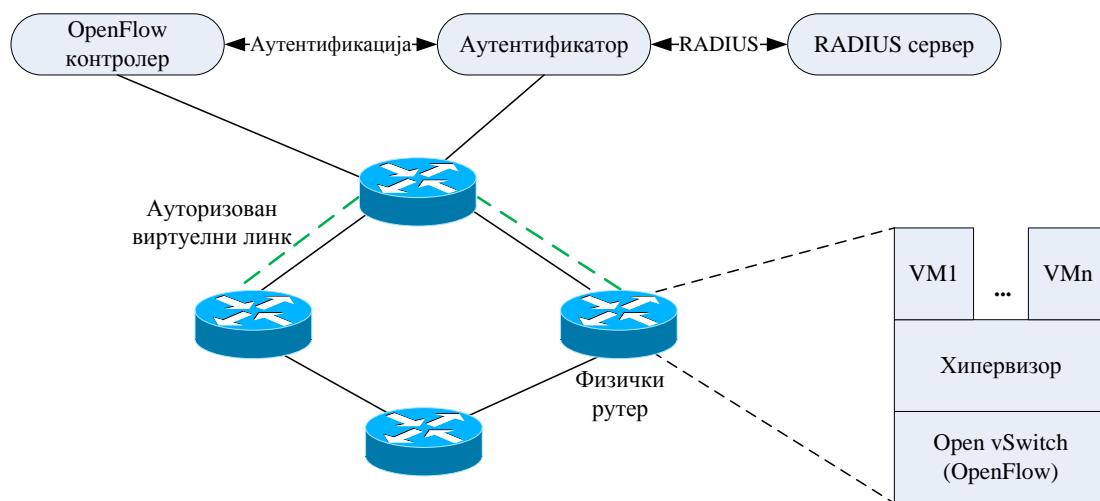
AuthFlow представља пример механизма који се може користити у софтверски дефинисаним мрежама у циљу заштите [101]. Овај механизам функционише по следећем принципу:

- Аутентификација хостова на *network access* слоју, чиме се уноси мали *overhead* и обезбеђује висок степен гранулације контроле приступа.

- Контрола приступа на нивоу хоста у складу са дефинисаним привилегијама, поређењем креденцијала за идентификацију и скупа токова додељених одређеном хосту.

На слици 16 је приказана архитектура *AuthFlow* решења, коју чине:

- *OpenFlow* контролер
- Аутентификатор
- RADIUS сервер



Слика 16: *AuthFlow* архитектура [101]

Провера идентитета виртуелне машине се врши применом EAP (енг. *Extensible Authentication Protocol*) и IEEE 802.1X стандарда. Аутентификатор врши деенкапсулацију EAP пакета и издваја поруке за аутентификацију, затим врши проверу виртуелног рутера преко RADIUS сервера и на крају обавештава *OpenFlow* контролер о резултату аутентификације. У случају успешне аутентификације дозвољава се приступ у складу са дефинисаним привилегијама, док се у супротном случају хосту ускрађује приступ.

3.8. Квалитет сервиса у софтверски дефинисаним мрежама

Данашње рачунарске мреже примењују конвергентну архитектуру која подразумева коришћење јединствене мрежне инфраструктуре за пренос различитих типова саобраћаја. Одређене категорије саобраћаја имају посебне захтеве, који морају бити испуњени како би комуникација у мрежи имала одговарајући квалитет. Зато је неопходно применити адекватне механизме који треба да обезбеде квалитет сервиса. Квалитет сервиса се може дефинисати као способност мреже да омогући бољи или посебан третман одређеној групи корисника, односно апликација [102] и представља низ техника за управљање мрежним ресурсима.

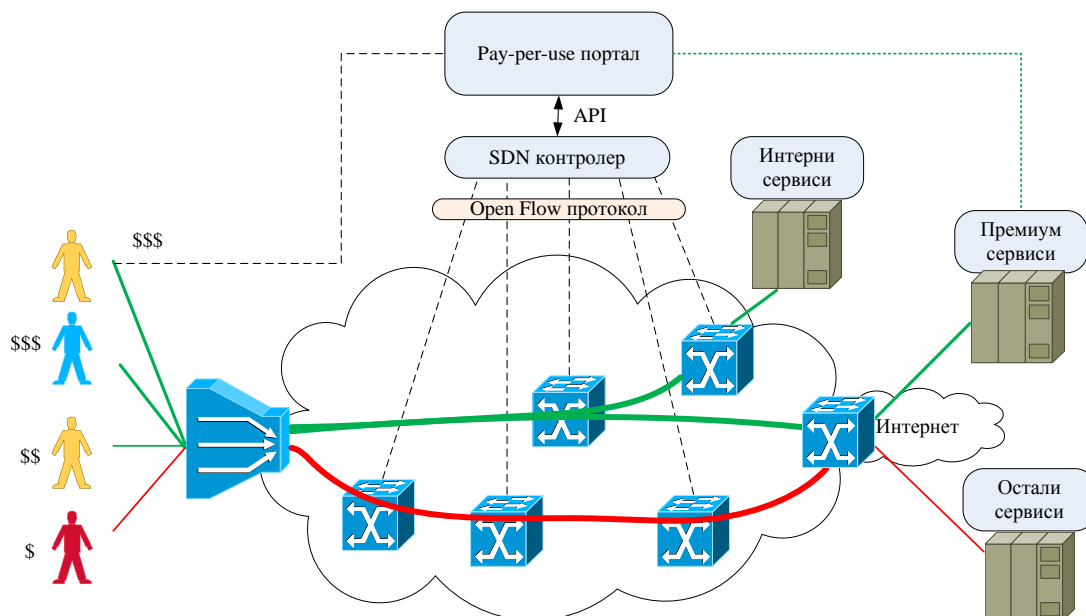
Постоји три модела мреже када је у питању квалитет сервиса:

- *Best Effort*
- *Integrated Services*
- *Differentiated Services*

Best Effort модел значи да квалитет сервиса није примењен у мрежи и да не постоји гаранција квалитета за одређене типове саобраћаја. Код *Integrated Services* модела се користи сигнализација приликом обавештавања мреже о потребним мрежним ресурсима. *Differentiated Services* модел се често примењује у мрежама, а процес примене квалитета сервиса у овом случају чине следећи кораци:

- Идентификација свих типова саобраћаја
- Подела саобраћаја на класе
- Примена одговарајућих QoS полиса за сваку класу саобраћаја

Примена концепта софтверски дефинисаних мрежа омогућава виши степен диверсификације саобраћаја и дефинисање QoS полиса на нивоу токова саобраћаја у односу на традиционалне мреже [103]. Слика 17 приказује реализацију квалитета сервиса у SDN окружењу.



Слика 17: QoS сервис у SDN окружењу [103]

У овом случају је могуће применити квалитет сервиса на нивоу тока података, па је неопходно прво идентификовати токове података како би се затим примениле одговарајуће полисе, односно доделили одређени мрежни ресурси. Процес доделе мрежних ресурса је у надлежности SDN контролера и врши се на основу претходно дефинисаних полиса.

Процес идентификације токова података има посебан значај и може се извршити на неки од следећих начина [103]:

- Према извору саобраћаја (нпр. изворишна IP адреса)
- Према одредишту саобраћаја (нпр. одредишна IP адреса)
- Према информацијама са четвртог до седмог слоја OSI модела.

Отворени програмабилни интерфејс за комуникацију са SDN контролером омогућава дефинисање различитих начина за разликовање токова података, као и различитих алгоритама за дефинисање полиса од којих зависи како ће одређени ток података бити третиран у мрежи.

3.9. Миграција на SDN архитектуру

Питање миграције са традиционалне на SDN архитектуру је комплексно пошто не постоји једнозначно решење. Први корак је дефинисање захтева које нова SDN архитектура треба да испуни, а затим у складу са тим избор одговарајућег начина миграције.

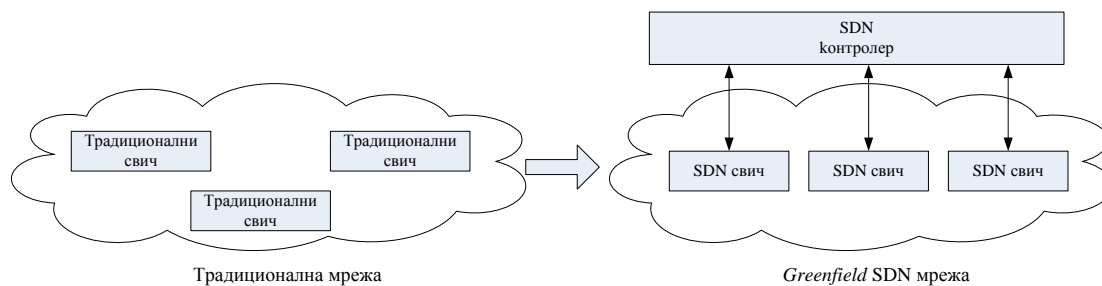
Кључни кораци приликом миграције на SDN архитектуру су [104]:

- Идентификација и дефинисање приоритета захтева нове мреже – Први корак представља идентификацију захтева нове мрежне инфраструктуре и утврђивање приоритетних захтева пошто у почетној фази није могуће реализовати све функционалности традиционалне мрежне инфраструктуре.
- Припрема традиционалне мреже за миграцију на SDN архитектуру – У зависности од изабраног начина миграције на SDN архитектуру постоји потреба да се традиционална мрежа припреми за миграцију.
- Имплементација миграције по фазама – У зависности од имплементираних уређаја у мрежи процес миграције на SDN архитектуру може бити специфичан, и зато се миграција изводи по фазама.
- Евалуација нове SDN архитектуре – Након миграције на SDN архитектуру потребно је проверити мрежне функционалности и тестирати ново мрежно окружење како би се потврдио исправан рад.

Постоје три основна начина миграције са традиционалне на SDN архитектуру [104]:

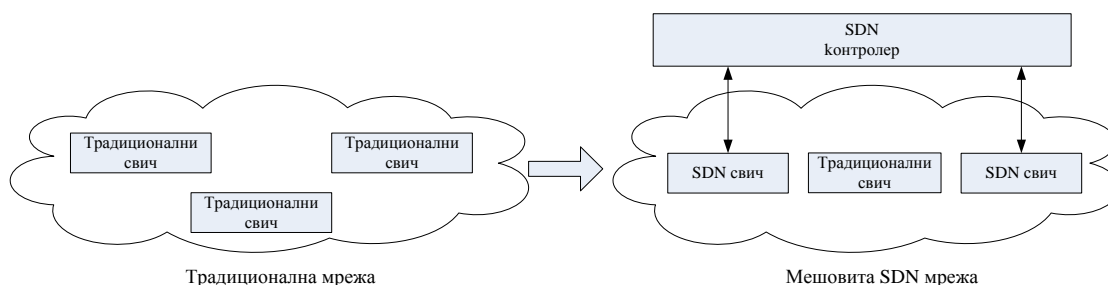
- Потпуна (енг. *Greenfield*)
- Мешовита (енг. *Mixed*)
- Хибридна

Једноставан начин миграције је потпуна миграција пошто не захтева интеграцију или интероперабилност са постојећом традиционалном инфраструктуром. Приказ потпуне миграције је дат на слици 18.



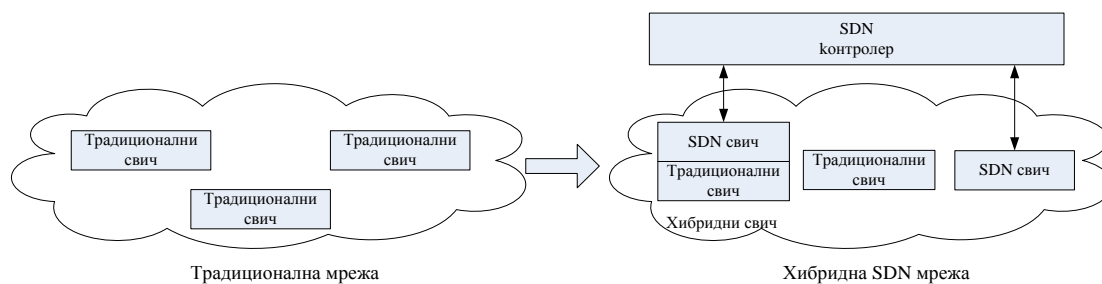
Слика 18: *Greenfield* миграција

Код мешовите миграције у мрежи постоје и традиционални мрежни уређаји и нови SDN свичеви (Слика 19). У овом случају SDN контролер размеђује информације са традиционалним мрежним уређајима преко традиционалног *control plane*-а.



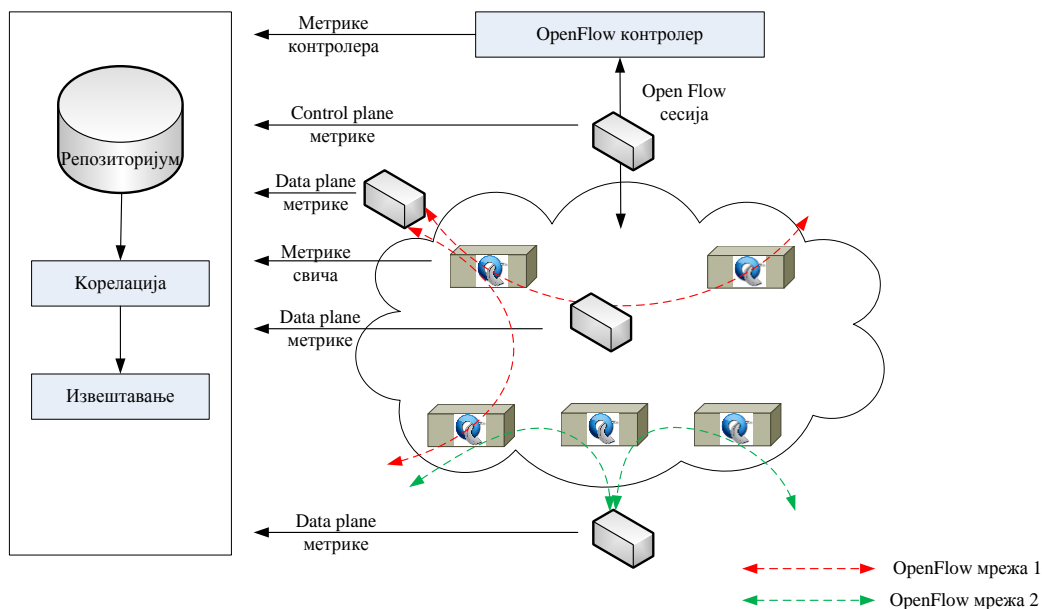
Слика 19: Мешовита миграција

Хибридна миграција подразумева имплементацију хибридних уређаја који комуницирају и на традиционалан начин, али и са SDN контролером (Слика 20).



Слика 20: Хибридна миграција

Да би се процес миграције извршио на ефикасан начин, неопходно је пре миграције, за време, као и након миграције пратити одређене мрежне параметре (метрике) [105]. Слика 21 приказује процес прикупљања вредности дефинисаних параметара са SDN контролера и SDN свичева.



Слика 21: Процес прикупљања метрика [105]

3.10. Емулација софтверски дефинисаних мрежа

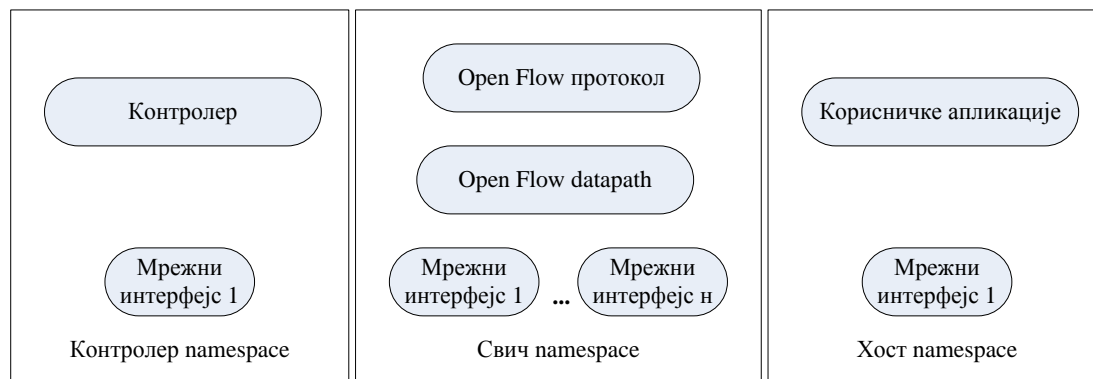
Mininet је мрежни емулатор за креирање виртуелног мрежног окружења са реалним компонентама који се покреће на једном рачунару, а све у циљу брзог и једноставног креирања прототипова различитих рачунарских мрежа [106].

Комбиновањем принципа виртуелизације, CLI и API интерфејса, *Mininet* омогућава креирање, интеракцију и прилагођавање SDN мреже. *Mininet* подржава креирање следећих мрежних елемената:

- Хостова
- Свичева
- Контролера
- Линкова

На *Mininet* хостовима је имплементиран *Linux*, док свичеви подржавају *OpenFlow* и SDN архитектуру [107].

На слици 22 је приказана *Mininet* архитектура.



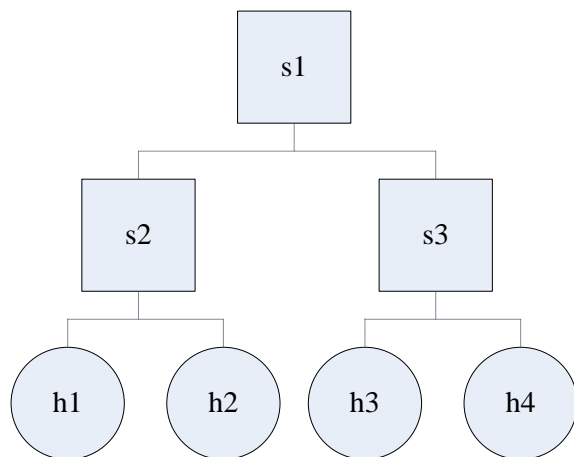
Слика 22: *Mininet* архитектура

Основне карактеристике *Mininet* алата су:

- Једноставно мрежно окружење за развој *OpenFlow* апликација.
- Подржава тестирање комплексних мрежних топологија без потребе за физичком реализацијом истих.
- Постојање предефинисаних топологија мрежа.
- Могућност креирања комплексних мрежа.
- CLI интерфејс.
- *Python* API.

У поређењу са другим мрежним емулаторима и симулаторима, *Mininet* се инсталира једноставно (постоји предефинисана виртуелна машина), покреће се знатно брже, представља скалабилније решење и има могућност повезивања са реалним рачунарским мрежама.

На слици 23 је приказан пример мрежне топологије креиране у *Mininet*-у, као и одговарајућа *Python* скрипта.



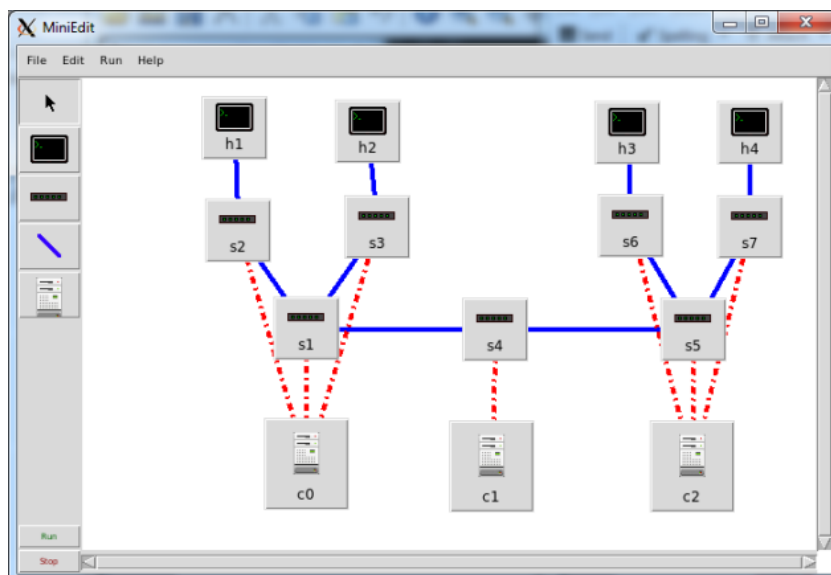
```

from mininet.net import
Mininet
from mininet.topolib import
TreeTopo
Tree22 = TreeTopo(depth=2,
fanout=2)
net = Mininet(topo=Tree22)
net.start()
net.pingAll()
net.stop()

```

Слика 23: Пример мрежне топологије креиране у *Mininet*-у

Поред могућности креирања одговарајућих мрежних топологија помоћу скрипти постоји и графички едитор *Miniedit* [108] (Слика 24).



Слика 24: *Miniedit*

Иако *Mininet* представља добар алат за тестирање мрежа, има одређена ограничења. Не може се применити у случају *OpenFlow* свичева или апликација које нису *Linux* компатибилне. Такође, на линковима није могуће дефинисати

одређену брзину. Међутим, велики недостатак представља немогућност гаранције перформанси, посебно у случају великог оптерећења. Због мултиплексирања оптерећења процесора не постоји гаранција да ће виртуелни хост, који има пакет за слање, исти одмах и проследити, као ни да ће сви виртуелни свичеви прослеђивати саобраћај истом брзином [109].

3.11. Примена софтверски дефинисаних мрежа

Принципи централизације, апстракције и програмабилности представљају основне карактеристике SDN мрежа, па се овај концепт може применити у различитим мрежним окружењима.

3.11.1. Примена софтверски дефинисаних мрежа у предузећима

Enterprise окружење има високе захтеве у погледу аутоматизације пословних процеса и виртуелизације, уз смањење оперативних трошкова, што SDN концепт може да омогући [110]. У *data* центрима постоји велики број сервера и виртуелних машина, па је питање скалабилности и ефикасности од посебног значаја. Хетерогеност ових система утиче и на комплексност управљања, повећава трошкове реализације и одржавања мреже, а може имати за последицу и проблеме са скалабилношћу и доступношћу у мрежи. Примена SDN архитектуре у мрежама предузећа, поред флексибилности, доноси и следеће предности [50]:

- Смањење времена креирања нових сервиса
- Повећање доступности сервиса
- Изолацију саобраћаја у логичким мрежама
- Оптимално коришћење ресурса
- Једноставно управљање мрежом.

Посебан проблем представља и енергетска ефикасност, па примена одговарајуће енергетски ефикасне SDN архитектуре може у знатној мери да смањи потрошњу

електричне енергије, а самим тим и трошкове [111]. Поред примене *OpenFlow* протокола за комуникацију између слоја управљања и прослеђивања података, SDN архитектура се може комбиновати и са стандардним технологијама, као што су VLAN-ови и MPLS (енг. *Multiprotocol Label Switching*) [112]. Примену у предузећима има и VXLAN технологија, која у односу на традиционалан VLAN концепт уводи висок степен флексибилности и скалабилности кроз подршку већег броја мрежних сегмената [87].

Значајна примена софтверски дефинисаних мрежа у *enterprise* окружењу је и у случају спајања мрежних инфраструктура два предузећа. Процес интеграције је поједностављен захваљујући централизацији управљања и примени једноставних мрежних уређаја и протокола отвореног кода. На овај начин се утиче и на смањење трошкова интеграције два предузећа.

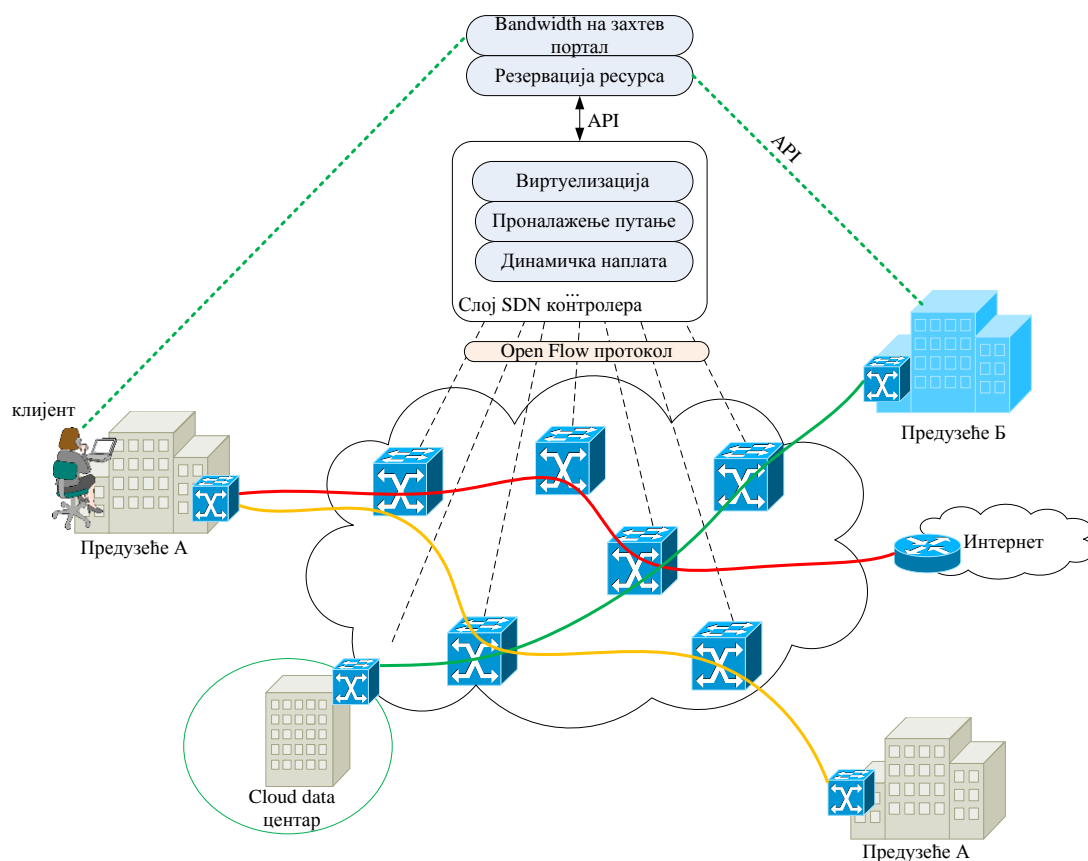
Cloud computing је распрострањена мрежна технологија, која у комбинацији са SDN концептом има позитиван утицај на функционисање мрежа [113]. Повећање скалабилности, флексибилности и смањење трошкова имплементације и одржавања мреже су предности ове интеграције [114]. Интеграција *cloud* и SDN концепта има значајан допринос у LAN (енг. *Local Area Network*) окружењу, али се може применити и у WAN (енг. *Wide Area Network*) окружењу [115].

3.11.2. Примена софтверски дефинисаних мрежа у провајдерским окружењима

Захтеви корисника су утицали на повећање комплексности сервиса које сервис провајдери треба да обезбеде, што је утицало да се и у овој области решење потражи у виду софтверски дефинисаних мрежа. Како би се процес имплементације софтверски дефинисаних мрежа у мрежама провајдера учинио ефикаснијим, велики број провајдера и произвођача опреме се удружио са циљем да се креира одговарајуће решење за провајдерске мреже. Пример таквог решења је SDN оперативни систем за сервис провајдере ONOS (енгл. *Open Network*

Operating System) [116]. Захваљујући скалабилности, високом степену доступности, могућношћу апстракције и добрим перформансама, ONOS омогућава једноставно креирање апликација и сервиса.

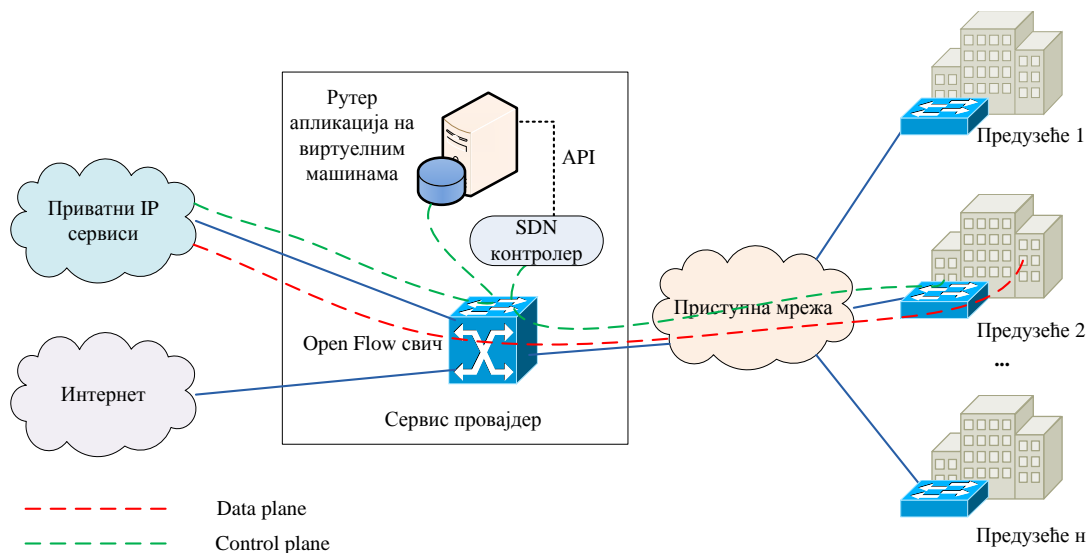
На слици 25 је приказан пример примене SDN концепта у оквиру реализације сервиса *bandwidth* на захтев (енг. *Bandwidth on Demand*). У овом случају се потребан *bandwidth* корисницима обезбеђује као NaaS.



Слика 25: Реализација сервиса *bandwidth* на захтев у SDN окружењу [103]

Известан број сервиса крајњих корисника се реализује као *managed*, односно постављањем одређене опреме на локацији корисника чија је конфигурација и управљање у надлежности сервис провајдера. Примена концепта софтверски дефинисаних мрежа у овом случају може знатно допринети поједностављењу реализације сервиса имплементацијом једноставних *Layer 2* мрежних уређаја на

локацији корисника. На слици 26 је приказана реализација *managed* сервиса код кога је комплексна функција рутирања реализована у виду апликације на виртуелним машинама. Овакав начин реализације сервиса повећава скалабилност и смањује трошкове реализације сервиса [103].



Слика 26: Реализација *managed* сервиса у SDN окружењу [103]

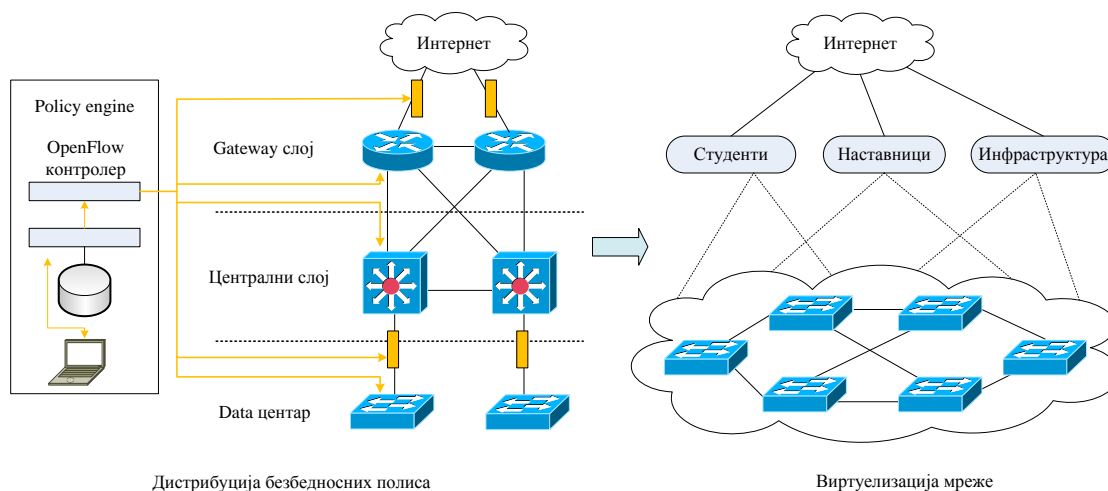
Поред стандардних захтева у погледу скалабилности, ефикасности и безбедности [117], сервис провајдери су суочени и са захтевом да корисницима пруже сервисе који испуњавају одређене SLA параметре [44]. Како би ови услови били испуњени, неопходно је минимизовати време опоравка у случају инцидентне ситуације у мрежи, као и омогућити да преусмерење саобраћаја нема утицаја на сервисе крајњих корисника, што се остварује применом техника за *traffic engineering* [118].

3.11.3. Примена софтверски дефинисаних мрежа у академским кампус окружењима

Примена софтверски дефинисаних мрежа у академском кампус окружењу треба да омогући подршку за постојеће образовне сервисе, али и једноставну примену

будућих сервиса и нових мрежних технологија [119]. Да би се ови захтеви реализовали у пракси, неопходно је креирати адекватну ИТ инфраструктуру, која ће пре свега имати висок степен скалабилности и програмабилности [120]. Такође, потребно је обезбедити подршку за примену различитих образовних модула који треба да се прилагоде захтевима и потребама студената [121]. Ови захтеви су од изузетног значаја када је у питању образовање студената у области рачунарства и информационо-комуникационих технологија.

Велики број високошколских установа поседује *cloud* инфраструктуру у оквиру које се може применити SDN концепт [122], [123] како би се побољшала ефикасност образовног процеса. На овај начин је могуће раздвајање продукционих и истраживачких података [124], што је посебно важно када су у питању истраживања и развој нових протокола, мрежних модела и сл.



Слика 27: SDN инфраструктура у академском окружењу [50]

Постојање инфраструктуре која може да подржи *multi-tenant* принцип је значајно у академском окружењу, где јединствена мрежна инфраструктура има задатак да подржи комуникацију између великог броја различитих корисника, који имају различите нивое привилегија и права за приступ одређеним сервисима и ресурсима (Слика 27). Зато је важно да инфраструктура омогући изолацију саобраћаја између корисника и неометан рад већег броја логичких мрежа. SDN

инфраструктура може да обезбеди ове услове, као и једноставан и брз начин за креирање логичких мрежа на захтев [50].

Коришћењем SDN архитектуре се може на једноставан начин реализовати и платформа за учење концепта софтверски дефинисаних мрежа. Да би се у потпуности савладала ова материја, пре свега је потребно познавање функција апликативног слоја, док се инфраструктурном слоју не мора посветити велика пажња. Овај приступ је могуће реализовати применом виртуелизованог мрежног окружења, односно применом *Mininet* алата, који има задатак да емулира потребно мрежно окружење на инфраструктурном нивоу [125].

4. МОДЕЛ SDN ИНФРАСТРУКТУРЕ ЗА Е-ОБРАЗОВАЊЕ

4.1. Развој модела инфраструктуре

ИТ инфраструктура представља основу система за е-образовање. Има задатак да обезбеди подршку за функционисање великог броја различитих сервиса за наставнике, студенте, истраживаче и административно особље. Зато је потребно процесу развоја модела инфраструктуре академске мреже посветити посебну пажњу. Пре свега, неопходно је дефинисати принципе, на којима инфраструктура образовне институције мора да почива [126]:

- Подршка за образовне сервисе коришћењем заједничке инфраструктуре.
- Подршка сервисима за пренос говора, видео-садржаја и података уз гаранцију квалитета сервиса.
- Могућност прилагођавања новим техничким иновацијама и примене нових мрежних сервиса.
- Интеграција нових сервиса и технологија уз задржавање постојеће мрежне опреме, протокола и начина комуникације.

Како би се реализовала оваква ИТ инфраструктура у академској мрежи, потребно је применити различите мрежне уређаје и технологије, при чему следећи захтеви морају бити испуњени:

- Дизајн заснован на хијерархијском моделу
- Висок степен доступности
- Флексибилност
- Примена отворених стандарда
- Висок степен безбедности

У табели 1 је дат приказ основних захтева које инфраструктура за е-образовање мора да испуни, уз њихов опис.

Табела 1: Карактеристике инфраструктуре за е-образовање

Карактеристика	Опис
Хијерархија	Подела мреже на слојеве уз дефинисање функција сваког слоја, што поједностављује реализацију и одржавање мреже
Доступност	Константан приступ мрежним уређајима и сервисима
Флексибилност	Могућност динамичког прилагођавања мреже према новим захтевима
Отворени стандарди	Примена <i>open-source</i> решења са циљем постизања интероперабилности
Безбедност	Заштита мрежних ресурса и сервиса од злоупотреба

4.2. Анализа постојећих модела инфраструктуре

Промене у процесу едукације, као и примена нових информационо-комуникационих технологија, постављају нове захтеве које један едукативни систем мора да испуни [3]. Ово је посебно значајно када је у питању едукација у области рачунарских наука и информационих технологија [127]. За учење нових технологија неопходна је одговарајућа мрежна инфраструктура која треба да омогући висок степен доступности, ефикасности и безбедности. Како би се обезбедила подршка за нове ИТ курсеве и студентима пружила могућност стицања искуства у реалном мрежном окружењу, неопходно је изменити традиционалну ИТ инфраструктуру и применити различите технике виртуелизације и емулације мрежног окружења [128]. Овакав приступ даје добре резултате у процесу учења комплексних тема у области рачунарских мрежа [129].

У данашњим академским мрежама примена апликација за реализацију курсева из информационо-комуникационих технологија захтева велика улагања и различите хардверске и софтверске ресурсе [6]. Главни изазов је креирати комплексан систем за е-образовање који може да користи различите технологије и сервисе [5]. Поред потребе за интеграцијом нових технологија, повећање броја образовних сервиса, као и учесника у процесу е-образовања имају велики утицај на даљи развој архитектуре академске мреже. Системи за е-образовање су углавном засновани на традиционалној мрежној архитектури, коју карактеришу висока комплексност и

дуго време реализације нових сервиса [7]. Ови недостаци представљају разлог за увођење вишег степена скалабилности, програмабилности и динамичке архитектуре у академске мреже.

Како би испуниле наведене захтеве, образовне институције често примењују *cloud computing* технологију [124], [8]. На овај начин се остварује висок степен скалабилности и омогућава наставницима да дефинишу нове сервисе за е-учење и интегришу их у курсеве који се односе на нове технологије, као што је SDN [130]. Додатно, овакав приступ омогућава и истраживачима и студентима да креирају и користе напредне мрежне апликације и сервисе [11], [131].

Развој нових технологија поставља нове захтеве пред образовне институције и захтеви, које академске мреже морају да испуне, постају комплекснији. Из тог разлога се намеће потреба за вишим степеном програмабилности у академском окружењу, што се може постићи применом концепта софтверски дефинисаних мрежа. Апстракција, као основна карактеристика SDN технологије, заједно са динамичком доделом мрежних ресурса чине SDN адекватним решењем које може да одговори на постављене захтеве. Могућност креирања виртуелног мрежног окружења на једноставан начин представља добру основу за реализацију и тестирање различитих сервиса и апликација, а све у циљу подршке образовног процеса. Крајњи циљ је поједноставити процес креирања и увођења нових предмета, који се баве изучавањем нових технологија, као што су интернет интелигентних уређаја, *big data*, SDN итд.

Мрежна инфраструктура, која је заснована на концептима *cloud computing*-а и SDN-а, представља добру основу за развој модерних система е-образовања, која може да испуни наведене захтеве. Идеја ове дисертације је развој модела инфраструктуре за е-образовање, који је заснован на концепту софтверски дефинисаних мрежа. Циљ модела је да се корисницима омогући да на ефикасан и једноставан начин користе постојеће образовне сервисе, али и да се обезбеди подршка за развој нових сервиса и њихову интеграцију у систем е-образовања.

4.3. Структура предложеног модела инфраструктуре

Модел инфраструктуре за е-образовање заснован на софтверски дефинисаним мрежама треба да омогући флексибилно и ефикасно образовно окружење, као и интеграцију постојећих инфраструктурних и образовних сервиса. Примарни задатак је обезбедити инфраструктуру која омогућава једноставну и брзу имплементацију нових наставних садржаја, пре свега у области рачунарства, информационих технологија, мобилног пословања итд. Инфраструктура за е-образовање треба да подржи интеграцију са системом за управљање учењем.

Предлог модела инфраструктуре за е-образовање заснован на софтверски дефинисаним мрежама обухвата следеће компоненте:

- K1: Архитектура физичке инфраструктуре
 - Пасивна опрема
 - Рутери
 - Свичеви
 - Сервери

- K2: Архитектура *cloud* инфраструктуре
 - Инфраструктура као сервис
 - Платформа као сервис
 - Софтвер као сервис
 - Виртуелизација

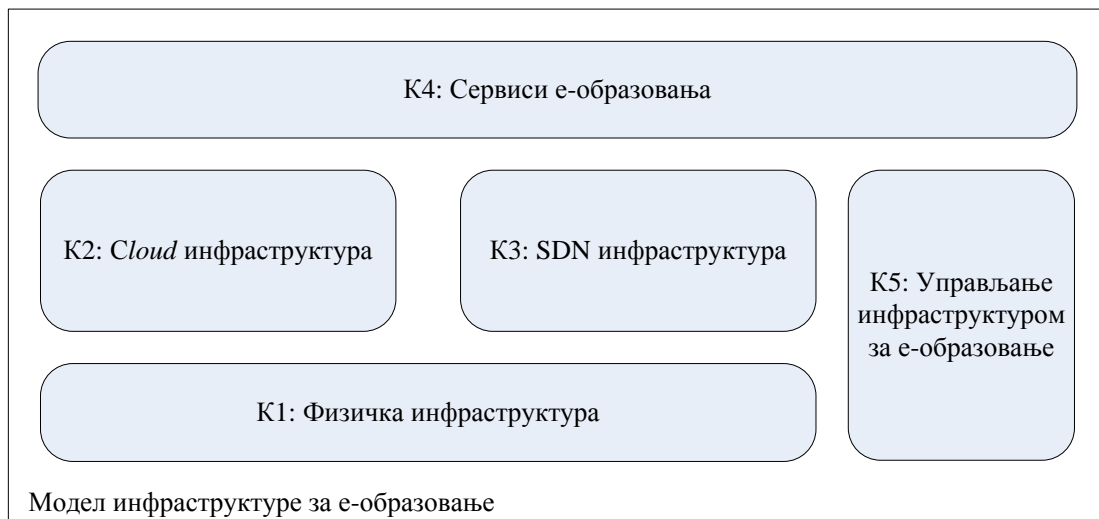
- K3: Архитектура SDN инфраструктуре
 - Виртуелни свичеви
 - SDN контролери
 - Мрежа као сервис (енг. *Network as a Service* – NaaS)

- K4: Сервиси е-образовања
 - Инфраструктурни сервиси

- Образовни сервиси
 - Сервиси за управљање учењем
 - Сервиси за комуникацију
 - Сервиси за подршку настави
 - Јединствени веб-портал

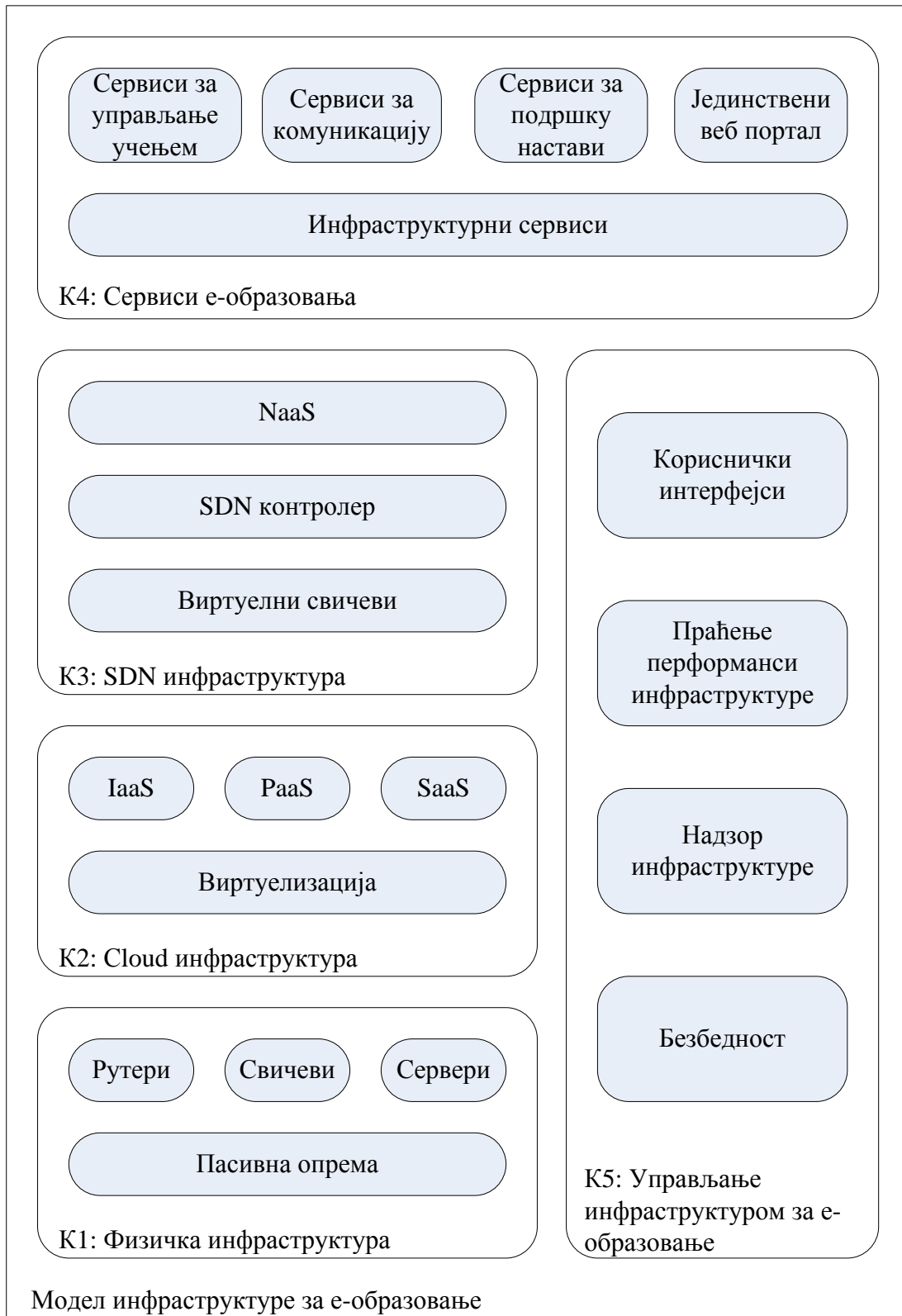
- K5: Управљање инфраструктуром за е-образовање
 - Кориснички интерфејси
 - Праћење перформанси инфраструктуре
 - Надзор инфраструктуре
 - Безбедност

На Слици 28 приказане су компоненте модела инфраструктуре за е-образовање.



Слика 28: Компоненте модела инфраструктуре за е-образовање

Слика 29 приказује детаљну структуру наведених компоненти, као и њихове међусобне везе.



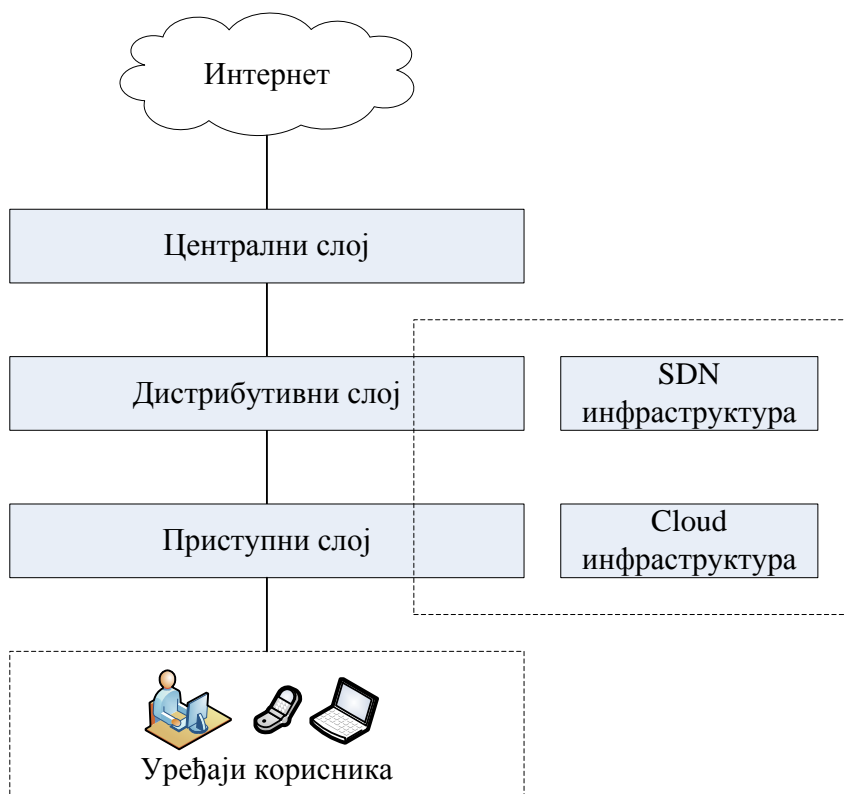
Слика 29: Детаљна структура модела

4.4. Приказ главних компоненти структуре предложеног модела

4.4.1. Логички модел архитектуре

Основу архитектуре представља хијерархијски мрежни модел који дефинише: приступни, дистрибутивни и централни слој. Основне карактеристике овог модела су модуларност и редунданса и захваљујући њима хијерархијски модел пружа могућност једноставне реализације и одржавања у традиционалним мрежама. Међутим, традиционалне мреже не пружају могућност динамичке алокације ресурса, па је зато неопходно увести измене. Како би се обезбедило динамичко прилагођавање мреже у складу са новим захтевима, потребно је имплементирати виртуелизацију и применити концепт програмабилности. Зато је на приступном и дистрибутивном слоју реализована *cloud* и SDN инфраструктура.

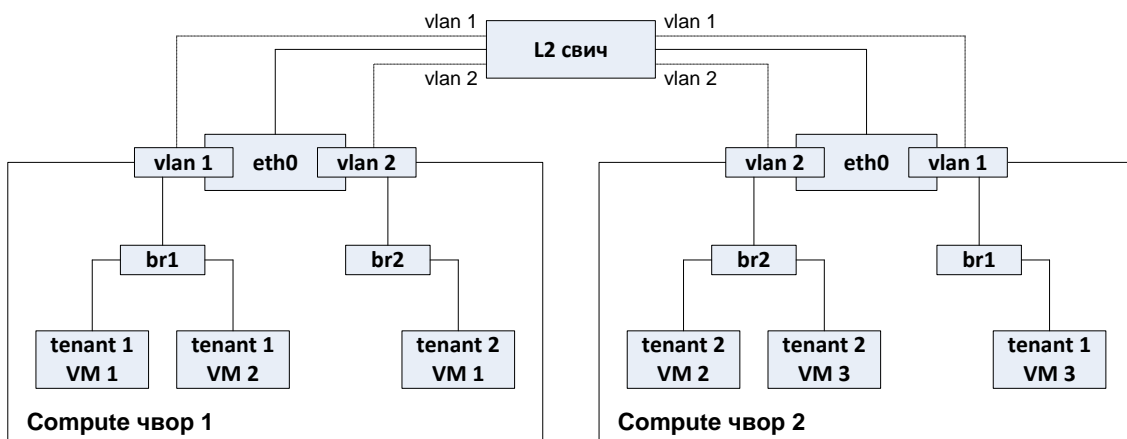
На слици 30 је приказан логички модел архитектуре за е-образовање.



Слика 30: Логички модел архитектуре за е-образовање

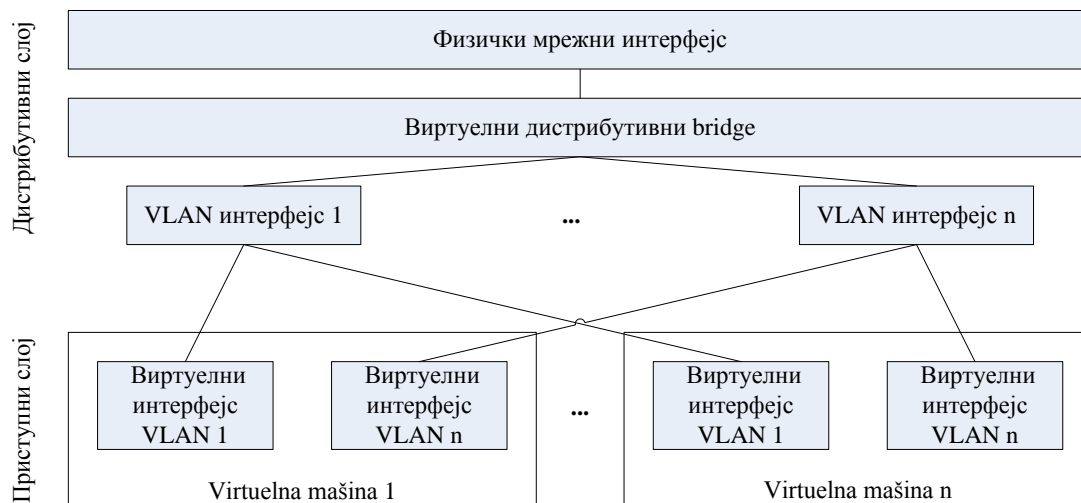
4.4.2. Модел *cloud* инфраструктуре

Cloud computing технологија је нашла примену у многим академским окружењима. Захваљујући концепту виртуелизације, који се налази у основи ове технологије, образовне институције могу истовремено да користе јединствену инфраструктуру за различите потребе. Управо овај *multi-tenancy* концепт омогућава да различити корисници креирају потпуне *end-to-end* мрежне топологије према својим захтевима, а уз потпуну изолацију од осталих корисника (Слика 31). Ово је од велике важности у академским окружењима где постоји потреба да велики број наставника и студената користи исту инфраструктуру за различите задатке и пројекте [122].



Слика 31: *Multi-tenancy* концепт [122]

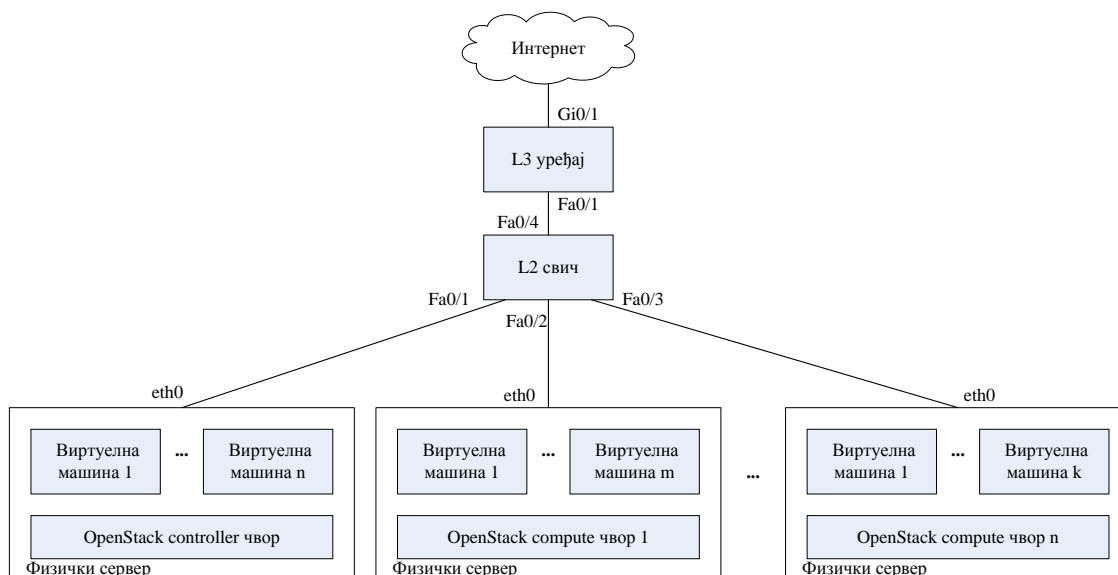
На слици 32 је приказан концептуални модел *cloud* инфраструктуре. Како би се обезбедило раздвајање саобраћаја различитих корисника, примењени су VLAN-ови. На овај начин је извршена сегментација мреже, изолација саобраћаја и обезбеђена флексибилност [132] потребна за имплементацију различитих образовних сервиса.



Слика 32: Концептуални модел *cloud* инфраструктуре [37]

Како би се обезбедило једноставно управљање *cloud* инфраструктуром, потребно је изабрати одговарајући систем за управљање. Пошто је један од почетних захтева приликом развоја модела инфраструктуре за е-образовање примена решења отвореног стандарда, одлучено је да се примени *OpenStack*. У питању је IaaS алат, који карактерише висока скалабилност, *multi-tenancy* концепт и *plug-in* архитектура [133], што га чини добрим избором за академско окружење. Од посебне је важности споменути *OpenStack Networking* (Neutron) компоненту [134], која даје могућност реализације мреже као сервиса (енгл. *Network as a Service* - NaaS).

На слици 33 је приказан пример физичке реализације *cloud* инфраструктуре применом *OpenStack* решења.

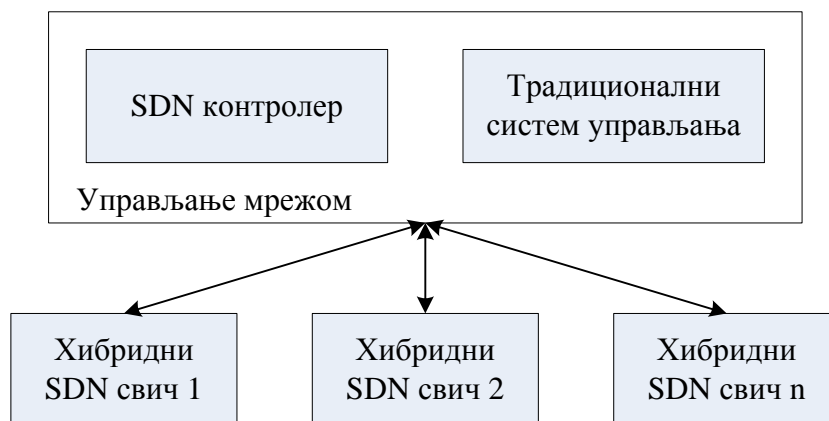


Слика 33: Пример физичке реализације *cloud* инфраструктуре

4.4.3. Модел SDN инфраструктуре

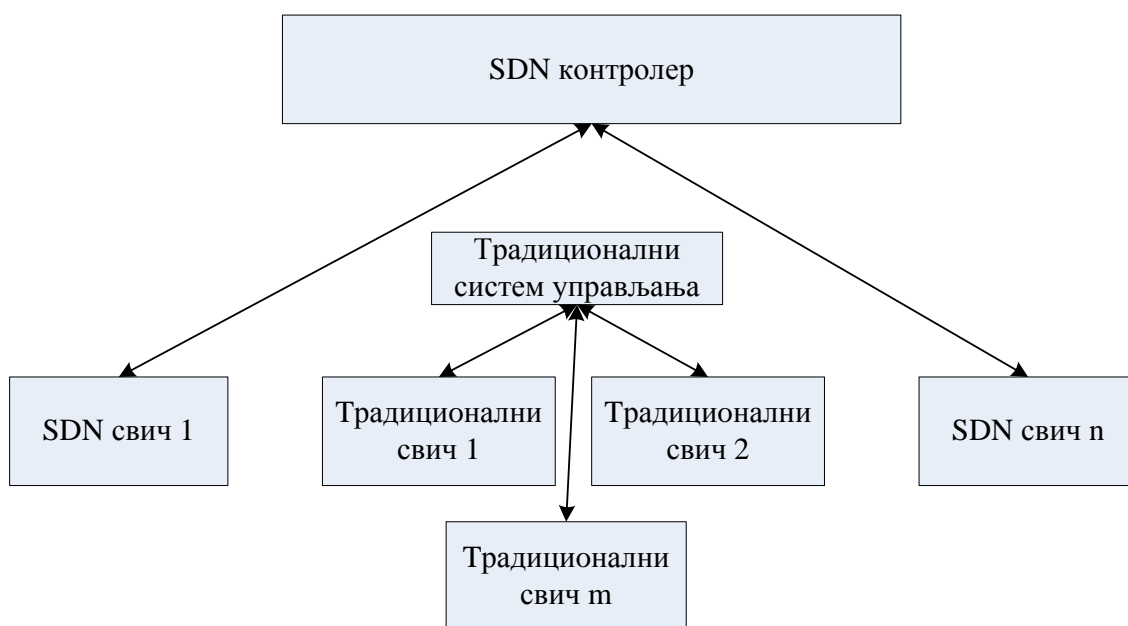
У процесу избора технологија за реализацију инфраструктуре у академском окружењу важан параметар је економска оправданост. Зато је приликом дефинисања модела инфраструктуре као један од полазних принципа наведено коришћење постојеће инфраструктуре уз имплементацију одговарајућих нових решења. Имајући на уму ове претпоставке предложено је хибридно решење које представља интеграцију постојеће инфраструктуре и SDN концепта.

Један од првих задатака приликом дефинисања модела хибридне SDN инфраструктуре је избор позиције SDN уређаја. На слици 34 је приказан *dual-stack* принцип, који подразумева примену хибридних свичева, који поред традиционалног начина прослеђивања саобраћаја користе и SDN принцип.



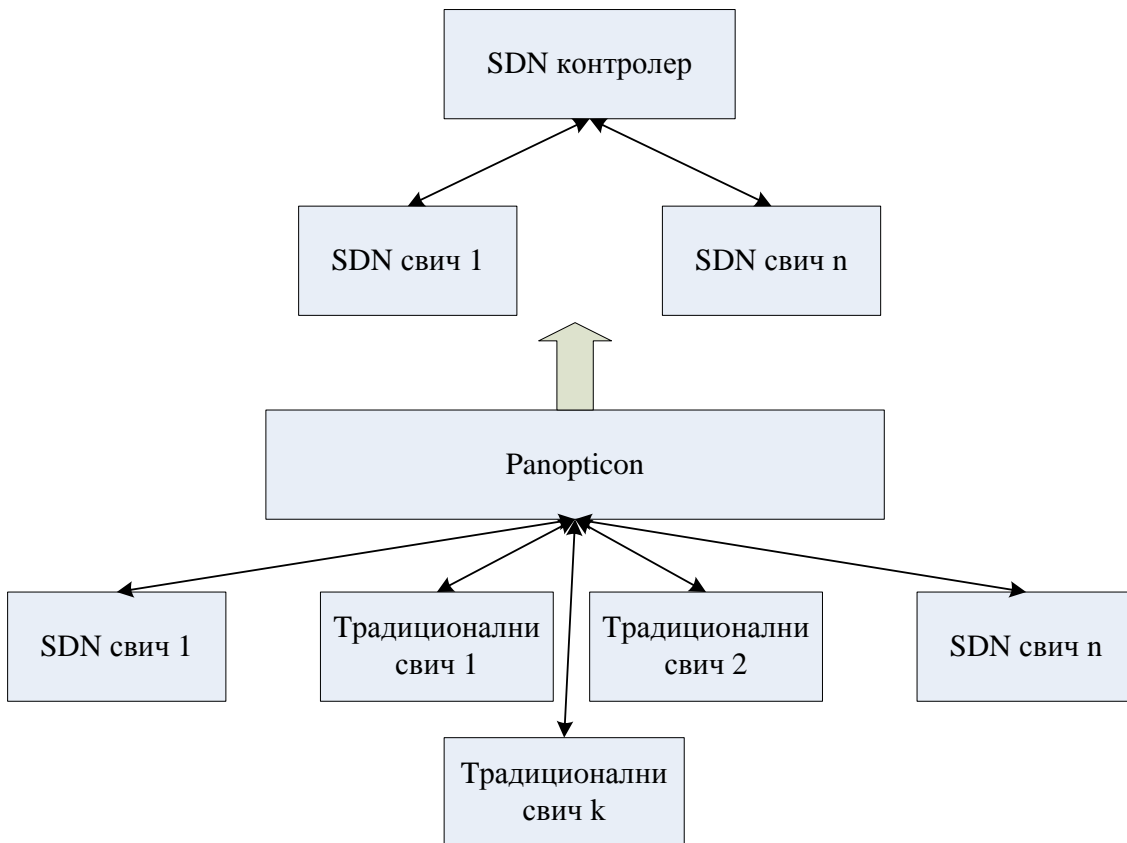
Слика 34: Хибридни *dual-stack* модел инфраструктуре

Други начин реализације хибридне инфраструктуре је приказан на слици 35 и може се назвати приступни SDN модел пошто се у овом случају SDN свичеви постављају на обод мреже [135], док се у централном делу и даље налазе традиционални уређаји. Иако овај модел омогућава примену нових функционалности одмах на уласку у мрежу, тј. на њеном ободу, реализација неких функционалности које захтевају дефинисање правила прослеђивања саобраћаја дуж целе путање кроз мрежу није могућа.



Слика 35: Приступни SDN модел инфраструктуре

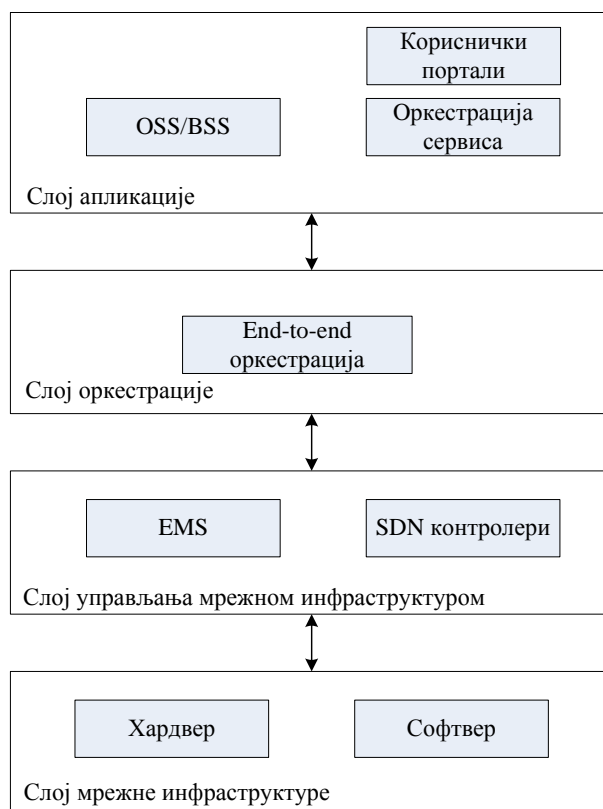
Трећи начин реализације хибридне инфраструктуре представља *Panopticon* решење [136], које је засновано на поставци да за сваку путању од било ког изворног до било ког одредишног уређаја мора да постоји бар један SDN свич. Овај модел је приказан на слици 36. Основу овог модела чини процес апстракције хетерогеног мрежног окружења у логичку SDN топологију.



Слика 36: *Panopticon* модел инфраструктуре [136]

Концептуални модел SDN инфраструктуре је приказан на слици 37. У структури се разликују следећи слојеви [137]:

- Слој мрежне инфраструктуре
- Слој управљања мрежном инфраструктуром
- Слој оркестрације
- Слој апликације



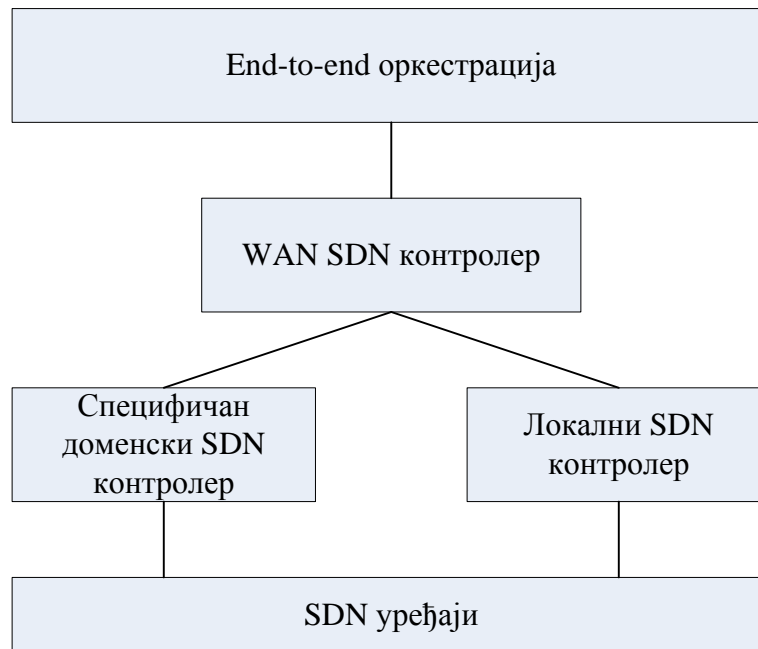
Слика 37: Концептуални модел SDN инфраструктуре

У табели 2 је дат приказ основних компоненти модела SDN инфраструктуре и њихов опис.

Табела 2: Компоненте модела SDN инфраструктуре

Компонента	Опис
Мрежна инфраструктура	Скуп хардверских и софтверских компоненти
EMS (<i>Element Management System</i>)	Систем за управљање специфичним мрежним елементима
SDN контролери	Управљају SDN уређајима
End-to-end оркестрација	Функција задужена за управљање животним током мрежних сервиса
OSS (<i>Operations Support Systems</i>) и BSS (<i>Business Support Systems</i>)	Подржавају скуп различитих функција, као што су праћење перформанси, <i>billing</i> , <i>trouble ticketing</i> итд.
Оркестрација сервиса	Функција која обезбеђује каталог сервиса корисничким порталима
Кориснички портали	Омогућавају корисницима дефинисање и праћење сервиса

SDN контролери представљају важну компоненту инфраструктуре пошто су задужени за дефинисање правила прослеђивања саобраћаја и целокупна мрежна интелигенција је ту централизована. На слици 38 је приказан модел управљања у SDN мрежи.



Слика 38: SDN контролери [137]

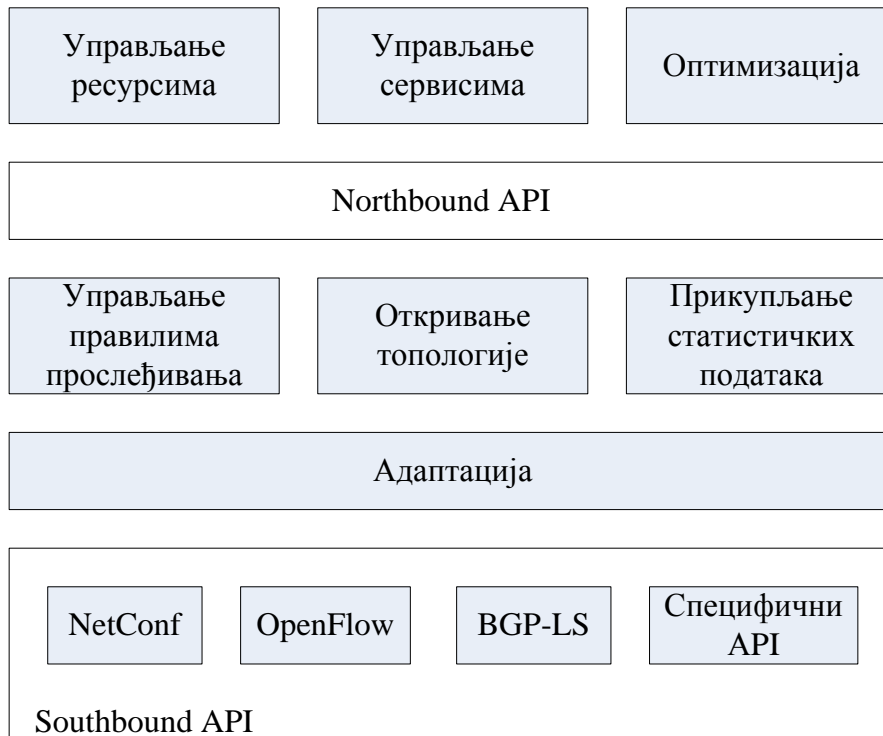
У оквиру сваког мрежног окружења постоји локални SDN контролер, при чему је претпостављена могућност да постоје и специфични SDN контролери за реализацију посебних функционалности у мрежи. Изнад њих у хијерархији се налази WAN SDN контролер који је задужен за дефинисање правила прослеђивања саобраћаја између различитих мрежних окружења.

WAN SDN контролер има задатак да подржи велики број функционалности, од којих су најважније [137]:

- Постојање одговарајућих *northbound* API-ја за апстракцију мреже
- Постојање функције адаптације
- Аутоматско управљање *end-to-end* сервисима
- Подршка за опрему различитих произвођача

- Оптимизација ресурса
- Откривање топологије
- Прикупљање статистичких података.

На слици 39 је приказан WAN SDN контролер.



Слика 39: WAN SDN контролер

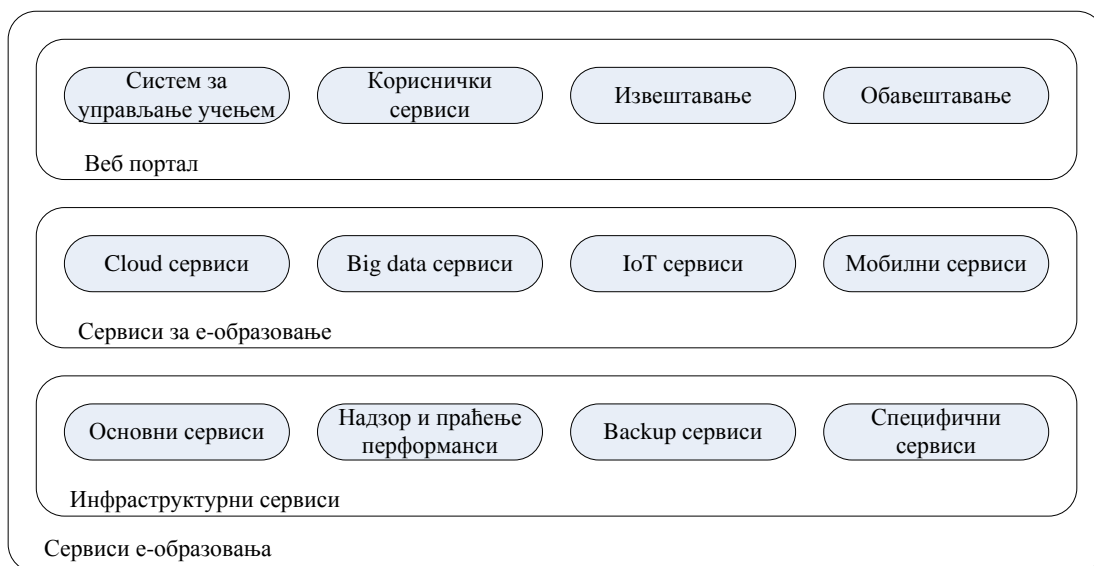
Функција управљања ресурсима и сервисима има задатак да омогући креирање, ажурирање и уклањање ресурса [137]. Поред тога, неопходно је да постоји и функција оптимизације, која на основу прикупљених статистичких података треба да примени одговарајуће мере у циљу оптималног коришћења ресурса (нпр. проналажење оптималне путање, *load-balancing* и сл). *Northbound API* омогућава апстрактну дефиницију сервиса. На пример, један сервис може бити дефинисан протоколом, начином комуникације, крајњим тачкама, одређеним атрибутима, као што је QoS. Функција управљања правилима прослеђивања дефинише тачан скуп правила на основу којих се врши прослеђивање саобраћаја. Ова правила се могу

дефинисати на основу податка о топологији или на основу улазних података које обезбеђују функција управљања ресурсима и сервисима. Функција откривања топологије има задатак да обезбеди улазне податке за функцију управљања ресурсима. Функција прикупљања статистичких података има задатак да обезбеди статистичке податке о ресурсима и сервисима. Функција адаптације врши транслацију података које користе функције управљања ресурсима и сервисима у специфичне податке које користе уређаји. *Southbound API* омогућава комуникацију са мрежним уређајима. Постоји читав низ протокола, од којих су најзначајнији *OpenFlow*, *NetConf* [138], *BGP-LS* [139], а постоји и могућност коришћења специфичних протокола.

4.4.4. Приказ сервиса е-образовања

На слици 40 су приказани сервиси е-образовања. Може се уочити вишеслојна структура у којој се разликује следећи типови сервиса:

- Инфраструктурни сервиси
- Сервиси за е-образовање
- Веб-портал



Слика 40: Сервиси е-образовања

Инфраструктурни сервиси представљају основу за функционисање осталих сервиса за е-образовање и омогућавају интерфејс ка другим сервисима у оквиру веб-портала. Основни сервиси представљају кључну компоненту и омогућавају мрежну конективност и комуникацију између слојева. Примери основних сервиса су DNS (енг. *Domain Name System*), размена електронске поште (енг. *e-mail*), сервиси за удаљени приступ мрежним ресурсима итд. Како би мрежна инфраструктура обезбедила одговарајући ниво квалитета сервиса, неопходна је реализација проактивног праћења перформанси и надзор мрежне инфраструктуре, што се реализује применом одговарајућих протокола (нпр. *Simple Network Management Protocol* - SNMP) и алата за управљање мрежом. *Backup* сервиси имају задатак да обезбеде креирање резервних копија мрежних ресурса како би се у случају инцидентних ситуација смањио утицај на сервисе корисника.

Сервиси за е-образовање имају задатак да подрже процес образовања и обухватају: *cloud*, *big data*, IoT (енг. *Internet of Things*) и мобилне сервисе. *Cloud* сервиси обухватају сервисе за администрацију *cloud* дела мрежне инфраструктуре, која треба да омогући постављање и конфигурацију мрежних ресурса у складу са захтевима наставника и студената приликом реализације наставе. *Big data* сервиси обухватају сервисе за управљање *big data* инфраструктуром. IoT сервиси обухватају сервисе који имају задатак да обезбеде ефикасно управљање паметним окружењима, као и одговарајуће образовне сервисе. Мобилни сервиси имају задатак да обезбеде све наведене функционалности сервиса за е-образовање коришћењем мобилних технологија. Поред образовних сервиса који студентима треба да омогуће савладавање градива и једноставан приступ ресурсима преко мобилних уређаја, за наставнике је важна и могућност удаљеног приступа за администрацију.

Веб-портал представља место интеграције свих сервиса за е-образовање. Преко њега се, између осталог, приступа систему за управљање учењем који омогућава корисницима креирање и управљање курсевима, управљање корисничким налозима, креирање наставних материјала итд. Пример једног система за управљање учењем је *Moodle* [140]. Веб-портал представља место на коме постоји

низ корисничких интерфејса за управљање мрежним ресурсима, колаборацију, комуникацију итд. За наставнике и администраторе је посебно важна могућност праћења перформанси инфраструктуре и приказ статистичких података о реализацији наставних процеса, као и могућност обавештавања о инцидентним ситуацијама, најављеним радовима у мрежи итд.

4.4.5. Приказ интеграције компоненти инфраструктуре

Компоненте инфраструктуре е-образовања могу се интегрисати применом веб-портала. На слици 41 је приказан модел интеграције компоненти инфраструктуре е-образовања. У питању је вишеслојна и модуларна структура, тако да се на једноставан начин може извршити проширење функционалности, тј. интеграција нових платформи. Интеграција са осталим компонентама система за е-образовање се врши применом концепта веб-сервиса.

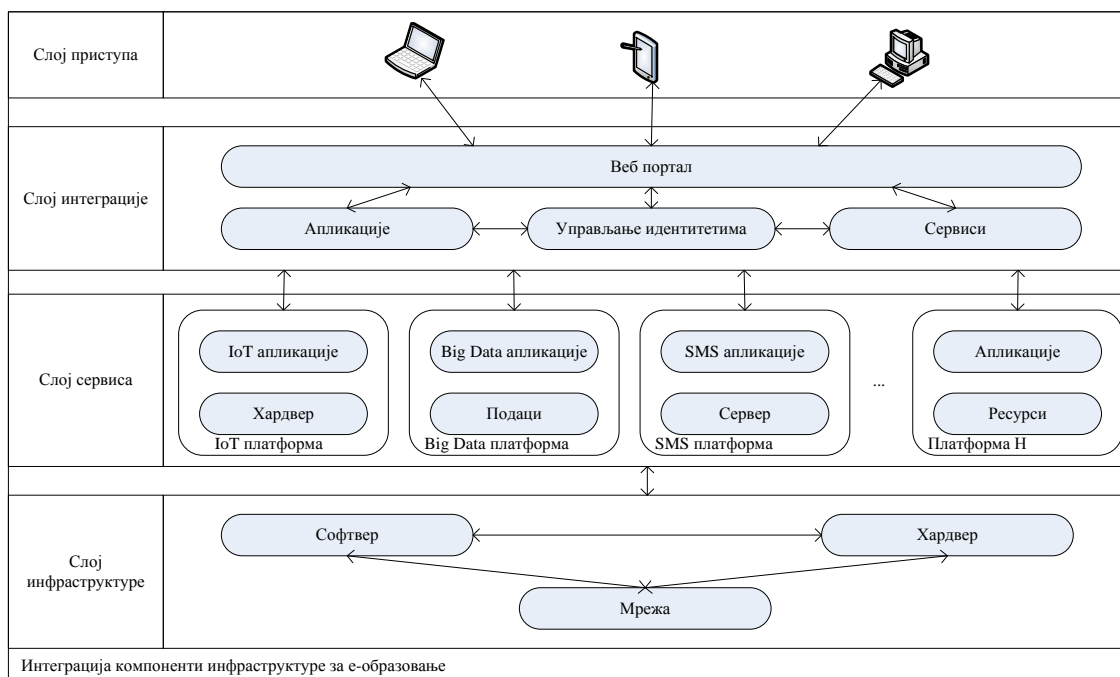
Слој приступа има задатак да обезбеди приступ мрежној инфраструктури без обзира на тип уређаја које наставници и студенти користе.

Слој интеграције представља интерфејс ка корисницима и заснован је на веб-порталу, који повезује све апликације и сервисе. На овом слоју постоји и систем за управљање идентитетима, који има улогу у аутентификацији и ауторизацији корисника.

На слоју сервиса се налазе одговарајуће релационе и нерелационе базе података неопходне за реализацију сервиса и апликација. Тренутно су препознате одређене групе апликација, као што су IoT апликације, *big data* апликације и SMS апликације, али захваљујући модуларности предложеног решења могуће је на једноставан начин извршити интеграцију нових апликација.

Слој инфраструктуре је најнижи слој и има задатак да обезбеди неопходне ресурсе за извршавање сервиса и апликација у складу са захтевима одређене

платформе, а чине га одговарајуће хардверске и софтверске компоненте између којих мора постојати мрежна конективност.



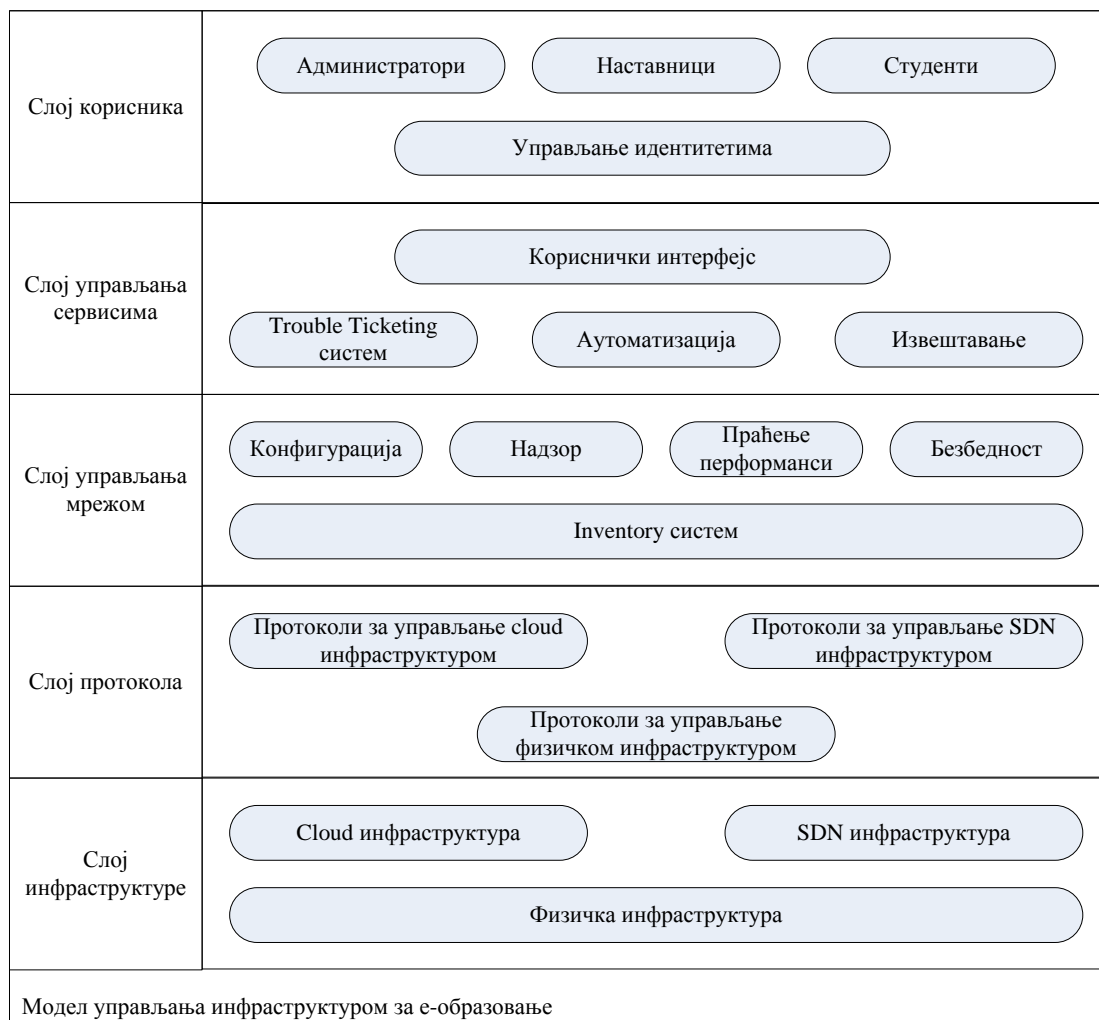
Слика 41: Интеграција компоненти инфраструктуре за е-образовање

4.4.6. Модел управљања инфраструктуром

На слици 42 је приказан модел управљања инфраструктуром за е-образовање.

Разликују се следећи слојеви:

- Слој инфраструктуре
- Слој протокола
- Слој управљања мрежом
- Слој управљања сервисима
- Слој корисника



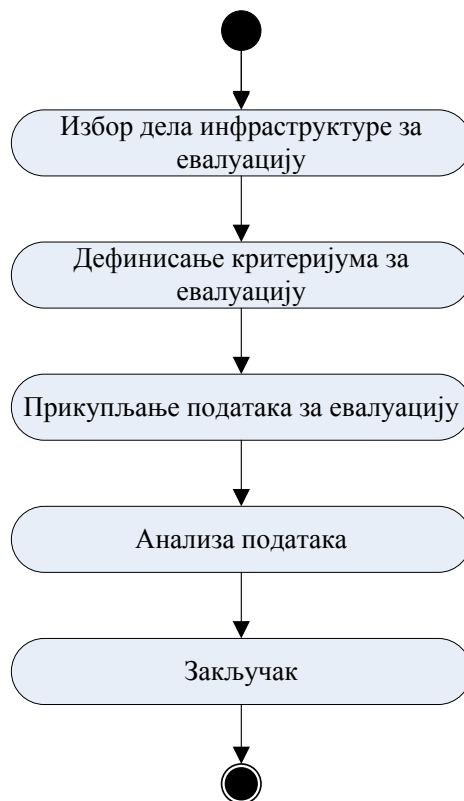
Слика 42: Управљање инфраструктуром за е-образовање

Слој инфраструктуре чине физичка, *cloud* и SDN инфраструктура. Управљање мрежном инфраструктуром се остварује применом одговарајућих протокола за управљање (енг. *management protocol*). Компоненте слоја управљања мрежом су конфигурација (енг. *configuration management*), надзор (енг. *fault management*), праћење перформанси (енг. *performance management*) и безбедност преноса информација. За функционисање овог слоја неопходан је одговарајући *inventory* систем. На слоју управљања сервисима постоји кориснички интерфејс преко кога се врши управљање. Поред њега постоји и *trouble ticketing* систем, чија је улога отварање тикета за уочене проблеме у функционисању система, као и систем за извештавање о уоченим проблемима, перформансама система и приказ различитих статичких података. Карактеристика овог слоја је и примена функције

аутоматизације. Највиши слој представља слој корисника који приступају систему за управљање инфраструктуром. Администратори имају највише привилегије и могућност управљања комплетном инфраструктуром. Наставници имају одређена права која им омогућавају креирање мрежне инфраструктуре неопходне за реализацију курсева за које су задужени, док студенти имају најмање привилегије, али довољне за реализацију наставних задатака и пројеката у оквиру курсева које похађају.

4.5. Евалуација перформанси

Евалуација предложеног модела (Слика 43) састоји се из следећих активности: избор дела инфраструктуре за евалуацију, дефинисање критеријума за евалуацију, прикупљање података за евалуацију, анализа података, закључак.

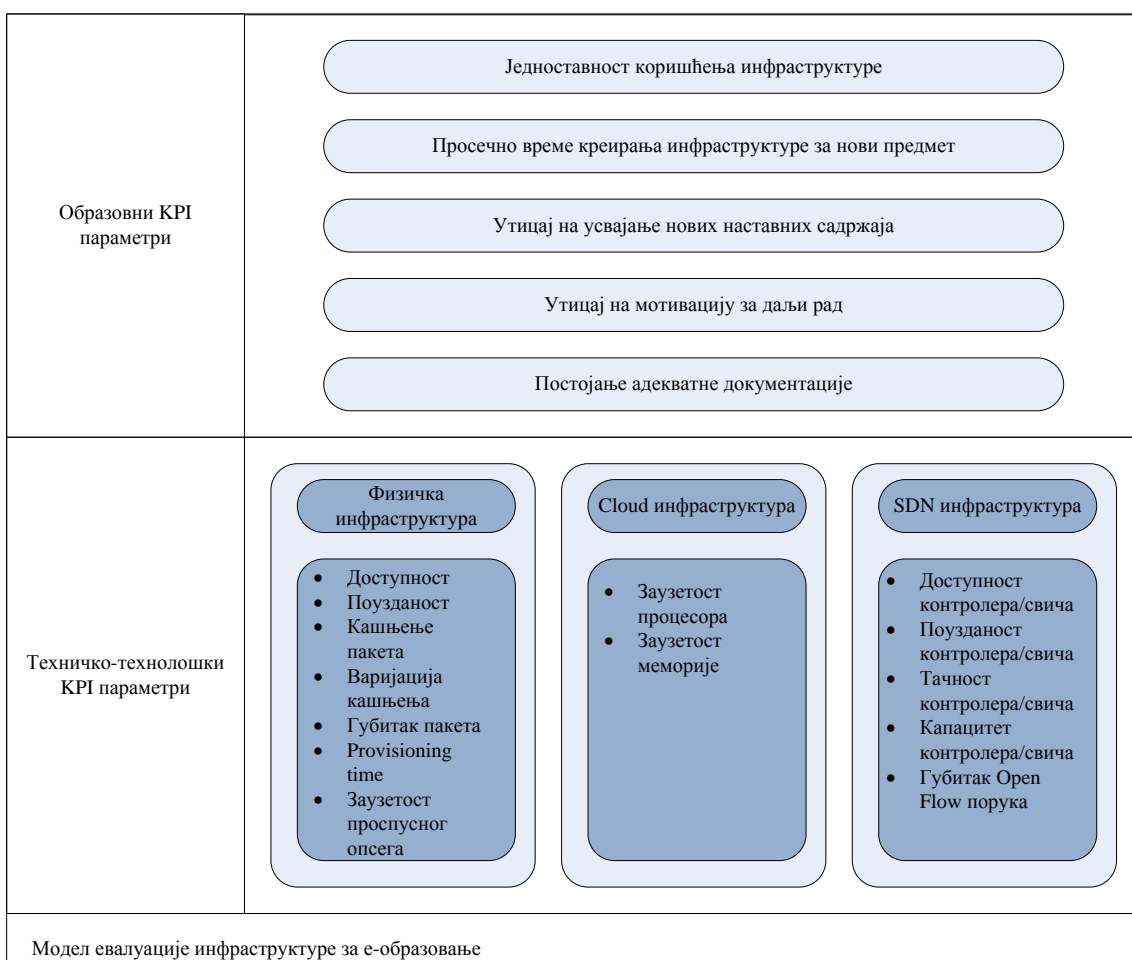


Слика 43: Евалуација предложеног модела

Критеријуми по којима се врши евалуација се могу поделити у две категорије:

- Техничко-технолошки
- Образовни.

На слици 44 је приказан модел евалуације инфраструктуре за е-образовање. Приказани су техничко-технолошки и образовни KPI (енг. *Key Performance Indicator*) параметри које треба пратити како би се обезбедио одговарајући ниво квалитета и доступности сервиса.



Слика 44: Модел евалуације инфраструктуре за е-образовање

Детаљнији приказ техничко-технолошких параметара је дат у табели 3.

Табела 3: Техничко-технолошки KPI параметри инфраструктуре за е-образовање

Назив	Опис	Јединица
Физичка инфраструктура		
Доступност	Време у коме је уређај/интерфејс оперативан	%
Поузданост	Способност уређаја/интерфејса да извршава своју функцију	%
Кашњење пакета	Време преноса пакета између две тачке у мрежи	msec
Губитак пакета	Број пакета који нису примљени у односу на укупан број послатих пакета	%
Варијација кашњења (<i> jitter </i>)	Варијанса кашњења пакета између две тачке у мрежи	msec
<i> Provisioning time </i>	Време активације сервиса (време од дефинисања захтева за сервис до успоставе сервиса)	min
Заузетост пропусног опсега	Количина заузетог пропусног опсега у односу на укупни капацитет линка	%
Cloud инфраструктура		
Заузетост процесора	Количина заузетости процесора односу на укупни капацитет процесора	%
Заузетост меморије	Количина заузете меморије у односу на укупну меморију	%
SDN инфраструктура		
Доступност контролера	Време у коме је контролер недоступан у односу на укупно време рада контролера	%
Поузданост контролера	Способност контролера да извршава своју функцију	%
Тачност контролера (<i> accuracy </i>)	Број пакета прослеђених у складу са правилима дефинисаним на контролеру у односу на укупан број пакета	%
Капацитет контролера	Заузетост ресурса контролера у односу на укупан капацитет	%
Доступност свича	Време у коме је свич недоступан у односу на укупно време рада свича	%
Поузданост свича	Способност свича да извршава своју функцију	%
Капацитет свича	Заузетост ресурса свича (<i> flow </i> табела) у односу на укупан капацитет	%
Тачност свича (<i> accuracy </i>)	Број пакета прослеђених у складу са правилима дефинисаним на свичу (<i> flow </i> табела) у односу на укупан број пакета	%
Губитак <i> OpenFlow </i> порука	Број <i> OpenFlow </i> порука које нису примљене у односу на укупан број послатих <i> OpenFlow </i> порука	%

Табела 4 приказује образовне KPI параметре. Наведени параметри се могу мерити применом прилагођеног USE упитника [141].

Табела 4: Образовни KPI параметри инфраструктуре за е-образовање

Назив	Опис
Време креирања инфраструктуре за нови наставни предмет	Просечно време од дефинисања захтева до реализације инфраструктуре за нови наставни предмет
Приступачност	Једноставност коришћења SDN инфраструктуре за е-образовање
Оцена знања	Ниво стеченог знања коришћењем SDN инфраструктуре за е-образовање
Мотивација	Степен мотивисаности студента да стекне и прошири знања коришћењем SDN инфраструктуре за е-образовање
Документација	Постојање адекватне документације за коришћење SDN инфраструктуре за е-образовање
Задовољство студената	Степен задовољства студената у процесу коришћења SDN инфраструктуре за е-образовање
Задовољство наставника	Степен задовољства наставника у процесу коришћења SDN инфраструктуре за е-образовање

Евалуација перформанси инфраструктуре за е-образовање засноване на софтверски дефинисаним мрежама је комплексан процес. Проактивно праћење техничко-технолошких параметара треба да омогући поуздан рад и висок степен доступности уређаја и сервиса у академском окружењу. Подједнако је важно праћење и образовних параметара, пошто они представљају улазне податке за унапређење процеса едукације. Евалуација перформанси на конкретном систему је предмет будућих истраживања.

5. ОБРАЗОВНИ МОДУЛ ЗА УЧЕЊЕ СОФТВЕРСКИ ДЕФИНИСАНИХ МРЕЖА

Процес стицања знања у области рачунарских мрежа је комплексан пошто захтева да студенти стекну знања о различитим мрежним технологијама, протоколима и мрежним уређајима. Увођење концепта програмабилности у мреже, тежња ка унификацији рачунарских мрежа и примени решења отвореног стандарда представљају основне смернице које један образовни модул у овој области мора да подржи. Основни проблем који постоји је последица чињенице да је SDN нова технологија која се развија, што утиче и на потребу за континуалним изменама образовног модула. Да би процес едукације у области софтверски дефинисаних мрежа био успешан, неопходно је да студентима пружи фундаментална знања о мрежним концептима и начинима да у будућности нове технологије примене у продукционим мрежним окружењима са високим степеном аутоматизације. Зато се у процесу едукације у области софтверски дефинисаних мрежа посебна пажња мора посветити практичном раду и реализацији лабораторијских вежби [142]. Све ово треба студенте да припреми за будући рад и изазове у реалним мрежним окружењима. Циљ дисертације је да представи образовни модул за учење софтверски дефинисаних мрежа и да допринос у процесу формализације процеса едукације у области нових мрежних технологија, као што је SDN.

5.1. Пројектни захтеви

Основни циљ пројекта јесте имплементација пилот образовног модула за учење софтверски дефинисаних мрежа у оквиру Лабораторије за електронско пословање на Факултету организационих наука у Београду. Имплементирани образовни модул треба да унапреди процес образовања у области информационо-комуникационих технологија, оствари позитиван утицај на исходе учења студената и процес припреме образовних ресурса учини једноставним.

У реализацији пројекта препознате су следеће основне активности:

- Припрема наставних материјала
- Припрема инфраструктурних ресурса (хардвер, софтвер, мрежа)
- Реализација наставног процеса
- Евалуација примене софтверски дефинисане инфраструктуре

Евалуација се врши са образовног аспекта. Образовни параметри се посматрају у току реализације образовног модула за учење софтверски дефинисаних мрежа праћењем перформанси образовног процеса. Евалуација образовних параметара се обавља на два начина: обрадом резултата испита за проверу стеченог знања студената и обрадом упитника који студенти попуњавају. На основу добијених резултата се утврђује да ли и у којој мери образовни модул за учење софтверски дефинисаних мрежа може допринети унапређењу образовног процеса и исхода учења.

5.2. Процес креирања образовног модула за учење софтверски дефинисаних мрежа

У циљу стицања знања у области нових мрежних технологија, као што је SDN, потребно је превазићи недостатке који постоје код традиционалног начина едукације [143]. Иако постоји одређен број образовних модула који су реализовани на универзитетима широм света [144], у пракси је показано да реализација оваквог модула представља комплексан задатак. Пре свега, SDN образовни модул захтева да студенти стекну одређена знања из области програмирања [10], али и да стекну практично искуство [145]. Ово су основни разлози за примену педагошког модела заснованог на *problem-based* учењу или *flipped* учионицама [146]. Пример добре праксе примене *problem-based* учења применом процеса емулације мреже је описан у [147].

Приликом креирања SDN образовног модула неопходно је креирати основу за једноставно разумевање постојећих решења и ефикасну примену будућих

технологија [136] имајући на уму да је SDN нова технологија која се брзо развија. Како би дизајн био ефикасан и интероперабилан и омогућио даље измене применом развијених апликација, користе се *open-standard* решења.

На основу детаљне анализе литературе и примера добре праксе закључено је да SDN образовни модул треба да покрије следеће области [148]:

- Разлике између SDN и традиционалних мрежа, као и између дистрибуираног и централизованог управљања мрежом.
- Апстракција и виртуелизација у SDN мрежама.
- Програмирање SDN контролера и API-ја заснованих на *open-source* протоколима и стандардима, као што је *OpenFlow* протокол.
- Случајеви коришћења у хибридном мрежама са традиционалним и виртуелним уређајима.
- SDN апликације, укључујући *firewall* и *load balancing*.

Процес креирања SDN образовног модула се може поделити на пет фаза, као што је приказано на слици 45.



Слика 45: Фазе у креирању SDN образовног модула [148]

5.3. Пројектовање и имплементација образовног модула за учење софтверски дефинисаних мрежа

5.3.1. Фаза припреме SDN образовног модула

Дефинисање садржаја и циља предмета у складу са примерима добре праксе и потребама и могућностима учења студената представља први корак у фази припреме SDN образовног модула. У овој фази се дефинише метод учења, а препорука је да се користи комбинација различитих приступа. Поред традиционалног предавања у учионици (енгл. *face-to-face*), образовни модул захтева примену и мултимедијалних материјала. Практичан рад студената је важан део овог образовног модула, па је потребно припремити писане инструкције и решења за све лабораторијске вежбе. Такође, у овој фази се дефинише процес провере стеченог знања и он представља комбинацију усмене одбране пројекта и завршног теста са питањима са вишеструким понуђеним одговорима (енгл. *multiple-choice*). На крају реализације образовног модула се помоћу упитника врши евалуација мишљења студената о одржаном образовном модулу.

5.3.2. Фаза планирања SDN образовног модула

Табела 5 приказује предложену структуру SDN образовног модула. Образовни модул је подељен на три дела. У првом делу се изучавају основни концепти рачунарских мрежа као што су рутирање, *switching*, IP адресирање итд. Практичан рад у овом делу образовног модула је заснован на хардверским мрежним уређајима и мрежним симулаторима. У другом делу SDN образовног модула студенти се упознају са виртуелизацијом и *cloud computing* технологијом користећи *OpenStack* [149]. Завршни део образовног модула врши интеграцију стеченог знања и изучава програмабилне компоненте мреже специфичне за SDN. Практичан рад у овом сегменту је заснован на коришћењу Mininet симулатора [106] и *OpenDaylight* SDN контролера [75].

Табела 5: Структура SDN образовног модула [148]

Тема	Циљ	Садржај	Алати и технологије	Метод учења		
Основни концепти рачунарских мрежа	Стицање знања у области рачунарских мрежа и технологија неопходних за ефикасну реализацију комплексних мрежних архитектура	<ul style="list-style-type: none"> • Традиционалне рачунарске мреже • Класификација рачунарских мрежа • Референтни мрежни модели • Протоколи у рачунарским мрежама • Недостаци традиционалних рачунарских мрежа 	<ul style="list-style-type: none"> • Мрежни уређаји • Мрежни симулатор 	Предавање: <i>Face-to-face</i> предавање Писане инструкције Мултимедијални материјали		
	Примена теоријског знања, квантитативне и квалитативне методе за креирање рачунарских мрежа					
Виртуелизација и <i>cloud computing</i>	Стицање знања у области виртуелизације, са фокусом на виртуелизацију мреже	<ul style="list-style-type: none"> • Концепт виртуелизације • Виртуелизација мреже • <i>Open vSwitch</i> • Основни концепти <i>cloud computing</i> технологије • <i>OpenStack</i> 	<ul style="list-style-type: none"> • <i>OpenStack</i> • <i>Open vSwitch</i> 	Учење: Практичан рад Пројекат/ <i>case study</i> Квизови		
	Моделирање нових решења и решавање тренутних проблема у имплементацији <i>cloud computing</i> инфраструктуре и сервиса					
	Креирање модела <i>cloud</i> инфраструктуре, апликација и <i>cloud</i> сервиса					
SDN	Стицање знања у области SDN у циљу превазилажења ограничења традиционалних мрежа	<ul style="list-style-type: none"> • SDN архитектура • SDN протоколи • SDN мрежна инфраструктура • <i>OpenFlow</i> протокол • SDN контролери • SDN имплементација • Евалуација SDN • Емулација SDN • SDN у <i>OpenStack</i> окружењу • SDN примена 	<ul style="list-style-type: none"> • Mininet • <i>Open Daylight</i> 	Евалуација: Упитник		
	Упознавање са техничким и организационим аспектима SDN-а кроз анализу модела, техничких решења и безбедносних аспеката SDN технологије					

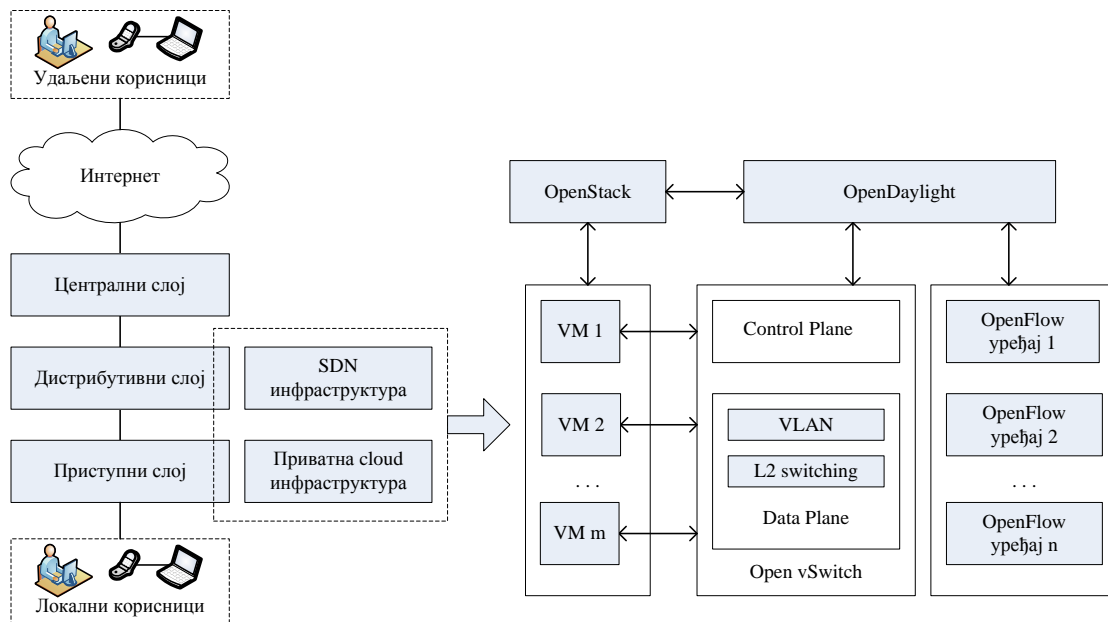
5.3.3. Фаза развоја инфраструктуре за SDN образовни модул

За реализацију SDN образовног модула неопходно је, поред креирања наставних материјала, креирати и одговарајућу мрежну инфраструктуру. Инфраструктура је такође заснована на SDN технологији. На овај начин креирано решење омогућава коришћење исте мрежне инфраструктуре од стране различитих корисника са различитим привилегијама, при чему свако од њих има могућност креирања сопствене *end-to-end* мреже, која је потпуно одвојена од осталих [122]. Ова предност је посебно значајна у академским институцијама са ограниченим мрежним ресурсима.

Дизајниран је хибридни мрежни модел, који комбинује традиционалну и SDN инфраструктуру [120]. Хибридни мрежни модел је погодан за SDN образовни модул зато што задржава постојећу инфраструктуру, а у исто време уводи нову димензију програмабилности. Овај модел обезбеђује поуздану, скалабилну и безбедну инфраструктуру, коју карактерише следеће:

- Ефикасна примена постојећих ресурса
- Аутоматска контрола и управљање мрежом
- Минимално време креирања нових сервиса и апликација
- Побољшање отпорности на отказе.

Слика 46 приказује концептуални модел инфраструктуре, који описује имплементацију SDN технологије у *cloud* инфраструктуру академске институције. Више детаља о архитектури развијене *cloud* инфраструктуре може се наћи у [37]. Имплементирани модел је заснован на интеграцији *open-source OpenDaylight* контролера у мрежну инфраструктуру засновану на *OpenStack* решењу. То значи да морају да постоје и SDN и традиционална управљачка раван (енгл. *control plane*). Правила за прослеђивање саобраћаја, који креће или је намењен SDN контролисаним портovima, дефинише се искључиво на *OpenDaylight* контролеру. Ови портови треба да буду доступни и ван мреже како би постојала потпуна конективност. Циљ оваквог решења је да се ограничи комплексност процеса управљања мрежом.

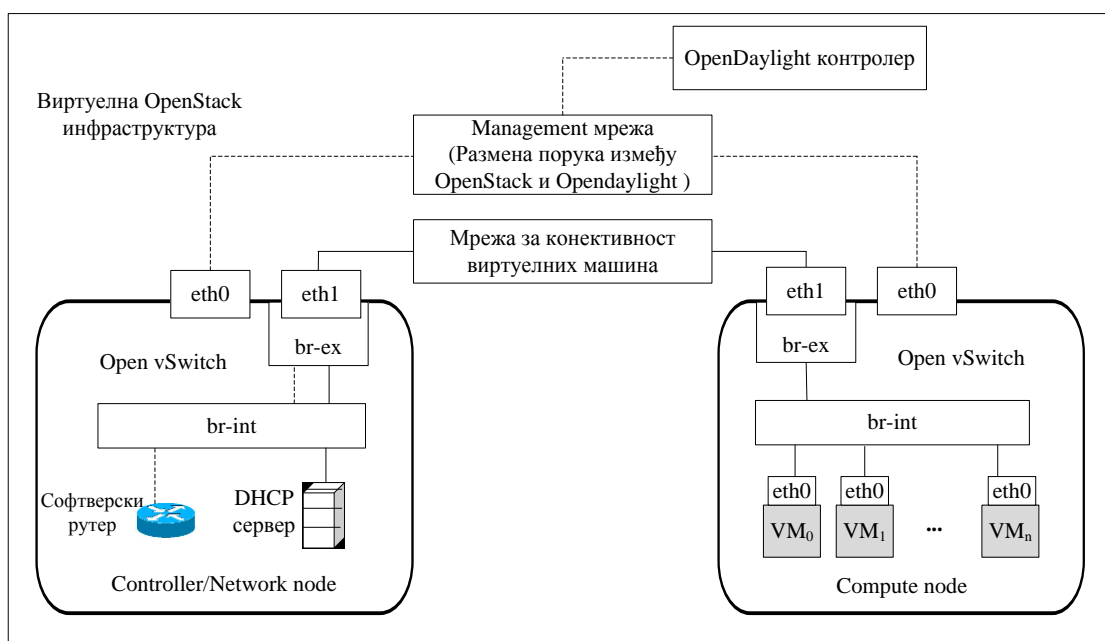


Слика 46: Концептуални модел инфраструктуре заснован на *OpenStack* и *OpenDaylight* решењу [148]

OpenDaylight контролер преко *northbound* API-ја остварује интеракцију са *OpenStack*-ом, који због модуларне архитектуре има могућност интеграције са другим технологијама. Основа ове интеграције је конекција између *OpenDaylight* драјвера и *OpenStack* Neutron ML2 *plugin*-а. За *southbound* конфигурацију Open vSwitch-а на *compute node*-у се користи OVSDB протокол [81]. Комуникација са *OpenFlow-enabled* уређајима се остварује преко *OpenFlow* протокола. *Open vSwitch* као *multilayer* софтвер омогућава функцију виртуелизације и повезује виртуелне машине у оквиру KVM *kernel* модула у *Linux*-у.

Као што је приказано на слици 47, *OpenStack* се користи за реализацију многих функционалности као што су рутирање, *switching*, *firewall* итд. Основни циљ примене *OpenStack*-а је коришћење предности *open-source* решења и његових функционалности за креирање приватног *cloud*-а. Његов мрежни модул, Neutron, се инсталира независно од осталих модула и може бити подељен на неколико виртуелних хостова конфигурисаних на једном чвору (енгл. *node*). Све виртуелне машине се повезују на *Open vSwitch* преко *integration bridge*-а (*br-int*). *Integration*

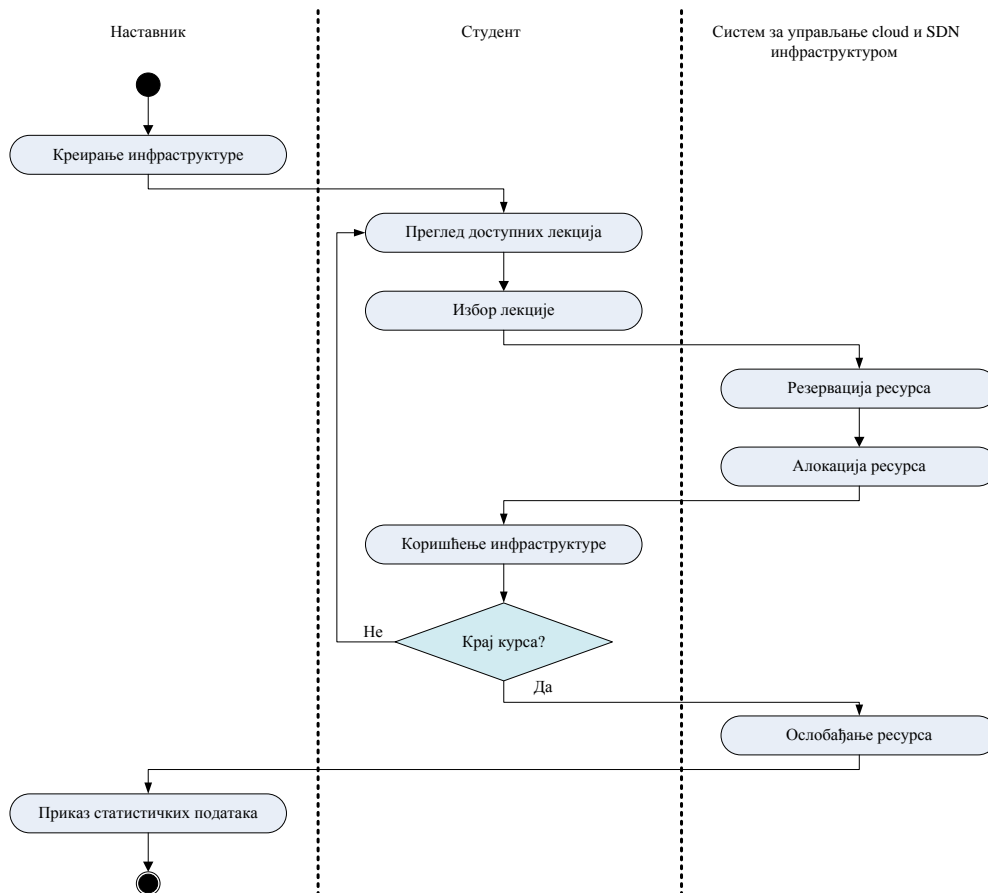
bridge повезује виртуелне машине на другом слоју OSI модела и подржава различите протоколе. Он раздваја повезане виртуелне портове на основу додељених обележја (енгл. *tag*) логичке мреже и успоставља одговарајућа правила за прослеђивање токове података. Такође, *Open vSwitch* се користи и за комуникацију између виртуелних инстанци и уређаја ван *OpenStack* домена, па се зато на *external bridge*-у (*br-ex*) конфигурише IP адреса. Применом *Open vSwitch* решења више логичких мрежа може да дели један или више физичких интерфејса уз потпуну изолацију.



Слика 47: Логички дијаграм *OpenDaylight* и *OpenStack* интеграције [148]

5.3.4. Фаза имплементације SDN образовног модула

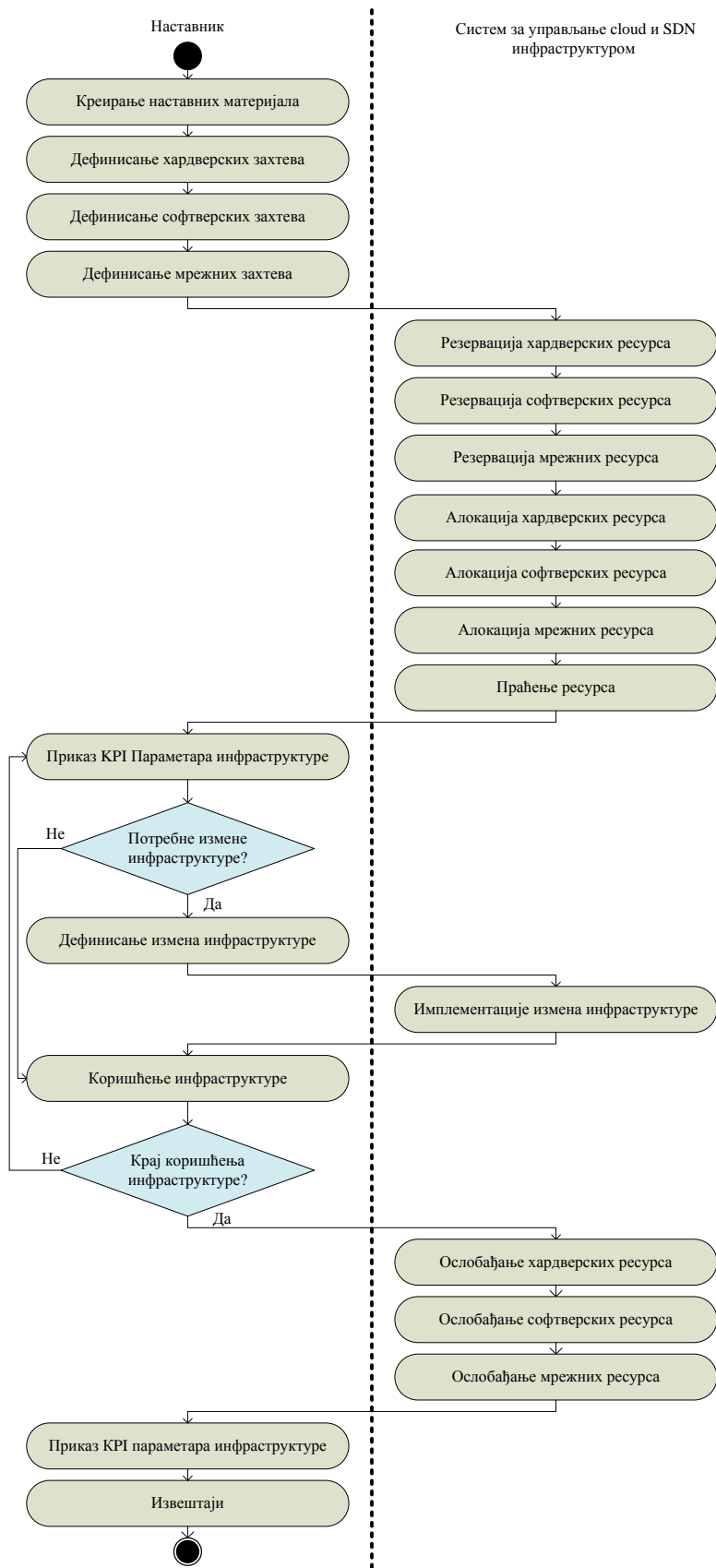
У фази имплементације наставници и студенти користе наставне материјале и мрежну инфраструктуру за образовни модул. На слици 48 је приказан процес реализације наставе.



Слика 48: Процес реализације наставе

Наставник у складу са дефинисаним предметом истраживања у оквиру одређеног образовног модула врши креирање потребног мрежног окружења. Студент може похађати наставу из више предмета и имати могућност приступа већем броју лекција, па је потребно да изабере одговарајућу тему. Након тога се врши резервација потребних хардверских, софтверских и мрежних ресурса како би студент имао на располагању комплетну инфраструктуру за одговарајућу лекцију или пројекат. По завршетку одређеног пројекта или лекције заузети ресурси се ослобађају. Наставник има могућност приказа статистичких података о перформансама мрежне инфраструктуре, реализацији образовног модула, постигнутим резултатима студената итд.

Слика 49 приказује процес алоцирања потребних ресурса приликом реализације SDN образовног модула.



Слика 49: Процес креирања инфраструктуре за SDN образовни модул [148]

Након дефинисања наставних тема које ће се обрађивати у оквиру новог предмета, наставник креира наставне материјале и дефинише потребне хардверске, софтверске и мрежне ресурсе. Ови задаци се могу реализовати преко јединственог портала, који је интегрисан са системом за управљање учењем, као и системом за управљање *cloud* и SDN инфраструктуром. Затим следи резервација и алокација ресурса потребних за реализацију наставе. Предвиђена је и могућност измене дефинисаних хардверских, софтверских и мрежних захтева на основу показатеља перформанси инфраструктуре (KPI). Након завршетка коришћења мрежне инфраструктуре, заузети мрежни ресурси се ослобађају и стављају на располагање другим корисницима. По завршетку образовног модула коришћени ресурси се ослобађају, а наставник преко веб-портала има могућност прегледа KPI параметара инфраструктуре, као и приказа различитих статистичких података везаних за реализацију образовног модула.

5.3.5. Фаза евалуације SDN образовног модула

Фаза евалуације је последња фаза у процесу креирања образовног модула. Евалуација се може извршити са образовног и техничког аспекта. Евалуација са образовног аспекта се врши провером стеченог знања студената у форми усменог или писменог испита, у оквиру кога се решавају задаци и тестови. Важан параметар у евалуацији представља мишљење студената о SDN образовном модулу и оно се испитује коришћењем одговарајућег упитника. Техничка евалуација је заснована на праћењу параметара мреже, где се поред параметара за традиционалне мреже прате и параметри за SDN инфраструктуру. Прикупљени подаци су од великог значаја за наставнике који на основу њих могу да измене и прилагоде метод учења и начин реализације образовног модула.

5.4. Анализа исхода образовног процеса

Евалуација предложеног решења је извршена у оквиру Лабораторије за електронско пословање на Факултету организационих наука у Београду у форми два пилот SDN образовна модула у коме су учествовали студенти основних, мастер, специјалистичких и докторских студија. За реализацију образовног модула је коришћено тестно окружење које интегрише *OpenStack*, *OpenDaylight*, и постојећу инфраструктуру засновану на *Mikrotik* и *Cisco* уређајима.

Циљ евалуације је био да се добију одговори на два питања:

П1: Који је утицај образовног модула на знање студената?

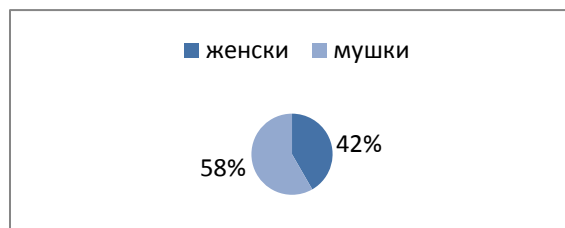
П2: Какво је мишљење студената о учењу SDN технологије применом развијеног приступа?

5.4.1. Образовни модул 1

Први SDN пилот образовни модул је реализован у току зимског семестра школске 2015/16. године.

Учесници

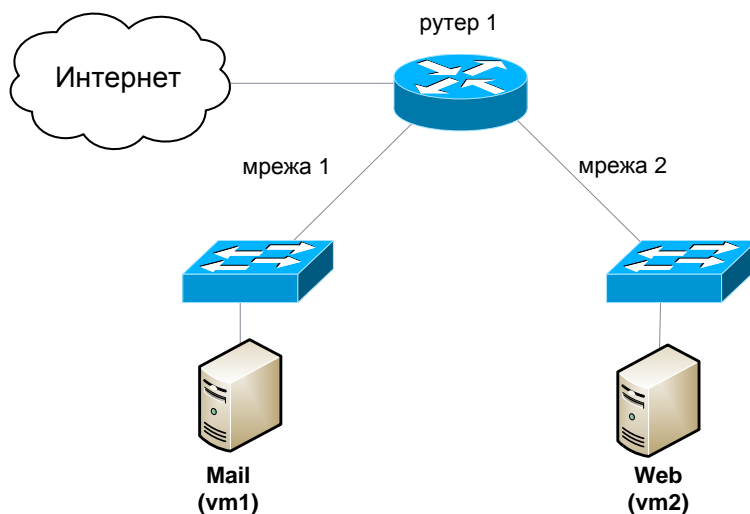
SDN образовни модул је похађало 24 студента смера Електронско пословање, при чему су сви имали слично предзнање и интересовање у области рачунарских мрежа и интернет технологија. На слици 50 је приказана полна структура полазника.



Слика 50: Полна структура полазника образовног модула

Процедура

Образовни модул је садржао предавања и практичан, лабораторијски део. У оквиру теоријског дела студенти су упознати са основним концептима рачунарских мрежа, *cloud computing* технологије и софтверски дефинисаних мрежа. Завршни задатак је био да се на основу дефинисаних захтева креира мрежна инфраструктура за одређену компанију. Студенти су на располагању имали преконфигурисане виртуелне машине у *OpenStack* окружењу, као и писане инструкције. На слици 51 је приказан пример задатка који су студенти решавали, односно пример мрежне топологије коју је требало реализовати коришћењем доступних ресурса.



Слика 51: Пример мрежне топологије за завршни задатак

Инструменти

У процесу евалуације су коришћена два инструмента:

- Тест знања
- Упитник за испитивање ставова студената о образовном модулу.

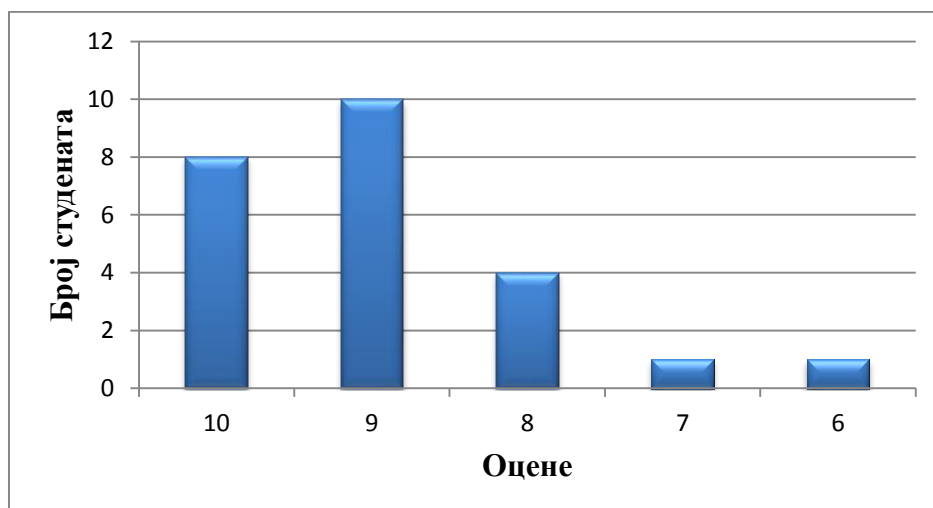
По завршетку образовног модула извршена је евалуација стеченог знања

студената применом теста знања у циљу добијања одговора на питање П1. Тест је обухватио 10 питања и базиран је на питањима типа *multiple-choice*. Сва питања су имала исти тежински фактор. Студенти су добили оцене на скали од 5 до 10 (5 – није положио, 6 – најнижа оцена за пролаз, 10 – највиша оцена за пролаз).

Поред теста знања, студенти су имали задатак да попуне упитник како би се испитало њихово мишљење о одржаном образовном модулу и добио одговор на питање П2. Коришћен је USE упитник [141], а питања су груписана у четири категорије: општи утисак о SDN решењу, једноставност коришћења, утицај на стицање знања и садржај образовног модула. Питања су базирана на *Likert*-овој скали са пет понуђених одговора.

Анализа резултата

Просечна оцена коју су студенти остварили на тесту знања износи 8.96 од максималних 10 поена, при чему је стандардна девијација 1.04. На слици 52 су приказани остварени резултати студената на тесту знања. Анализом одговора студената долази се до закључка да је већина студената дала тачне одговоре на питања из области основних концепата рачунарских мрежа и виртуелизације, док су студенти имали извесне потешкоће са питањима о софтверски дефинисаним мрежама.



Слика 52: Резултати теста знања

Табела 6 приказује сумарне резултате анализе упитника (\bar{X} – средња вредност, од 1 до 5; δ – стандардна девијација) који су студенти попуњавали на крају образовног модула. Коришћена је петостепена скала (У потпуности се слажем – 5; Слажем се – 4; Нисам сигуран – 3; Не слажем се – 2; У потпуности се не слажем – 1).

Табела 6: Анализа резултата упитника

Питања	\bar{X}	δ
Питања везана за SDN		
Имам искуства у раду са <i>cloud computing</i> технологијама	1.58	0.65
SDN решење је ефикасно	3.62	0.77
SDN решење се може применити у пракси	3.71	0.86
Примена SDN решења може смањити време извршења задатака	3.83	0.70
Примена SDN решења може побољшати едукацију у области рачунарских мрежа	4.04	0.81
Питања везана за једноставност коришћења		
Конфигурација виртуелног мрежног окружења применом <i>OpenStack</i> решења је једноставна	3.12	0.80
Решавање задатака захтева мали број корака	2.96	1.08
Задатак се може решити без писаних инструкција	2.75	0.94
За решавање задатка је потребна помоћ асистента	3.71	0.86
Питања везана за једноставност учења		
Основе конфигурације <i>OpenStack</i> и <i>OpenDaylight</i> решења се могу научити лако	2.92	0.78
Основе конфигурације <i>OpenStack</i> и <i>OpenDaylight</i> решења се могу научити брзо	2.87	0.80
Питања везана за садржај образовног модула		
Задатак је био занимљив	3.92	0.72
Задатак је био користан	4.00	0.72
Знања стечена при изради задатка се могу применити у пракси	3.71	0.91

Већина студената није имала претходно искуство у раду са *cloud computing* технологијом, али 80% њих сматра да SDN решење засновано на *OpenStack* и *OpenDaylight* решењу може имати позитиван утицај, посебно у образовном окружењу. Студенти су оценили да је тежина задатка и креирање мреже компаније било на средњем нивоу, а већина (око 75%) сматра да је за израду задатка потребна помоћ наставника. Општа оцена студената јесте да су задовољни

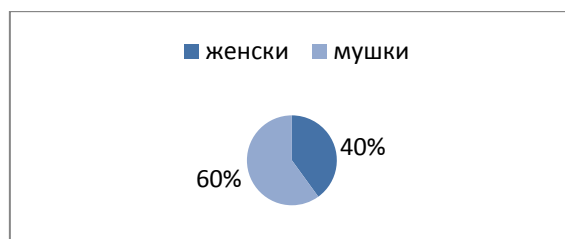
одржаним SDN образовним модулом, а око 80% испитаних сматра да је задатак био интересантан и користан.

5.4.2. Образовни модул 2

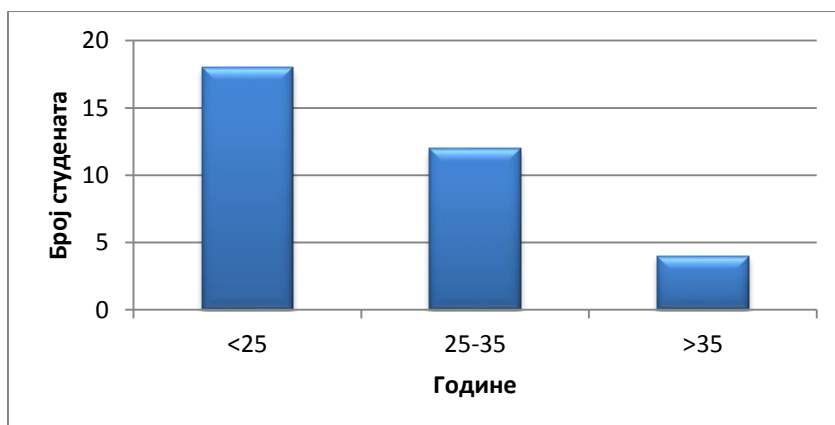
Други SDN пилот образовни модул је реализован у току зимског семестра школске 2016/17. године.

Учесници

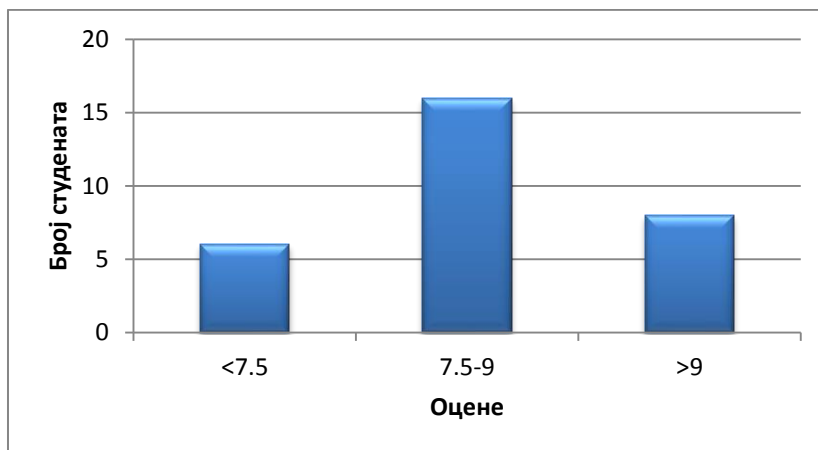
SDN образовни модул су похађали студенти смера Електронско пословање на основним, мастер, специјалистичким и докторским студијама. Пилот образовни модул је похађало 30 студената. Основни демографски подаци о студентима су приказани на сликама 53, 54, 55 и 56 [148].



Слика 53: Полна структура полазника образовног модула



Слика 54: Старосна структура полазника образовног модула



Слика 55: Оцене полазника образовног модула на досадашњим студијама



Слика 56: Професионално искуство полазника образовног модула

Подаци показују да је већина полазника млађа од 25 година и да имају до 3 године професионалног искуства у ИТ области.

Сви полазници су попунили улазну анкету у којој су исказали своје мишљење о досадашњем знању и искуству у раду са рачунарским мрежама, виртуелизацијом, *cloud computing* и SDN технологијом. Такође, требало је да процене могућност примене ових технологија у професионалном раду. Анкета је имала 11 питања са петостепеном скалом (У потпуности се слажем – 5; Слажем се – 4; Нисам сигуран

– 3; Не слажем се – 2; У потпуности се не слажем – 1). Резултати улазне анкете су приказани у табели 7 (\bar{X} – средња вредност, од 1 до 5; δ – стандардна девијација) и показују да мали број студената има одређено знање и професионално искуство у разматраним областима. Може се видети да је њихово знање више у области рачунарских мрежа, нешто ниже у области виртуелизације и *cloud computing* технологије, а најниже у области SDN технологије. Ипак, већина студената је изразила мишљење да наведене области могу наћи примену у њиховом професионалном раду.

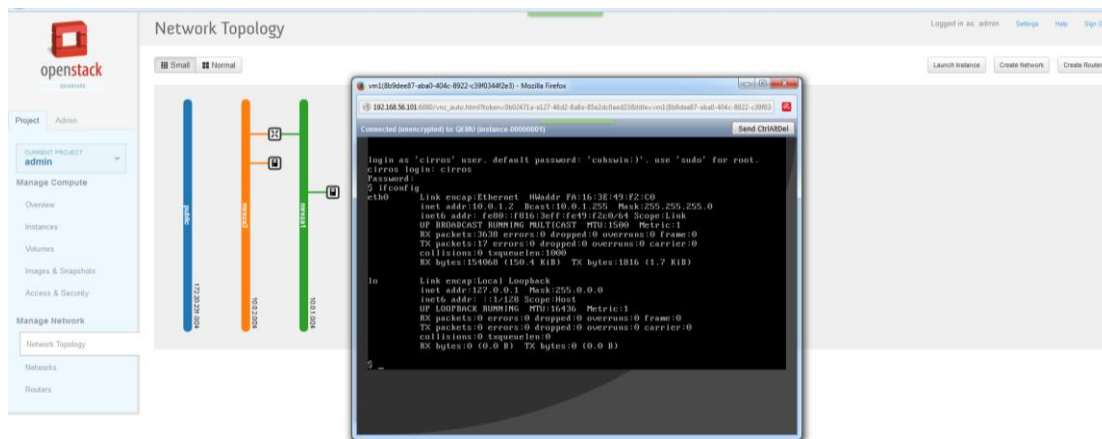
Табела 7: Резултати улазне анкете [148]

Питања	\bar{X}	δ
Процена теоријског знања		
Упознат/а сам са основним концептима традиционалних мрежа	3.50	0.86
Упознат/а сам са основним концептима виртуелних мрежа	2.83	0.83
Упознат/а сам са основним концептима <i>cloud computing</i> технологија	2.90	0.96
Упознат/а сам са основним концептима SDN технологија	2.00	0.87
Процена практичног искуства		
Имам искуства у раду са традиционалним мрежама	2.73	1.08
Имам искуства у раду са виртуелним мрежама	2.23	0.90
Имам искуства у раду са <i>cloud computing</i> технологијама	2.23	1.07
Имам искуства у раду са SDN технологијама	1.80	0.76
Процена примене у пракси		
Сматрам да ће знања стечена у области виртуелних мрежа моћи да се искористе у пракси	4.23	0.77
Сматрам да ће знања стечена у области <i>cloud computing</i> -а моћи да се искористе у пракси	4.43	0.68
Сматрам да ће знања стечена у области SDN-а моћи да се искористе у пракси	4.23	0.68

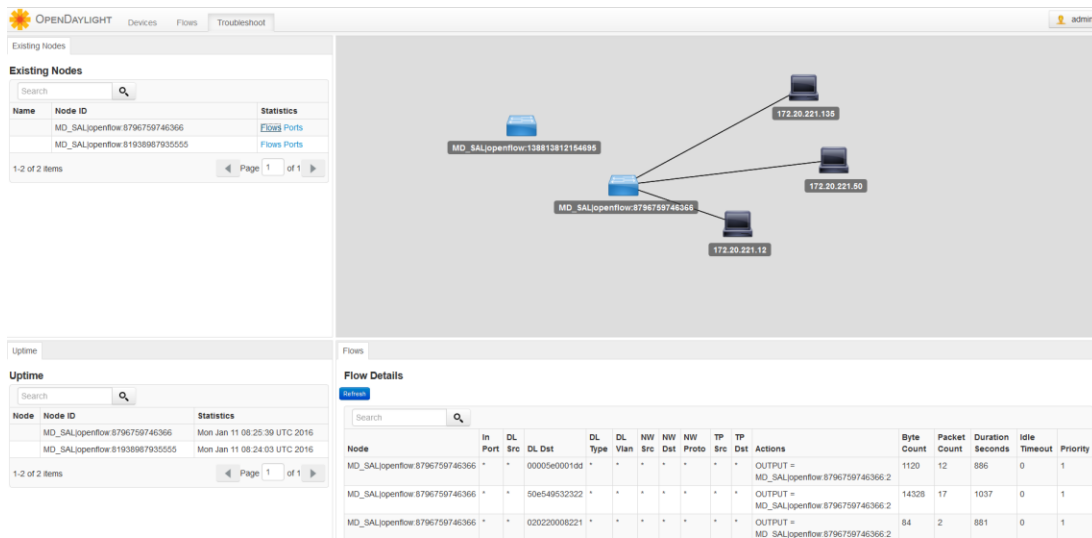
Процедура

Образовни модул се састоји из низа предавања, тестова и практичних вежби. Студенти су добили уводно теоријско знање о рачунарским мрежама, виртуелизацији, *cloud computing* и SDN технологији, као и о процесу креирања

рачунарских мрежа. У првом делу практичног задатка студенти су креирали виртуелизовано мрежно окружење коришћењем *Mininet* алата за емулацију SDN инфраструктуре. Други део практичног задатка је био заснован на креирању SDN мрежне инфраструктуре за одређену компанију на основу захтева у погледу мрежне архитектуре и подржаних сервиса. Студенти су имали на располагању преконфигурисане виртуелне машине са *OpenStack* окружењем и *OpenDaylight* контролером, као и писане инструкције за рад. Сlike 57a и 57b приказују пример реализације задатака коришћењем *OpenStack* и *OpenDaylight* решења.



(a) *OpenStack*



(b) *OpenDaylight*

Слика 57: Пример реализације задатака коришћењем *OpenStack* и *OpenDaylight* решења [148]

Инструменти

Да бисмо добили одговор на питање П1, примењен је комбиновани приступ. Коришћен је *pre-test post-test* принцип да бисмо упоредили знање студената пре и после образовног модула. Студенти су радили тест знања на почетку и на крају образовног модула. Тест се састојао из 20 питања, која су типа *multiple-choice*. Сва питања су имала исти тежински фактор. Студенти су добили оцене на скали од 5 до 10 (5 – није положио, 6 – најнижа оцена за пролаз, 10 – највиша оцена за пролаз).

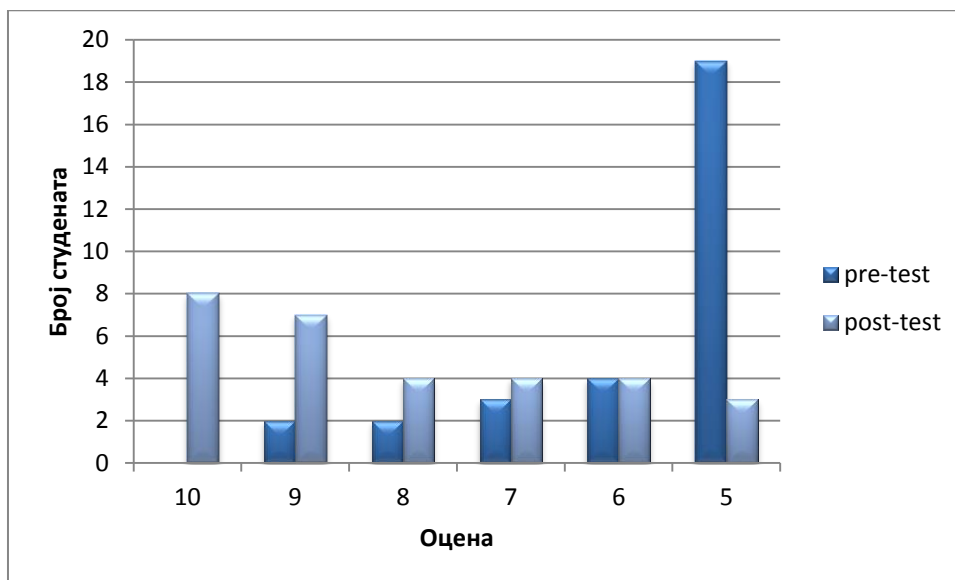
Додатно, коришћен је лонгитудинални дизајн, са два теста у току образовног модула, како би се пратио напредак студената. Први тест се односио на област виртуелизације и *cloud computing* технологије, а други на SDN. Сваки тест се састојао из 15 *multiple-choice* питања.

Да бисмо добили одговор на питање П2 и добили податке о мишљењу студената о образовном модулу, коришћен је упитник. Коришћен је измењен USE упитник. Питања су груписана у две категорије: питања која се односе на *Mininet* вежбу и питања која се односе на *OpenStack* вежбу. Свака категорија питања се састојала од четири групе питања: опште мишљење, једноставност коришћења, утицај на знање и садржај вежбе. Питања су заснована на *Likert*-овој петостепеној скали.

Анализа резултата

Просечна оцена коју су студенти добили на *pre-test*-у је 7.99 од 10, са стандардном девијацијом од 1.64 и 95% интервал поверења (7.40,8.58). Просечна оцена коју су студенти добили на *post-test*-у је 8.06 од 10 са стандардном девијацијом од 1.72 и 95% интервал поверења (7.45, 8.68). Слика 58 приказује оцене (x оса) које су студенти добили на *pre-test*-у и *post-test*-у. Y оса приказује број студената који су добили одређену оцену. Анализа питања на *pre-test*-у показује да је на питања о основним концептима рачунарских мрежа и виртуелизацији већина студената дала тачне одговоре, а да студенти нису имали претходно знање у области SDN-а. Ово

је у складу са резултатима улазне анкете који су приказани у табели 7.



Слика 58: Резултати теста знања [148]

За поређење резултата *pre-test*-а и *post-test*-а коришћен је Т-тест са зависним узорцима. Овај тест показује да је разлика у резултатима постигнутим на *pre-test*-у и *post-test*-у статистички значајна ($t(29)=-7.053$, $p<0.05$). Резултати показују да су студенти постигли значајно боље резултате на *post-test*-у у поређењу са резултатима постигнутим на *pre-test*-у.

Напредак студената је праћен током реализације образовног модула кроз два теста: један у области виртуелизације и *cloud computing* технологије, а други у области SDN-а. Просечна оцена на првом тесту је 8.97 са стандардном девијацијом 0.93, 95% интервал поверења (8.63, 9.30), а на другом тесту 7.93 са стандардном девијацијом 1.00, 95% интервал поверења (7.57, 8.29). Ово показује да су студенти лакше усвајали знања у области виртуелизације и *cloud computing* технологије у поређењу са SDN технологијом, што се може довести у везу са претходним искуством студената, као што је приказано у табели 7.

Табела 8 приказује сумарне резултате анализе упитника (\bar{X} – средња вредност, од 1 до 5; δ – стандардна девијација) који су студенти попунили на крају образовног

модула. Коришћена је петостепена скала (У потпуности се слажем – 5; Слажем се – 4; Нисам сигуран – 3; Не слажем се – 2; У потпуности се не слажем – 1).

Табела 8: Анализа резултата упитника за процену мишљења студената о образовном модулу [148]

Питања	<i>Mininet</i>		<i>OpenStack</i>	
	\bar{X}	δ	\bar{X}	δ
Питања о SDN решењу (<i>Mininet</i> и <i>OpenStack</i>)				
Решење је ефикасно	3.77	0.50	4.07	0.52
Решење се може применити у пракси	3.73	0.58	4.40	0.50
Примена решења може смањити време извршења радних задатака	3.90	0.55	4.03	0.56
Примена решења може побољшати едукацију у области рачунарских мрежа	4.30	0.60	4.47	0.57
Питања о једноставности коришћења				
Конфигурација виртуелног мрежног окружења применом решења је једноставна	3.80	0.76	3.67	0.80
Решавање задатка захтева мали број корака	3.83	0.83	3.63	0.81
Задатак се може решити без писаних инструкција	3.07	0.91	2.70	0.99
За решавање задатка је потребна помоћ асистента	3.47	0.97	3.90	1.06
Питања о утицају на знање				
Основе конфигурације решења се могу научити лако	4.07	0.69	3.57	0.82
Основе конфигурације решења се могу научити брзо	4.07	0.45	3.53	0.78
Питања о садржају вежбе				
Задатак је био занимљив	4.23	0.73	4.50	0.51
Задатак је био користан	4.33	0.80	4.60	0.50
Знања стечена при изради задатка ће моћи да се искористе у пракси	3.77	0.73	3.93	0.74

Одговори на питања која се односе на ефикасност и примену *Mininet* и *OpenStack* решења показују да студенти дају незнатну предност *OpenStack*-у, што се може објаснити чињеницом да је ово решење нашло широку примену у пракси, док се *Mininet* користи углавном у образовне сврхе. Са друге стране, када је у питању једноставност учења, *Mininet*-у је дата предност. Оба решења су добила високе оцене када је у питању занимљивост задатка и могућност примене у пракси, тако

да оба решења представљају добар избор за реализацију SDN образовног модула. Ипак, треба имати на уму чињеницу да су оба решења добила релативно ниске оцене када је у питању једноставност коришћења, тако да је неопходно креирати квалитетне наставне материјале и инструкције за вежбе као подршку за учење ових решења.

Студенти су дали мишљење да је тежина решавања задатака била на средњем нивоу и да је већини била потребна помоћ наставника за успешну реализацију задатка. Резултати евалуације показују да су студенти имали позитивно мишљење о SDN образовном модулу.

6. НАУЧНИ И СТРУЧНИ ДОПРИНОСИ

Најзначајнији допринос докторске дисертације је развој модела инфраструктуре за е-образовање заснованог на софтверски дефинисаним мрежама за примену у академским мрежама, као и развој модела образовног модула за учење софтверски дефинисаних мрежа. Финални резултат истраживања је имплементиран образовни модул за учење софтверски дефинисаних мрежа у оквиру Катедре за електронско пословање на Факултету организационих наука, Универзитета у Београду.

Кључни научни допринос дисертације огледа се у:

- Формалном опису модела инфраструктуре за е-образовање заснованог на софтверски дефинисаним мрежама.
- Развоју модела архитектуре система е-образовања заснованог на софтверски дефинисаним мрежама.
- Развоју методологије пројектовања инфраструктуре за е-образовање засноване на софтверски дефинисаним мрежама.
- Развоју модела интеграције инфраструктурних и образовних сервиса у *cloud* инфраструктуру засновану на софтверски дефинисаним мрежама.
- Развоју метода за оцену техничких и образовних перформанси развијеног модела.
- Развоју модела образовног модула за учење софтверски дефинисаних мрежа.
- Развоју методологије пројектовања образовног модула за учење софтверски дефинисаних мрежа.
- Мерењу остварених резултата примене образовног модула за учење софтверски дефинисаних мрежа у образовном процесу.

Истраживања у оквиру ове дисертације резултовала су низом стручних доприноса од којих су најважнији: анализа примене концепта софтверски дефинисаних мрежа у е-образовању, преглед и анализа технологија потребних за имплементацију модела инфраструктуре за е-образовање заснованог на

софтверски дефинисаним мрежама у *cloud computing* окружењу, анализа софтверске и хардверске инфраструктуре неопходне за имплементацију предложене мрежне инфраструктуре, анализа примене образовног модула за учење софтверски дефинисаних мрежа, преглед и анализа инфраструктуре потребне за реализацију образовног модула за учење софтверски дефинисаних мрежа и реализација образовног модула за учење софтверски дефинисаних мрежа.

Истраживање проблематике увођења концепта и технологија софтверски дефинисаних мрежа у систем е-образовања са становишта друштвене корисности може имати вишеструке импликације:

- Резултати истраживања треба да помогну у анализи могућности даљег развоја и примене модела *cloud* инфраструктуре за е-образовање заснованог на софтверски дефинисаним мрежама.
- Резултате истраживања могу користити и друге образовне институције заинтересоване за развој и прилагођавање наставних активности имплементацијом софтверски дефинисаних мрежа.
- Резултати истраживања ће допринети да се прецизније одреде потребни временски, материјални и људски ресурси за успешну имплементацију *cloud* инфраструктуре за е-образовање засноване на софтверски дефинисаним мрежама и образовног модула за учење софтверски дефинисаних мрежа.
- Резултати истраживања ће допринети да се анализира проблематика даљег развоја образовних модула у области нових мрежних технологија.
- Резултати истраживања помоћи ће у прилагођавању образовног процеса у области рачунарства и информационих технологија где је примена нових технологија од посебног значаја.

Могућности примене резултата истраживања су велике с обзиром на актуелност теме и чињеницу да постоји потреба за постојањем ефикасне и адаптивне академске мреже, као и потреба за унапређењем процеса образовања у области нових мрежних технологија. Успешна имплементација предложеног модела е-

образовања за софтверски дефинисане мреже на Факултету организационих наука представља значајан резултат практичне примене.

Резултати истраживања реализованих у оквиру ове докторске дисертације објављени су у више радова у научним часописима и саопштени на научним скуповима и то:

Радови објављени у часопису међународног значаја на *SCI* листи:

1. J. Šuh, Ž. Bojović, M. Despotović-Zrakić, Z. Bogdanović, A. Labus.: *Designing a Course and Infrastructure for Teaching Software Defined Networking*, Computer Applications in Engineering Education, Online ISSN: 1099-0542, 2017, импакт фактор за 2015= 0.935, DOI: 10.1002/cae.21820, рад прихваћен за објављивање (M22).

Радови у зборницима међународних скупова штампани у целини (M30):

2. J. Šuh, V. Vujin, Z. Bogdanović.: *Software Defined Network Infrastructure for an E-learning Cloud*, - Proceedings of the XIV International Symposium SymOrg 2014, Zlatibor, Serbia, 6-10. June 2014, pp. 357-364, ISBN 978-86-7680-295-1 (M33).
3. J. Šuh, Ž. Bojović, M. Radenković, A. Labus, Z. Bogdanović.: *Designing SDN Infrastructure for an E-learning Cloud*, - Proceedings of the 7th annual International Conference on Education and New Learning Technologies EDULEARN15, Barcelona, Spain, 6-8. July 2015, pp. 4352-4358, ISBN 978-84-606-8243-1 (M33).

Радови у часописима националног значаја (M50):

4. D. Barać, A. Milić, A. Nastevski, I. Vojinović, J. Šuh.: *Developing Services for E-Learning Courses Adaptation*, - Journal for Theory and Practice Management, Vol.

69, pp. 55-63, Beograd, 2013, ISSN 1820-0222, doi: 10.7595/management.fon.2013.0023 (M51).

5. J. Šuh, K. Simić, M. Milutinović, V. Vujić, S. Jevremović.: *Sistem za praćenje performansi mrežne infrastrukture visokoškolske ustanove*, - InfoM, Časopis za informacione tehnologije i multimedijalne sisteme, Vol. 49, pp. 28-34, Beograd, 2014, ISSN 1451-4397 (M52).
6. J. Šuh, V. Vujić, D. Barać, Z. Bogdanović, B. Radenković.: *Designing Cloud Infrastructure for Big Data in E-government*, - Journal of Universal Excellence, Faculty of Organisation Studies in Novo mesto, Slovenia, Vol. 4, No. 1, pp. A26–A38, March 2015, ISSN 2232-5204 (M50).

Радови у зборницима националних скупова штампани у целини (M60):

7. J. Šuh, A. Obradović, K. Simić.: *Komparativna analiza performansi EIGRPv6 i OSPFv3 protokola*, - Zbornik radova XXXI Simpozijum o novim tehnologijama u poštanskom i telekomunikacionom saobraćaju – PosTel 2013, Beograd, 3. i 4. decembar 2013, pp.333-342, ISBN 978-86-7395-314-4 (M63).
8. J. Šuh, V. Čulum.: *Protokoli za redudansu u IP mrežama*, – Zbornik radova XIX Naučno-stručni skup Informacione tehnologije 2014, Žabljak, Crna Gora, 24-28. februar 2014, pp. 80-83, ISBN 978-86-85775-15-4 (M63).
9. J. Šuh, B. Sisojević.: *Informaciono-komunikacioni alati za upravljanje IP/MPLS mrežom*, – Zbornik radova XX Naučno-stručni skup Informacione tehnologije 2015, Žabljak, Crna Gora, 23-27. februar 2015, pp. 88-91, ISBN 978-86-85775-16-1 (M63).
10. J. Šuh, J. Petrović.: *Primena koncepta blended učenja u korporativnom okruženju telekomunikacione kompanije*, – Zbornik radova XXI Naučno-stručni skup

Informacione tehnologije 2016, Žabljak, Crna Gora, 29. februar – 05. mart 2016, pp. 14-17, ISBN 978-86-85775-18-5 (M63).

11. J. Šuh, S. Stefanovski Pajović.: *Zaštita od DoS napada primenom Remotely Triggered Black Hole tehnike filtriranja saobraćaja*, – Zbornik radova XXII Naučno-stručni skup Informacione tehnologije 2017, Žabljak, Crna Gora, 27. februar – 04. mart 2017, pp. 277-280, ISBN:978-86-85775-20-8 (M63).

7. БУДУЋА ИСТРАЖИВАЊА

Примена нових технологија у образовним установама је увек везана за доступност ресурса, њихово рационално коришћење и ограничена финансијска средства. Универзитети немају могућност да замене постојећу мрежну инфраструктуру у циљу примене решења заснованих на новим технологијама. Имајући у виду ове чињенице, представљена је измена постојеће мрежне инфраструктуре, која ради заједно са SDN технологијом [150].

Ова докторска дисертација отвара могућности за даља истраживања која се односе на примену у различитим областима. Предложени модел мрежне инфраструктуре је веома погодан за наставу и учење у области напредних информационо-комуникационих технологија, као што је наш пилот образовни модул показао. Зато смернице развоја за будуће образовне модуле треба да иду у два правца [148]: 1) да се подигне ниво мрежне писмености међу већим бројем студената информациононих технологија и 2) да се ефикасније образују мрежни професионалци. Напредак ка овим циљевима треба да повећа брзину усвајања знања у области SDN технологије и других савремених мрежних технологија.

Међутим, захваљујући високом степену интероперабилности предложеног решења, инфраструктура заснована на SDN технологији се може применити и у другим областима образовања. Како је за креирање ресурса потребних за реализацију образовног процеса неопходно знање наставника из области информациононих технологија, треба размотрити могућност унапређења, односно развоја и имплементације алата за једноставно и ефикасно креирање наставних ресурса.

Посебан правац истраживања треба да буде усмерен ка развоју модела интеграције инфраструктуре засноване на SDN технологији различитих образовних установа. Даља примена SDN технологије треба да омогући ефикасну комуникацију интелигентних уређаја коришћењем IP протокола, комуникацију у мобилним мрежама и у оквиру дата центара [151]. Следећи корак је и примена

квалитета сервиса како би се постигао виши степен оптимизације коришћења ресурса [152]. Крајњи циљ је креирање мрежне инфраструктуре која може да се примени у различитим окружењима и обезбеди комуникацију високог квалитета и високог степена ефикасности.

8. ЗАКЉУЧАК

У оквиру ове дисертације је извршена анализа главних проблема и предложено решење за развој модела *cloud* инфраструктуре за е-образовање заснованог на софтверски дефинисаним мрежама и за развој модела е-образовања за учење софтверски дефинисаних мрежа. Анализиран је појам софтверски дефинисаних мрежа, кључне карактеристике, технологије и протоколи неопходни за примену у оквиру *cloud* инфраструктуре високошколске установе, као и образовни аспекти неопходни за реализацију образовног модула за учење софтверски дефинисаних мрежа.

Резултат истраживања је развој модела *cloud* инфраструктуре за е-образовање који ће на ефикасан и ефективан начин применити концепте и технологије софтверски дефинисаних мрежа, у циљу побољшања квалитета и ефикасности образовних процеса. Поред стандардних захтева, инфраструктура захтева виртуелизацију мреже, динамичко прилагођавање мреже [153], једноставно решавање проблема, отпорност и скалабилност [154].

У практичном делу дисертације пројектован је и имплементиран модел образовног модула за учење софтверски дефинисаних мрежа. Модел ће пружити подршку у прилагођавању образовног процеса потребама студената у реалном времену. Предложени модел је флексибилан, скалабилан и омогућава интеграцију различитих метода учења. У оквиру евалуације модела реализовано је тестирање и мерење релевантних параметара који утичу на ефикасност модела. Резултати истраживања показују да примена предложеног концепта има позитиван утицај на процес учења.

Кључне импликације ове дисертације се могу посматрати са становишта образовних институција, наставника и студената [148]. Са SDN технологијом, учење мрежних технологија се преусмерава са произвођача опреме на академска окружења, дајући академским институцијама могућност да буду иноватор у стварању нових студијских програма или нових образовних модула у оквиру

постојећих програма. Такође, SDN може бити добар избор за образовни модул у оквиру програма животног учења (енгл. *life-learning*) у области рачунарских мрежа. Треба нагласити да увођење образовних модула из области SDN-а захтева адекватну ИТ инфраструктуру, што може захтевати неке промене или чак инвестиције у академској мрежи. Зато је за образовне институције од велике важности чињеница да флексибилна ИТ инфраструктура може наћи примену и у другим ИТ образовним модулима, као на пример, *cloud computing* или *big data*, који се могу ослонити на ову врсту инфраструктуре. Имајући на уму чињеницу да се SDN технологија брзо развија, наставници су обавезни да уложе напор у припреми наставних материјала и редовно их ажурирају у складу са развојем ове технологије. Међутим, наставници имају на располагању флексибилну ИТ инфраструктуру која омогућава ефикаснију примену нових образовних модула и тиме добијају већу слободу у креирању напредних практичних задатака и вежби. За студенте је највећа предност што су у могућности да стекну знања у новим ИТ областима. Захваљујући могућности да стичу практична знања у раду са најсавременијим SDN технологијама студенти добијају знање неопходно за рад у продукционим мрежним окружењима на новим ИТ пословима, као што су SDN софтвер архитекта, SDN инжењер, итд.

9. ЛИТЕРАТУРА

- [1] B. Radenković, M. Despotović-Zrakić, Z. Bogdanović, D. Barać, and A. Labus, “E-obrazovanje,” in *Elektronsko poslovanje*, Fakultet organizacionih nauka, Beograd, 2015, pp. 237–248.
- [2] R. Buyya, J. Broberg, and A. Goscinski, *Cloud Computing: Principles and Paradigms*. John Wiley & Sons, Inc., 2010.
- [3] Q. Li, R. W. H. Lau, B. Wah, H. Ashman, E. W. C. Leung, F. Li, and V. Lee, “Emerging Internet Technologies for E-Learning,” *IEEE Internet Computing*, vol. 13, no. 4, pp. 11–17, 2009.
- [4] Open Networking Foundation, “SDN definition,” 2015. [Online]. Available: <https://www.opennetworking.org/sdn-resources/sdn-definition>. [Accessed: 15-Apr-2017].
- [5] M. Despotović-Zrakić, K. Simić, A. Labus, A. Milić, and B. Jovanić, “Scaffolding Environment for e-Learning through Cloud Computing,” *Educational Technology & Society*, vol. 16, no. 3, pp. 301–314, 2013.
- [6] R. Austin, M. Sharma, P. Moore, and D. Newell, “Situating Computing and Virtual Learning Environments: e-Learning and the Benefits to the Students Learning,” in *2013 Seventh International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*, 2013, pp. 523–528.
- [7] F. Alam, I. Katib, and A. S. Alzahrani, “New Networking Era: Software Defined Networking,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 11, pp. 349–353, 2013.
- [8] N. Sultan, “Cloud computing for education: A new dawn?,” *International Journal of Information Management*, vol. 30, no. 2, pp. 109–116, 2010.
- [9] J. Tourrilhes, P. Sharma, S. Banerjee, and J. Pettit, “The Evolution of SDN and OpenFlow : A Standards Perspective,” *IEEE Computer Magazine*, vol. 47, no. 11, pp. 22–29, 2014.
- [10] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, “A Survey on Software-Defined Networking,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 27–51, 2015.
- [11] C. E. Rothenberg, R. Chua, J. Bailey, M. Winter, C. Correa, S. C. de Lucena, M. R. Salvador, and T. Nadeau, “When Open Source Meets Network Control Planes,” *IEEE Computer Magazine*, vol. 47, no. 11, pp. 46–54, 2014.
- [12] M. G. Moore and G. Kearsley, *Distance education: a systems view*. Wadsworth Pub. Co., 1996.
- [13] K. Passerini and M. J. Granger, “A developmental model for distance learning using the Internet,” *Computers & Education*, vol. 34, no. 1, pp. 1–15, Jan. 2000.

- [14] T. Anderson and F. Elloumi, *Theory and Practice of Online Learning*. Athabasca, Canada: Athabasca University, 2004.
- [15] R. Pastore, "Elearning in Education: An Overview," in *Proceedings of Society for Information Technology & Teacher Education International Conference 2002*, 2002, pp. 275–276.
- [16] B. G. Wilson, *Constructivist Learning Environments: Case Studies in Instructional Design*. Educational Technology Publications, 1996.
- [17] N. Hoic-Bozic, V. Mornar, and I. Boticki, "A Blended Learning Approach to Course Design and Implementation," *IEEE Transactions on Education*, vol. 52, no. 1, pp. 19–30, 2009.
- [18] J. Šuh and J. Petrović, "Primena koncepta blended učenja u korporativnom okruženju telekomunikacione kompanije," in *Zbornik radova XXI Naučno-stručni skup Informacione tehnologije 2016*, 2016, pp. 14–17.
- [19] F. J. Garcia-Penalvo, M. a Conde, M. Alier, and M. J. Casany, "Opening Learning Management Systems to Personal Learning Environments," *Journal of Universal Computer Science*, vol. 17, no. 9, pp. 1222–1240, 2011.
- [20] B. Islamov and D. Islamov, "Distance Learning and Economic Education," in *7th International Conference «Central Asia – 2013: Internet, Information and Library Resources in Science, Education, Culture and Business»*, 2013.
- [21] C. Severance, "Teaching the World: Daphne Koller and Coursera," *Computer*, vol. 45, no. 8, pp. 8–9, Aug. 2012.
- [22] M. G. Urtel, "Assessing Academic Performance between Traditional and Distance Education Course Formats.," *Educational Technology & Society*, vol. 11, no. 1, pp. 322–330, 2008.
- [23] V. Vujin, "Model IT infrastrukture za e-obrazovanje - doktorska disertacija," Fakultet Organizacionih nauka, Univerzitet u Beogradu, 2012.
- [24] S. Lann, *IT Infrastructure Architecture - Infrastructure building blocks and concepts*. Lulu Press Inc, 2013.
- [25] P. Weill and M. Vitae, "What IT Infrastructure Capabilities are Needed to Implement E-Business Models?," *Mis Quarterly Executive*, vol. 1, no. 1, pp. 17–34, 2002.
- [26] E. Lazowska, P. Lee, C. Elliott, and L. Smarr, "Infrastructure for eScience and eLearning in Higher Education," *Computing Community Consortium*, 2008.
- [27] B. Pfaff, J. Pettit, T. Koponen, K. Amidon, M. Casado, and S. Shenker, "Extending Networking into the Virtualization Layer," in *8th ACM Workshop on Hot Topics in Networks*, 2009.
- [28] A. Clemm, *Network Management Fundamentals*. Cisco Press, 2006.
- [29] R. V. Srinivasa, R. N. K. Nageswara, and E. K. Kumar, "Cloud Computing: An

- overview,” *Journal of Theoretical and applied Information Technology*, vol. 9, no. 1, 2009.
- [30] L. Qian, Z. Luo, Y. Du, and L. Guo, “Cloud Computing: An Overview,” in *IEEE International Conference on Cloud Computing, CloudCom 2009*, 2009, vol. 5931, pp. 626–631.
- [31] D. L. Beauty, “Cloud Computing 101,” *ASHRAE Journal*, vol. 55, no. 10, pp. 88–93, 2013.
- [32] P. Mell and T. Grance, “The NIST definition of cloud computing - Recommendations of the National Institute of Standards and Technology,” 2011. [Online]. Available: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>. [Accessed: 15-Apr-2017].
- [33] Z. He and G. Liang, “Research and Evaluation of Network Virtualization in Cloud Computing Environment,” *Third International Conference on Networking and Distributed Computing*, pp. 40–44, 2012.
- [34] A. di Costanzo, M. D. de Assunção, and R. Buyya, “Harnessing cloud technologies for a virtualized distributed computing infrastructure,” *IEEE Internet Computing*, vol. 13, no. 5, pp. 24–33, 2009.
- [35] W. Tian, S. Su, and G. Lu, “A Framework for Implementing and Managing Platform as a Service in a Virtual Cloud Computing Lab,” in *2010 Second International Workshop on Education Technology and Computer Science*, 2010, no. 2, pp. 273–276.
- [36] H. Jin, S. Ibrahim, T. Bell, W. Gao, D. Huang, and S. Wu, “Cloud Types and Services,” in *Handbook of Cloud Computing*, 2010, pp. 335–355.
- [37] B. Radenkovic, M. Despotovic-Zratic, Z. Bogdanovic, V. Vujin, and D. Barac, “Harnessing cloud computing infrastructure for e-learning services,” *Facta universitatis - series: Electronics and Energetics*, vol. 27, no. 3, pp. 339–357, 2014.
- [38] D. L. Watson, “Hey - Get Off My Cloud!,” in *Global Security, Safety and Sustainability, 5th International Conference*, 2009, pp. 224–232.
- [39] M. D. De Assunção, A. Di Costanzo, and R. Buyya, “Evaluating the cost-benefit of using cloud computing to extend the capacity of clusters,” in *Proceedings of the 18th ACM international symposium on High performance distributed computing*, 2009, pp. 141–150.
- [40] J. Šuh, V. Vujin, D. Barać, Z. Bogdanović, and B. Radenković, “Designing Cloud Infrastructure for Big Data in E-government,” *Journal of Universal Excellence*, vol. 4, pp. A26–A38, 2015.
- [41] T. Ercan, “Effective use of cloud computing in educational institutions,” *Procedia - Social and Behavioral Sciences*, vol. 2, no. 2, pp. 938–942, 2010.

- [42] N. M. Rao and C. Sasidhar, "Cloud Computing Through Mobile-Learning," *International Journal of Advanced Computer Science and Applications*, vol. 1, no. 6, 2012.
- [43] K. Yang and X. Jia, "Data storage auditing service in cloud computing: challenges, methods and opportunities," *World Wide Web-Internet and Web Information Systems*, vol. 15, no. 4, pp. 409–428, Jul. 2012.
- [44] V. C. Emeakaroha, M. A. S. Netto, R. N. Calheiros, I. Brandic, R. Buyya, and C. A. F. De Rose, "Towards autonomic detection of SLA violations in Cloud infrastructures," *Future Generation Computer Systems*, vol. 28, no. 7, pp. 1017–1029, Jul. 2012.
- [45] V. C. Emeakaroha, R. N. Calheiros, M. A. S. Netto, I. Brandic, and C. A. F. De Rose, "DeSVi: An Architecture for Detecting SLA Violations in Cloud Computing Infrastructures," in *2nd international ICST conference on Cloud Computing (CloudComp'10)*, 2010.
- [46] J. Šuh, K. Simić, M. Milutinović, V. Vujin, and S. Jevremović, "Sistem za praćenje performansi mrežne infrastrukture visokoškolske ustanove," *InfoM, Časopis za informacione tehnologije i multimedijalne sisteme*, vol. 49, pp. 28–34, 2014.
- [47] J. Šuh and B. Sisojević, "Informaciono-komunikacioni alati za upravljanje IP/MPLS mrežom," in *Zbornik radova XX Naučno-stručni skup Informacione tehnologije 2015*, 2015, pp. 88–91.
- [48] J. Šuh and V. Čulum, "Protokoli za redudansu u IP mrežama," in *Zbornik radova XIX Naučno-stručni skup Informacione tehnologije 2014*, 2014, pp. 80–83.
- [49] Open Networking Foundation, "How OpenFlow-Based SDN Transforms Private Cloud," 2012. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-private-cloud.pdf>. [Accessed: 15-Apr-2017].
- [50] Open Networking Foundation, "SDN in the Campus Environment," 2013. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-enterprise-campus.pdf>. [Accessed: 15-Apr-2017].
- [51] M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker, "Ethane: taking control of the enterprise," in *Sigcomm '07*, 2007.
- [52] N. Feamster, J. Rexford, and E. Zegura, "The Road to SDN: An Intellectual History of Programmable Networks," *ACM Sigcomm Computer Communication*, vol. 44, no. 2, pp. 87–98, 2014.
- [53] Open Networking Foundation, "Software-Defined Networking: The New Norm for Networks," 2012. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf>. [Accessed: 15-Apr-2017].

- [54] B. Heller, J. McCauley, K. Zarifis, P. Kazemian, C. Scott, N. McKeown, S. Shenker, A. Wundsam, H. Zeng, S. Whitlock, V. Jeyakumar, and N. Handigol, "Leveraging SDN layering to systematically troubleshoot networks," *ACM SIGCOMM Hot Topics in SDN*, p. 37, 2013.
- [55] R. V. Nunes, R. L. Pontes, and D. Guedes, "Virtualized network isolation using Software Defined Networks," *Proceedings - Conference on Local Computer Networks, LCN*, pp. 683–686, 2013.
- [56] T. Nadeau and P. Pan, "Software Driven Networks Problem Statement - Internet-Draft," 2012. [Online]. Available: <https://tools.ietf.org/html/draft-nadeau-sdn-problem-statement-01>. [Accessed: 15-Apr-2017].
- [57] D. Kreutz, F. M. V. Ramos, and P. Verissimo, "Towards secure and dependable software-defined networks," *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking - HotSDN '13*, pp. 55–60, 2013.
- [58] H. Xie, T. Tsou, D. Lopez, R. Sidi, H. Yin, and P. A. Aranda, "Software-defined Networking Efforts Debuted at IETF 84," *IETF Journal*, vol. 8, no. October, pp. 12–16, 2012.
- [59] H. Yin, H. Xie, T. Tsou, D. Lopez, P. Aranda, and R. Sidi, "SDNi: A Message Exchange Protocol for Software Defined Networks (SDNS) across Multiple Domains - Internet Draft," 2012. [Online]. Available: <https://tools.ietf.org/pdf/draft-yin-sdn-sdni-00.pdf>.
- [60] A. Dixi, F. Hao, S. Mukherjee, T. V. Lakshnan, and R. Kompella, "Towards an elastic distributed sdn controller," *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking - HotSDN '13*, pp. 7–12, 2013.
- [61] A. Gelberger, N. Yemini, and R. Giladi, "Performance analysis of Software-Defined Networking (SDN)," *Proceedings - IEEE Computer Society's Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, MASCOTS*, pp. 389–393, 2013.
- [62] M. Jammal, T. Singh, A. Shami, R. Asal, and Y. Li, "Software defined networking: State of the art and research challenges," *Computer Networks*, vol. 72, pp. 74–98, 2014.
- [63] "OpenFlow." [Online]. Available: <https://www.opennetworking.org/sdn-resources/openflow>.
- [64] E. Haleplidis, K. Pentikousis, S. Denazis, J. H. Salim, D. Meyer, and O. Koufopavlou, "RFC 7426: Software-Defined Networking (SDN): Layers and Architecture Terminology," 2015. [Online]. Available: <https://tools.ietf.org/html/rfc7426>. [Accessed: 15-Apr-2017].
- [65] J. Pettit, J. Gros, B. Pfaff, M. Casado, and S. Crosby, "Virtual Switching in an Era of Advanced Edges," in *2nd Workshop on Data Center - Converged and*

Virtual Ethernet Switching (DC CAVES), 2010, pp. 347–358.

- [66] Y. Li and G. Wang, “SDN-Based Switch Implementation on Network Processors,” *Communications and Networks*, vol. 5, no. September, pp. 434–437, 2013.
- [67] H. Kim and N. Feamster, “Improving network management with software defined networking,” *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, 2013.
- [68] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, “NOX: towards an operating system for networks,” *SIGCOMM Computer Communication Review*, vol. 38, no. 3, pp. 105–110, 2008.
- [69] L. R. Prete, C. M. Schweitzer, A. A. Shinoda, and R. L. S. de Oliveira, “Simulation in an SDN network scenario using the POX Controller,” in *2014 IEEE Colombian Conference on Communications and Computing (COLCOM)*, 2014.
- [70] D. Erickson, “The beacon openflow controller,” in *Proceedings of the second workshop on Hot topics in software defined networks HotSDN '13*, 2013, pp. 13–18.
- [71] “Floodlight.” [Online]. Available: <http://www.projectfloodlight.org/>. [Accessed: 15-Apr-2017].
- [72] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker, “Onix: A distributed control platform for large-scale production networks,” in *Proceedings of the 9th USENIX conference on Operating systems design and implementation OSDI'10*, 2010, pp. 1--6.
- [73] A. Tootoonchian and Y. Ganjali, “Hyperflow: a distributed control plane for openflow,” in *Proceedings of the 2010 internet network management conference on Research on enterprise networking INM/WREN'10*, 2010.
- [74] “HP SDN Controller Architecture,” *Technical Solution Guide Hewlett-Packard Development Company, L.P.*, 2013. [Online]. Available: http://h17007.www1.hp.com/docs/networking/solutions/sdn/devcenter/06_-_HP_SDN_Controller_Architecture_TSG_v1_3013-10-01.pdf. [Accessed: 15-Apr-2017].
- [75] “OpenDaylight, Project web site,” 2015. [Online]. Available: <http://www.opendaylight.org>. [Accessed: 15-Apr-2017].
- [76] “OpenStack and OpenDaylight.” [Online]. Available: https://wiki.opendaylight.org/view/OpenStack_and_OpenDaylight. [Accessed: 15-Apr-2017].
- [77] N. Mckeown, T. Anderson, H. Balakrishnan, G. M. Parulkar, L. L. Peterson, J. Rexford, S. Shenker, J. S. Turner, and S. Louis, “OpenFlow: enabling innovation in campus networks,” *ACM SIGCOMM Computer Communication Review*, vol.

38, no. 2, pp. 69–74, 2008.

- [78] Open Networking Foundation, “OpenFlow Switch Specification Version 1.5.1,” 2015. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-switch-v1.5.1.pdf>. [Accessed: 15-Apr-2017].
- [79] M. Kobayashi, S. Seetharaman, G. Parulkar, G. Appenzeller, J. Little, J. Van Reijendam, P. Weissmann, and N. McKeown, “Maturing of OpenFlow and Software-defined Networking through deployments,” *Computer Networks*, vol. 61, pp. 151–175, 2014.
- [80] M. H. Raza, S. C. Sivakumar, A. Nafarieh, and B. Robertson, “A comparison of software defined network (SDN) implementation strategies,” *Procedia Computer Science*, vol. 32, pp. 1050–1055, 2014.
- [81] B. Pfaff and B. Davie, “RFC 7047: The Open vSwitch Database Management Protocol,” 2013. [Online]. Available: <https://tools.ietf.org/html/rfc7047>. [Accessed: 15-Apr-2017].
- [82] B. N. Astuto, M. Mendon, B. N. Astuto, M. Mendon, B. A. A. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turlitti, “A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks,” 2014.
- [83] “Open vSwitch.” [Online]. Available: <http://openvswitch.org/>. [Accessed: 15-Apr-2017].
- [84] “Open vSwitch Features.” [Online]. Available: <http://openvswitch.org/features/>. [Accessed: 15-Apr-2017].
- [85] B. Pfaff, J. Pettit, T. Koponen, E. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar, K. Amidon, and M. Casado, “The Design and Implementation of Open vSwitch,” in *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI '15)*, 2015, pp. 117–130.
- [86] L. MacVittie, “Reactive, Proactive, Predictive: SDN Models,” 2013. [Online]. Available: <https://devcentral.f5.com/articles/reactive-proactive-predictive-sdn-models>. [Accessed: 15-Apr-2017].
- [87] “VXLAN Overview: Cisco Nexus 9000 Series Switches,” 2015. [Online]. Available: <http://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-729383.pdf>. [Accessed: 15-Apr-2017].
- [88] “Quagga.” [Online]. Available: <http://www.nongnu.org/quagga/>. [Accessed: 15-Apr-2017].
- [89] “Mikrotik.” [Online]. Available: <http://www.mikrotik.com/>. [Accessed: 15-Apr-2017].
- [90] R. Sherwood, G. Gibb, K. Yap, G. Appenzeller, M. Casado, N. Mckeown, and G. Parulkar, “FlowVisor: A Network Virtualization Layer,” 2009. [Online].

- Available: <http://archive.openflow.org/downloads/technicalreports/openflow-tr-2009-1-flowvisor.pdf>. [Accessed: 15-Apr-2017].
- [91] “OpenDaylight Developer Wiki.” [Online]. Available: https://wiki.opendaylight.org/view/Main_Page. [Accessed: 15-Apr-2017].
- [92] “Cisco Open Network Environment (ONE).” [Online]. Available: <http://www.cisco.com/c/en/us/products/software/one-software/index.html>. [Accessed: 15-Apr-2017].
- [93] “Cisco Open SDN Controller 1.0 Data Sheet.” [Online]. Available: <http://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/open-sdn-controller/datasheet-c78-733458.pdf>. [Accessed: 15-Apr-2017].
- [94] A. Singla and B. Rijsman, “OpenContrail Architecture Document.” [Online]. Available: <http://opencontrail.org/opencontrail-architecture-documentation/>. [Accessed: 15-Apr-2017].
- [95] “Cisco Application Centric Infrastructure.” [Online]. Available: <http://www.cisco.com/c/en/us/solutions/data-center-virtualization/application-centric-infrastructure/index.html>. [Accessed: 15-Apr-2017].
- [96] “OpFlex: An Open Policy Protocol,” 2014. [Online]. Available: <http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-731302.html>. [Accessed: 15-Apr-2017].
- [97] “Cisco Application Virtual Switch for Cisco Application Centric Infrastructure - Data Sheet,” 2016. [Online]. Available: <http://www.cisco.com/c/en/us/products/collateral/switches/application-virtual-switch/datasheet-c78-733605.pdf>. [Accessed: 15-Apr-2017].
- [98] “Cisco Application Centric Infrastructure: Use ACI as a Technology-Based Catalyst for IT Transformation,” 2015. [Online]. Available: <http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-734501.pdf>. [Accessed: 15-Apr-2017].
- [99] Open Networking Foundation, “SDN Security Considerations in the Data Center,” 2013. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-security-data-center.pdf>. [Accessed: 15-Apr-2017].
- [100] Open Networking Foundation, “Principles and Practices for Securing Software - Defined Networks,” 2015. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/Principles_and_Practices_for_Securing_Software-Defined_Networks_applied_to_OFv1.3.4_V1.0.pdf. [Accessed: 15-Apr-2017].
- [101] D. Mattos, L. Ferraz, and O. Duarte, “AuthFlow: Authentication and Access

- Control Mechanism for Software Defined Networking,” in *XXXII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos - SBRC'2014*, 2014.
- [102] W. Odom and M. J. Cavanaugh, *Cisco QOS Exam Certification Guide*. Cisco Press, 2004.
- [103] Open Networking Foundation, “Operator Network Monetization Through OpenFlow-Enabled SDN,” 2013. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-network-monetization.pdf>. [Accessed: 15-Apr-2017].
- [104] Open Networking Foundation, “SDN Migration Considerations and Use Cases,” 2014. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-sdn-migration-use-cases.pdf>. [Accessed: 15-Apr-2017].
- [105] Open Networking Foundation, “Migration Tools and Metrics,” 2014. [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/migration-tools-and-metrics.pdf>. [Accessed: 15-Apr-2017].
- [106] B. Lantz, B. Heller, and N. McKeown, “A network in a laptop: rapid prototyping for software-defined networks,” in *Workshop on Hot Topics in Networks-Hotnets '10*, 2010, pp. 1–6.
- [107] “Mininet.” [Online]. Available: <http://mininet.org/>. [Accessed: 15-Apr-2017].
- [108] “Mininet Apps.” [Online]. Available: <https://github.com/mininet/mininet/wiki/Mininet-Apps>. [Accessed: 15-Apr-2017].
- [109] B. Heller, “Reproducible Network Research with High-Fidelity Emulation (Ph.D Thesis),” Stanford University, 2013.
- [110] A. Lara, “Using Software-Defined Networking to Improve Campus, Transport and Future Internet Architectures (Ph.D. thesis),” University of Nebraska - Lincoln, 2015.
- [111] R. Tu, X. Wang, and Y. Yang, “Energy-saving model for SDN data centers,” *The Journal of Supercomputing*, vol. 70, no. 3, pp. 1477–1495, 2014.
- [112] S. Das, A. R. Sharafat, G. Parulkar, and N. Mckeown, “MPLS with a Simple OPEN Control Plane,” in *Optical Fiber Communication Conference and Exposition (OFC/NFOEC), 2011 and the National Fiber Optic Engineers Conference*, 2011, pp. 1–3.
- [113] F. Dürr, “Towards Cloud-assisted Software-defined Networking,” *Technical Report 2012/04, Institute of Parallel and Distributed Systems, Universität Stuttgart*, p. 19, 2012.

- [114] M. Y. Luo and J. Y. Chen, "Software defined networking across distributed datacenters over cloud," in *2013 IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom)*, 2013, vol. 1, pp. 615–622.
- [115] S. Baucke, R. Ben Ali, J. Kempf, R. Mishra, F. Ferioli, and A. Carossino, "Cloud Atlas: A Software Defined Networking Abstraction for Cloud to WAN Virtual Networking," in *2013 IEEE Sixth International Conference on Cloud Computing*, 2013, pp. 895–902.
- [116] "Open Network Operating System (ONOS)." [Online]. Available: <http://onosproject.org/>. [Accessed: 15-Apr-2017].
- [117] J. Šuh and S. Stefanovski Pajović, "Zaštita od DoS napada primenom Remotely Triggered Black Hole tehnike filtriranja saobraćaja," in *Zbornik radova XXII Naučno-stručni skup Informacione tehnologije 2017*, 2017, pp. 277–280.
- [118] S. M. Park, S. Ju, and J. Lee, "Efficient Routing for Traffic Offloading in Software-defined Network," *Procedia Computer Science*, vol. 34, pp. 674–679, 2014.
- [119] J. Šuh, A. Obradović, and K. Simić, "Komparativna analiza performansi EIGRPv6 i OSPFv3 protokola," in *Zbornik radova XXXI Simpozijum o novim tehnologijama u poštanskom i telekomunikacionom saobraćaju – PosTel 2013*, 2013, pp. 333–342.
- [120] J. Šuh, Ž. Bojović, M. Radenković, A. Labus, and Z. Bogdanović, "Designing SDN Infrastructure for an E-learning Cloud," in *Proceedings of the 7th annual International Conference on Education and New Learning Technologies EDULEARN15*, 2015, pp. 4352–4358.
- [121] D. Barać, A. Milić, A. Nasteovski, I. Vojinović, and J. Šuh, "Developing Services for E-Learning Courses Adaptation," *Journal for Theory and Practice Management*, vol. 69, pp. 55–63, 2013.
- [122] J. Šuh, V. Vujin, and Z. Bogdanović, "Software Defined Network Infrastructure for an E-learning Cloud," in *Proceedings of the XIV International Symposium SymOrg 2014*, 2014, pp. 357–364.
- [123] Y. J. Wu, J. X. Liang, H. Zhang, Z. W. Lin, Y. Ma, and Z. Tian, "Programmable virtual network instantiation in IaaS cloud based on SDN," *Journal of China Universities of Posts and Telecommunications*, vol. 20, no. August, pp. 121–125, 2013.
- [124] Y. Zhao, Y. Li, I. Raicu, S. Lu, W. Tian, and H. Liu, "Enabling scalable scientific workflow management in the Cloud," *Future Generation Computer Systems*, vol. 46, pp. 3–16, 2015.
- [125] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks," in *Workshop on Hot Topics in Networks-Hotnets '10*, 2010, pp. 1–6.
- [126] M. Pueblas, B. Cox, S. Tenneneti, S. Gyurindak, J. Strika, B. McMurdo, and R.

- Kachalia, *Cisco Service Ready Architecture for Schools Design Guide*. Cisco Systems, 2010.
- [127] N. Sousa, G. R. Alves, and M. G. Gericota, “An Integrated Reusable Remote Laboratory to Complement Electronics Teaching,” *IEEE Transactions on Learning Technologies*, vol. 3, no. 3, pp. 265–271, 2010.
- [128] J. P. Lavery, D. Wood, and J. Turchek, “Software Defined Networking (SDN) Network Virtualization for the IS Curriculum?,” in *Proceedings of the Information Systems Educators Conference 2014*, 2014, pp. 1–11.
- [129] M. Pizzonia and M. Rimondini, “Netkit: network emulation for education,” *Journal of Software: Practice and Experience*, vol. 46, no. 2, pp. 133–165, 2016.
- [130] D. K. Viswanath, S. Kusuma, and S. K. Gupta, “Cloud Computing Issues and Benefits Modern Education,” *Global Journal of Computer Science and Technology Cloud & Distributed*, vol. 12, no. 10, pp. 15–19, 2012.
- [131] R. Bauer, R. Milford, and L. Zhen, “Aligning Technology and Market Drivers in an Open Source Standards Testing Program,” *IEEE Computer Magazine*, vol. 47, no. 11, pp. 30–36, 2014.
- [132] M. Yu, J. Rexford, X. Sun, S. Rao, and N. Feamster, “A survey of virtual LAN usage in campus networks,” *IEEE Communications Magazine*, vol. 49, no. 7, pp. 98–103, 2011.
- [133] A. Corradi, M. Fanelli, and L. Foschini, “VM consolidation: A real case based on OpenStack Cloud,” *Future Generation Computer Systems*, vol. 32, no. 118–127, p. Futur. Gener. Comput. Syst, 2014.
- [134] “Neutron’s Developer Documentation.” [Online]. Available: <https://docs.openstack.org/developer/neutron/>. [Accessed: 15-Apr-2017].
- [135] M. Casado, T. Koponen, S. Shenker, and A. Tootoonchian, “Fabric: a Retrospective on Evolving SDN,” in *HotSDN*, 2012.
- [136] M. Canini, A. Feldmann, D. Levin, F. Schaffert, and S. Schmid, “Software-Defined Networks: Incremental Deployment with Panopticon,” *IEEE Computer Magazine*, vol. 47, no. 11, pp. 56–60, 2014.
- [137] “SDN-NFV Reference Architecture,” *Verizon Network Infrastructure Planning*, 2016. [Online]. Available: http://innovation.verizon.com/content/dam/vic/PDF/Verizon_SDN-NFV_Reference_Architecture.pdf. [Accessed: 15-Apr-2017].
- [138] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, “RFC 6241: Network Configuration Protocol (NETCONF),” 2011. [Online]. Available: <https://tools.ietf.org/html/rfc6241>. [Accessed: 15-Apr-2017].
- [139] H. Gredler, J. Medved, S. Previdi, A. Farrel, and S. Ray, “RFC 7752: North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP,” 2016. [Online]. Available: <https://tools.ietf.org/html/rfc7752>.

[Accessed: 15-Apr-2017].

- [140] “Moodle.” [Online]. Available: <https://moodle.org/>. [Accessed: 15-Apr-2017].
- [141] A. M. Lund, “Measuring usability with the USE questionnaire,” *STC Usability SIG Newsletter*, vol. 2, no. 8, 2001.
- [142] N. I. Sarkar, “Teaching computer networking fundamentals using practical laboratory exercises,” *IEEE Transactions on Education*, vol. 49, no. 2, pp. 285–291, 2006.
- [143] C. J. Sher-Decusatis and C. Decusatis, “Developing a Software Defined Networking curriculum through industry partnerships,” *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education - “Engineering Education: Industry Involvement and Interdisciplinary Trends”*, *ASEE Zone 1 2014*, 2014.
- [144] L. C. Wilcox and M. S. Wilcox, “A Review and Evaluation of Engineering Education in Transition,” in *2010 IEEE Transforming Engineering Education: Creating Interdisciplinary Skills for Complex Global Environments*, 2010, pp. 1–24.
- [145] N. Yalcin, Y. Altun, and U. Kose, “Educational material development model for teaching computer network and system management,” *Computer Applications in Engineering Education*, vol. 23, no. 4, pp. 621–629, 2015.
- [146] S. Zappe, R. Leicht, J. Messner, T. Litzginer, and H. W. Lee, “‘Flipping’ the classroom to explore active learning in a large undergraduate course,” *American Society for Engineering Education*, 2009.
- [147] Y. Ozturk, “Network in a Box: Facilitating problem-based learning through network emulation,” *Computer Applications in Engineering Education*, vol. 19, no. 3, pp. 433–446, 2011.
- [148] J. Šuh, Ž. Bojović, M. Despotović-Zrakić, Z. Bogdanović, and A. Labus, “Designing a Course and Infrastructure for Teaching Software Defined Networking,” *Computer Applications in Engineering Education*, 2017, DOI: 10.1002/cae.21820.
- [149] “OpenStack, Project web site,” 2015. [Online]. Available: <https://www.openstack.org/>. [Accessed: 15-Apr-2017].
- [150] P. Polezhaev, A. Shukman, and A. Konnov, “Development of educational resource datacenters based on software defined networks,” in *2014 International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC)*, 2014, pp. 1–7.
- [151] J. Rak, M. Pickavet, K. S. Trivedi, J. A. Lopez, A. M. C. A. Koster, J. P. G. Sterbenz, E. K. Çetinkaya, T. Gomes, M. Gunkel, K. Walkowiak, and D. Staessens, “Future research directions in design of reliable communication systems,” *Telecommunication Systems*, vol. 60, no. 4, pp. 423–450, 2015.

- [152] K. Govindarajan, K. C. Meng, H. Ong, W. M. Tat, S. Sivanand, and L. S. Leong, "Realizing the Quality of Service (QoS) in Software-Defined Networking (SDN) based Cloud infrastructure," in *2014 2nd International Conference on Information and Communication Technology (ICoICT)*, 2014, pp. 505–510.
- [153] O. Schuermann, "Migrating to the Software-Defined Networking in Higher Education to Improve Student Outcomes," 2014. [Online]. Available: <http://themodernnetwork.com/education/migrating-software-defined-networking-higher-education-improve-student-outcomes/>. [Accessed: 15-Apr-2017].
- [154] S. H. Yeganeh, A. Tootoonchian, and Y. Ganjali, "On scalability of software-defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 136–141, 2013.

10. СПИСАК СЛИКА

Слика 1: Компоненте ИТ инфраструктуре.....	13
Слика 2: Традиционална кампус мрежна инфраструктура	17
Слика 3: Концепт софтверски дефинисаних мрежа.....	20
Слика 4: Архитектура софтверски дефинисаних мрежа [4].....	21
Слика 5: Компоненте <i>OpenFlow</i> свича [78].....	26
Слика 6: Прослеђивање саобраћаја код <i>OpenFlow</i> свича	26
Слика 7: Ток података кроз <i>OpenFlow</i> свич [78].....	28
Слика 8: <i>Open vSwitch</i> архитектура [27].....	30
Слика 9: <i>Open vSwitch</i> модел прослеђивања саобраћаја [85]	31
Слика 10: Формат VXLAN фрејма.....	34
Слика 11: <i>OpenDaylight</i> архитектура [75]	35
Слика 12: <i>Cisco Open SDN</i> платформа [93].....	37
Слика 13: <i>OpenContrail</i> архитектура [94].....	38
Слика 14: ACI архитектура [95]	39
Слика 15: Реализација аутоматског карантина за <i>malware</i> применом софтверски дефинисаних мрежа [99]	43
Слика 16: <i>AuthFlow</i> архитектура [101]	44
Слика 17: QoS сервис у SDN окружењу [103].....	46
Слика 18: <i>Greenfield</i> миграција	48
Слика 19: Мешовита миграција	48
Слика 20: Хибридна миграција	48
Слика 21: Процес прикупљања метрика [105].....	49
Слика 22: <i>Mininet</i> архитектура	50
Слика 23: Пример мрежне топологије креиране у <i>Mininet</i> -у	51
Слика 24: <i>Miniedit</i>	51

Слика 25: Реализација сервиса <i>bandwidth</i> на захтев у SDN окружењу [103]	54
Слика 26: Реализација <i>managed</i> сервиса у SDN окружењу [103]	55
Слика 27: SDN инфраструктура у академском окружењу [50].....	56
Слика 28: Компоненте модела инфраструктуре за е-образовање	62
Слика 29: Детаљна структура модела.....	63
Слика 30: Логички модел архитектуре за е-образовање.....	64
Слика 31: <i>Multi-tenancy</i> концепт [122].....	65
Слика 32: Концептуални модел <i>cloud</i> инфраструктуре [37].....	66
Слика 33: Пример физичке реализације <i>cloud</i> инфраструктуре	67
Слика 34: Хибридни <i>dual-stack</i> модел инфраструктуре	68
Слика 35: Приступни SDN модел инфраструктуре.....	68
Слика 36: <i>Panopticon</i> модел инфраструктуре [136]	69
Слика 37: Концептуални модел SDN инфраструктуре	70
Слика 38: SDN контролери [137]	71
Слика 39: WAN SDN контролер	72
Слика 40: Сервиси е-образовања	73
Слика 41: Интеграција компоненти инфраструктуре за е-образовање	76
Слика 42: Управљање инфраструктуром за е-образовање	77
Слика 43: Евалуација предложеног модела	78
Слика 44: Модел евалуације инфраструктуре за е-образовање	79
Слика 45: Фазе у креирању SDN образовног модула [148].....	84
Слика 46: Концептуални модел инфраструктуре заснован на <i>OpenStack</i> и <i>OpenDaylight</i> решењу [148]	88
Слика 47: Логички дијаграм <i>OpenDaylight</i> и <i>OpenStack</i> интеграције [148].....	89
Слика 48: Процес реализације наставе.....	90
Слика 49: Процес креирања инфраструктуре за SDN образовни модул [148]	91

Слика 50: Полна структура полазника образовног модула	93
Слика 51: Пример мрежне топологије за завршни задатак	94
Слика 52: Резултати теста знања.....	95
Слика 53: Полна структура полазника образовног модула	97
Слика 54: Старосна структура полазника образовног модула	97
Слика 55: Оцене полазника образовног модула на досадашњим студијама	98
Слика 56: Професионално искуство полазника образовног модула	98
Слика 57: Пример реализације задатака коришћењем <i>OpenStack</i> и <i>OpenDaylight</i> решења [148]	100
Слика 58: Резултати теста знања [148]	102

11. СПИСАК ТАБЕЛА

Табела 1: Карактеристике инфраструктуре за е-образовање.....	59
Табела 2: Компоненте модела SDN инфраструктуре.....	70
Табела 3: Техничко-технолошки KPI параметри инфраструктуре за е-образовање	80
Табела 4: Образовни KPI параметри инфраструктуре за е-образовање.....	81
Табела 5: Структура SDN образовног модула [148].....	86
Табела 6: Анализа резултата упитника.....	96
Табела 7: Резултати улазне анкете [148].....	99
Табела 8: Анализа резултата упитника за процену мишљења студената о образовном модулу [148]	103

БИОГРАФИЈА АУТОРА

Јелена Шух је рођена 14.10.1983. године у Земуну. Основну школу и гимназију завршила је у Земуну са одличним успехом, као носилац Вукове дипломе, а гимназију и као Ученик генерације. Електротехнички факултет Универзитета у Београду уписала је школске 2002/03. године. Дипломирала је на Одсеку за електронику, телекомуникације и аутоматику – Смер за телекомуникације у јулу 2007. године, са просечном оценом 8.86. Дипломски рад под насловом „Технике рутирања у сензорским мрежама“ одбранила је са оценом 10 (десет). Специјалистичке академске студије, студијски програм Организационе науке – модул Електронско пословање, уписала је на Факултету организационих наука у Београду 2012. године и завршила са просечном оценом 10 (десет). Завршни (Специјалистички академски) рад под насловом „Управљање мрежном инфраструктуром и сервисима у академском окружењу“ одбранила је у августу 2013. године са оценом 10 (десет). Докторске студије, студијски програм Информациони системи и менаџмент, изборно подручје Електронско пословање, уписала је на Факултету организационих наука 2013. године. Положила је свих девет, програмом предвиђених, испита на докторским студијама са просечном оценом 10 (десет). Од фебруара 2008. године запослена је у компанији Телеком Србија а.д., где је стекла искуство у области експлоатације, одржавања и надзора IP/MPLS (*Internet Protocol/Multiprotocol Label Switching*) мреже. Тренутно је запослена у Сектору за надзор и контролу квалитета мреже (NOC) на позицији Инжењер за сервисну подршку на системима у транспортној и IP/MPLS мрежи. Од 2010. године ангажована је на локалној Cisco академији, *Telekom Educational Initiative* као Cisco CCNA (*Cisco Certified Network Associate*) инструктор за извођење предавања и вежби. Поседује следеће стручне сертификате:

- CCNA (*Cisco Certified Network Associate*)
- CCNA Security (*Cisco Certified Network Associate Security*)
- CCNP (*Cisco Certified Network Professional*)
- CCNP SP (*Cisco Certified Network Professional Service Provider*)
- CCAI (*Cisco Certified Academy Instructor*)
- JNCIA (*Juniper Networks Certified Internet Associate*)

Прилог 1.

Изјава о ауторству

Име и презиме аутора **Јелена Шух**

Број индекса 5020/2013

Изјављујем

да је докторска дисертација под насловом

ИНФРАСТРУКТУРА ЗА Е-ОБРАЗОВАЊЕ ЗАСНОВАНА НА СОФТВЕРСКИ ДЕФИНИСАНИМ МРЕЖАМА

- резултат сопственог истраживачког рада;
- да дисертација у целини ни у деловима није била предложена за стицање друге дипломе према студијским програмима других високошколских установа;
- да су резултати коректно наведени и
- да нисам кршио/ла ауторска права и користио/ла интелектуалну својину других лица.

Потпис аутора

У Београду, _____

Прилог 2.

**Изјава о истоветности штампане и електронске верзије
докторског рада**

Име и презиме аутора **Јелена Шух**

Број индекса 5020/2013

Студијски програм Информациони системи и менаџмент

Наслов рада **ИНФРАСТРУКТУРА ЗА Е-ОБРАЗОВАЊЕ ЗАСНОВАНА НА
СОФТВЕРСКИ ДЕФИНИСАНИМ МРЕЖАМА**

Ментор Проф. др Божидар Раденковић

Изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао/ла ради похрањена у **Дигиталном репозиторијуму Универзитета у Београду**.

Дозвољавам да се објаве моји лични подаци везани за добијање академског назива доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

Потпис аутора

У Београду, _____

Прилог 3.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

ИНФРАСТРУКТУРА ЗА Е-ОБРАЗОВАЊЕ ЗАСНОВАНА НА СОФТВЕРСКИ ДЕФИНИСАНИМ МРЕЖАМА

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигиталном репозиторијуму Универзитета у Београду и доступну у отвореном приступу могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио/ла.

1. Ауторство (CC BY)
2. Ауторство – некомерцијално (CC BY-NC)
- 3. Ауторство – некомерцијално – без прерада (CC BY-NC-ND)**
4. Ауторство – некомерцијално – делити под истим условима (CC BY-NC-SA)
5. Ауторство – без прерада (CC BY-ND)
6. Ауторство – делити под истим условима (CC BY-SA)

(Молимо да заокружите само једну од шест понуђених лиценци.
Кратак опис лиценци је саставни део ове изјаве).

Потпис аутора

У Београду, _____

1. **Ауторство.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.
2. **Ауторство – некомерцијално.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.
3. **Ауторство – некомерцијално – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.
4. **Ауторство – некомерцијално – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.
5. **Ауторство – без прерада.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.
6. **Ауторство – делити под истим условима.** Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.