Нов български университет

# Проектиране на корпоративни мрежи

# Част II
# Оптимизиране на Слой 2

**Доц. Д-р Емил Стоилов**

**Департамент по Информатика на НБУ**

**София, май 2014**

# Съдържание

Сложността, присъща на мрежата във фирмения комплекс на едно предприятие, изисква процес на проектиране, който позволява разделянето на цялостното решение на отделни, основни елементи. Под фирмен комплекс (campus) се разбира комплекса от фирмени сгради, намиращи се близко една до друга, заедно с прилежащите към тях терени. Йерархичния модел, разработен от Cisco Systems Inc., постига тази цел чрез разделяне на мрежовата архитектура на отделни, модулни компоненти [1]. В рамките на йерархията на фирмения комплекс, всеки модул представлява функционален слой на обслужване.

Този доклад е разделен на две части. В първата част са разгледани въпросите свързани с проектирането на компютърната мрежа във фирмения комплекс на предприятието, към която мрежа имаме изисквания за висока достъпност. Втората част е посветена на проектирането на оптималния дизайн на слой 2 на тази мрежа.

## 1. Проектиране на високо достъпна мрежа на фирмения комплекс

Възприетият йерархичен модел позволява проектирането на такава модулна топология на мрежата, че винаги да имаме достъп до тази мрежа, т.е. мрежата да бъде високо достъпна. Използването на мащабируеми градивни елементи подкрепя бързото развитие на бизнес нуждите. Модулният подход прави мрежата по-лесно мащабируема, разбираема и позволява по-лесно да откриваме и отстраняваме неизправности в нея. Той също така насърчава прилагането на детерминистични модели на трафика. В този раздел ще направим преглед на проектантските модели, препоръчваните практики и методологии за постигане на висока достъпност на мрежата на фирмения комплекс (campus network).

## 1.1 Преглед на инфраструктурата на фирмения комплекс

Градивните елементи на мрежата на фирмения комплекс са слоя за достъп, разпределителния слой и слоя на ядрото [1]. Основните характеристики, свързани с всеки слой са йерархичен дизайн и модулност. Йерархичният дизайн избягва необходимостта от използване на топология при която всички възли са свързани помежду си, всеки с всеки (full mesh). Модулният дизайн позволява отделни компоненти на мрежата да бъдат въвеждани и извеждани от експлоатация с малко или никакво влияние върху останалата част на мрежата. Тази методология също така улеснява управлението на мрежата, отстраняването на възникналите неизправности, както и изолирането на проблемите.

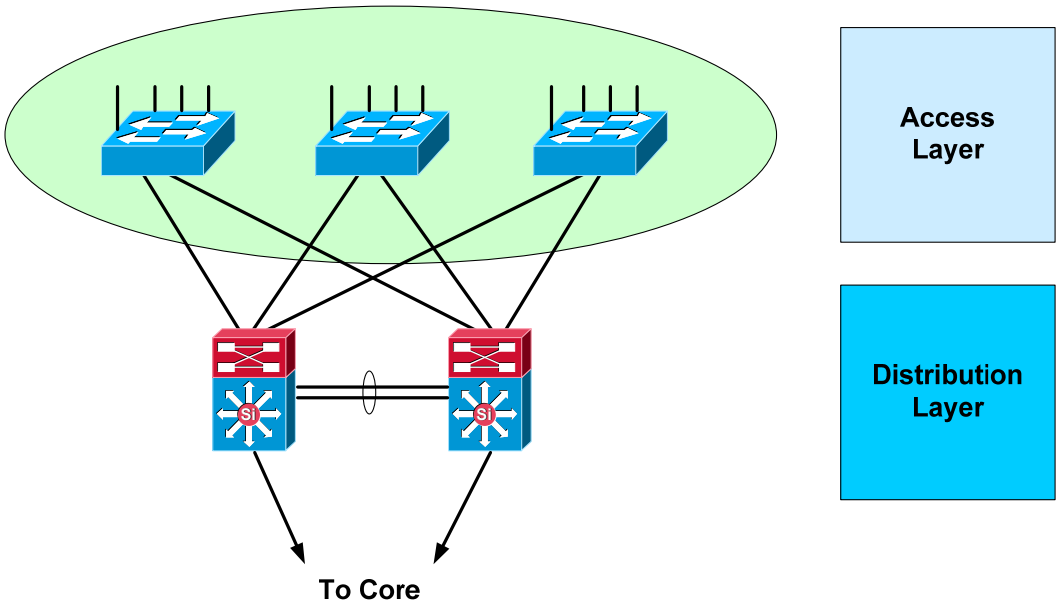
### 1.1.1 Слой за достъп (Access Layer)

През слоя за достъп крайните устройства се присъединяват към мрежата, както е показано на Фигура 1. Този слой агрегира крайните потребители и осигурява техния достъп към разпределителния слой. Слоят за достъп поддържа множество функции, като например:

- **Висока надеждност на достъпа (High availability):** Високата надеждност на достъпа през този слой е гарантирана от множество различни хардуерни и софтуерни характеристики. От хардуерна гледна точка, в свързващите комутатори имаме инсталирани резервни интелигентни управляващи модули (redundant supervisor engines) и резервни захранвания. Използват се резервни шлюзове по подразбиране, както и дублирани връзки между комутаторите в слоя за достъп и основните и резервни комутатори в разпределителния слой. От софтуерна гледна точка, високата надеждност се постига чрез използването на протоколи за дублиране на първия скок (First Hop Redundancy Protocol – FHRP), като например Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), и Gateway Load Balancing Protocol (GLBP).

- **Конвергенция (Convergence):** Слоят за достъп има вградена поддръжка за IP телефония. Тук са разположени и точките за безжичен достъп (Wireless Access

Points). Това позволява на клиентите да предават едновременно глас и данни, както и осигурява за тях роуминг в безжична локална мрежа.




Фигура 1 Слой за достъп

- **Сигурност (Security):** Слоят за достъп предоставя допълнителна защита срещу неоторизиран достъп до мрежата чрез използването на инструменти като IEEE 802.1X, сигурност на портовете (port security), DHCP подслушване (DHCP snooping), динамична проверка на ARP (dynamic ARP inspection), както и защита от определен източник на трафик (IP source guard).

- **Качество на услугата (QoS):** Слоят за достъп позволява приоритизирането на критичния за мрежата трафик, като използва класифициране на трафика и система от опашки. Тези опашки се разполагат колкото е възможно по-близко до точката на проникване в мрежата. Слоят поддържа използването на доверителна граница (trust boundary) на QoS.

- **IP групово предаване (IP multicast):** Слоят за достъп поддържа ефективно управление на трафика с помощта на софтуерни функции като например подслушване от страна на протокола за управление на груповото предаване (Internet Group Management Protocol - IGMP).

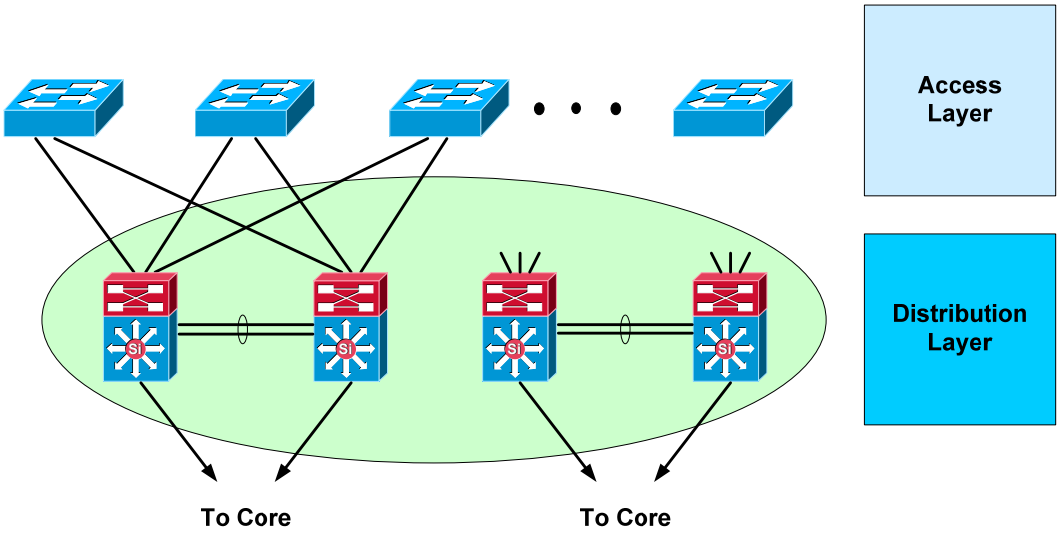
### 1.1.2 Разпределителен слой (Distribution Layer)

Разпределителния слой агрегира трафика от всички възли на слоя за достъп и осигурява свързаност съгласно възприета политика (policy-based connectivity), както е показано на Фигура 2.

При изграждането на този слой трябва да се съобразим с такива характеристики като висока надеждност, балансиране на натоварването, QoS и ефективна доставка на пакети и ресурси. Високата надеждност се осигурява чрез дублирани пътища от разпределителния слой към ядрото на мрежата и от слоя за достъп към разпределителния слой. Към ядрото обикновено трафикът преминава и по двата пътя, като с това се постига балансиране на натоварването.

Разпределителния слой е мястото, където се извършва обработката на пакетите и маршрутизацията. Този слой се явява като точка за преразпределение между маршрутизиращите домейни или като линия на разграничаване между статични и динамични протоколи за маршрутизация. Разпределителният слой изпълнява такива

задачи като контролирано маршрутизиране и филтриране въз основа на определена политика, както и QoS. За да се подобри ефективността на маршрутизиращите протоколи, разпределителния слой обобщава маршрутите от слоя за достъп. При някои мрежи разпределителния слой предлага подразбиращ се маршрут за маршрутизаторите от слоя за достъп, като в същото време използва динамични маршрутизиращи протоколи при комуникацията с маршрутизаторите в ядрото.




Фигура 2 Разпределителен слой

Разпределителния слой използва комбинация от обикновени комутатори от слой 2 и многослойни комутатори, за да раздели работните групи и да изолира мрежовите проблеми, като не позволи те да се пренесат към ядрото. В този слой се намират и крайните точки на виртуалните локални мрежи (Virtual Local Area Networks – VLANs). Разпределителният слой свързва мрежовите услуги със слоя за достъп и прилага разработените политики за QoS, сигурност, разпределение на трафика и маршрутизиране. Той осигурява дублиране на шлюза по подразбиране, като използва протокол FHRP (например HSRP, GLBP или VRRP). Това дава възможност повредата или отстраняването на един от възлите в този слой да не окаже влияние върху крайната свързаност на шлюза по подразбиране.

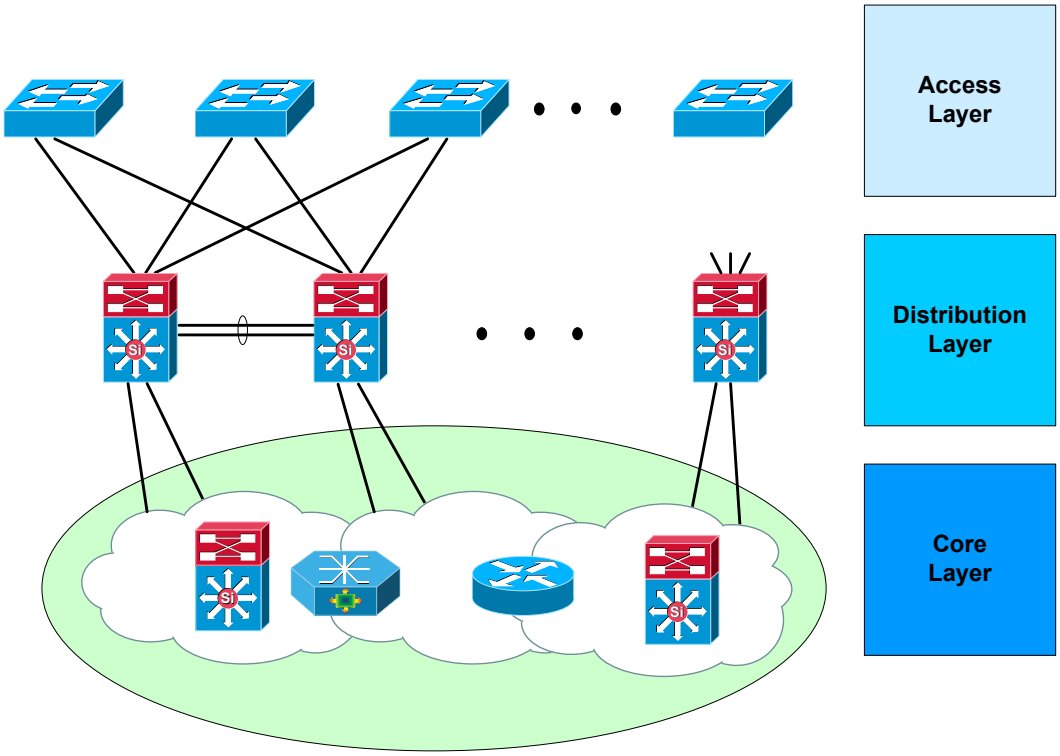
### 1.1.3 Слой на ядрото (Core Layer)

Слоят на ядрото (Фигура 3) осигурява мащабируемост и висока надеждност. Това е гръбнакът на мрежата на фирмения комплекс и служи като точка за агрегиране на останалите слоеве и модули в архитектурата на Cisco за корпоративна мрежа. В ядрото имаме висока степен на дублиране на връзките и устройствата, което позволява то бързо да се адаптира към настъпили промени. Устройствата, които се намират тук, са най-надеждни. Те могат да откриват възникнали неизправности и да реагират бързо, като пренасочват трафика при промени в топологията на мрежата [2]. Мрежовите устройства на ядрото използват мащабируеми протоколи и технологии, алтернативни пътища и балансиране на натоварването. С такива характеристики, значителното нарастване на ядрото не представлява проблем.

Ядрото е високоскоростна, комутираща среда от слой 3. Тя трябва да използва колкото е възможно повече хардуерно ускорени услуги [2]. Съществуването на много пътища от точка до точка позволява при повреда в някой възел или връзка мрежата да продължи да функционира и бързо да се възстанови нейната производителност. Ядрото е проектирано така, че да се избегне в него такива обработки на пакетите като проверка на списъци за достъп и филтриране, които забавят комутирането на пакетите.

Не всички реализации на мрежата на фирмения комплекс предполагат съществуването на ядро. При по-малките мрежи, функциите на слоя на ядрото и на разпределителния слой

могат да бъдат обединени и да бъдат поместени в разпределителния слой. Когато нямаме ядро, многослойните комутатори на разпределителния слой трябва да бъдат свързани всеки със всеки (full mesh), както е показано на Фигура 4. Такава конструкция е трудна за мащабиране и увеличава изискванията към окабеляването, защото всеки нов комутатор трябва да се свърже с всички останали. Сложността на маршрутизирането при такъв дизайн нараства, тъй като се добавят много нови съседи.

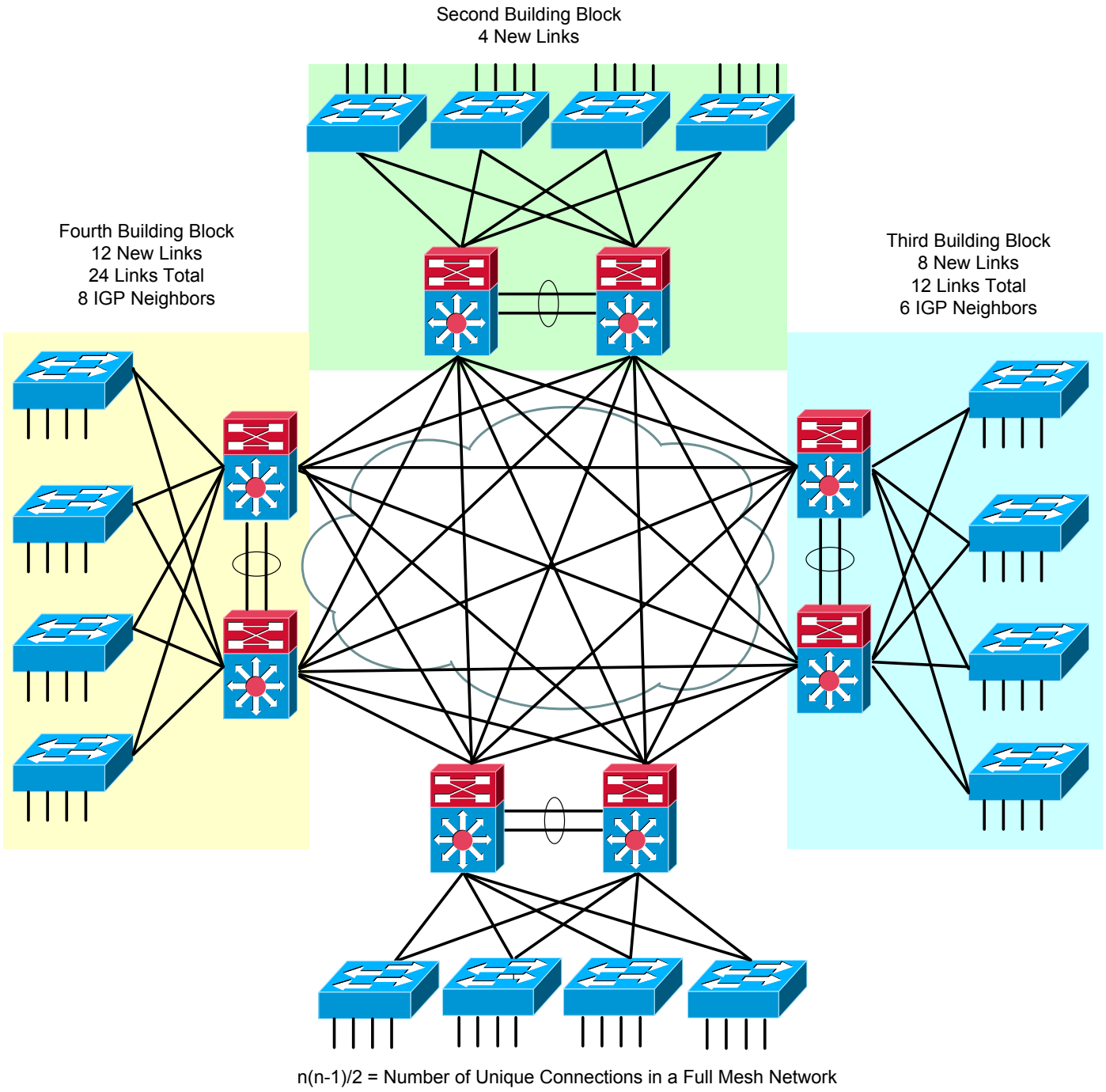



Фигура 3 Слой на ядрото

Обединението на функциите на разпределителния слой и на слоя на ядрото, т.е използването на деформирано ядро (collapsed core), изисква голяма гъстота на портовете на многослойните комутатори в разпределителния слой. Алтернативното решение е ядро изградено с комутатори от слой 2 с обособени виртуални локални мрежи (VLANs) на всеки комутатор от ядрото. Този сценарий изисква само два порта на даден комутатор към определено направление (към дадена сграда).

Както се вижда от Фигура 4, разпределителният модул във втората сграда, с два свързани комутатора, изисква четири допълнителни връзки за свързване с разпределителния модул в първата сграда. Третият разпределителен модул, обслужващ третата сграда, изисква осем допълнителни връзки, за да се свърже към всички комутатори в първата и втората сграда. Четвъртият модул, обслужващ четвъртата сграда, изисква 12 допълнителни връзки, за да се свърже с всички комутатори в първите три сгради, или имаме общо 24 връзки между комутаторите в разпределителния слой. При четири разпределителни модула, когато използваме вътрешен маршрутизиращ протокол (Interior Gateway Protocol -IGP), имаме вече за всеки комутатор по осем съседни устройства.

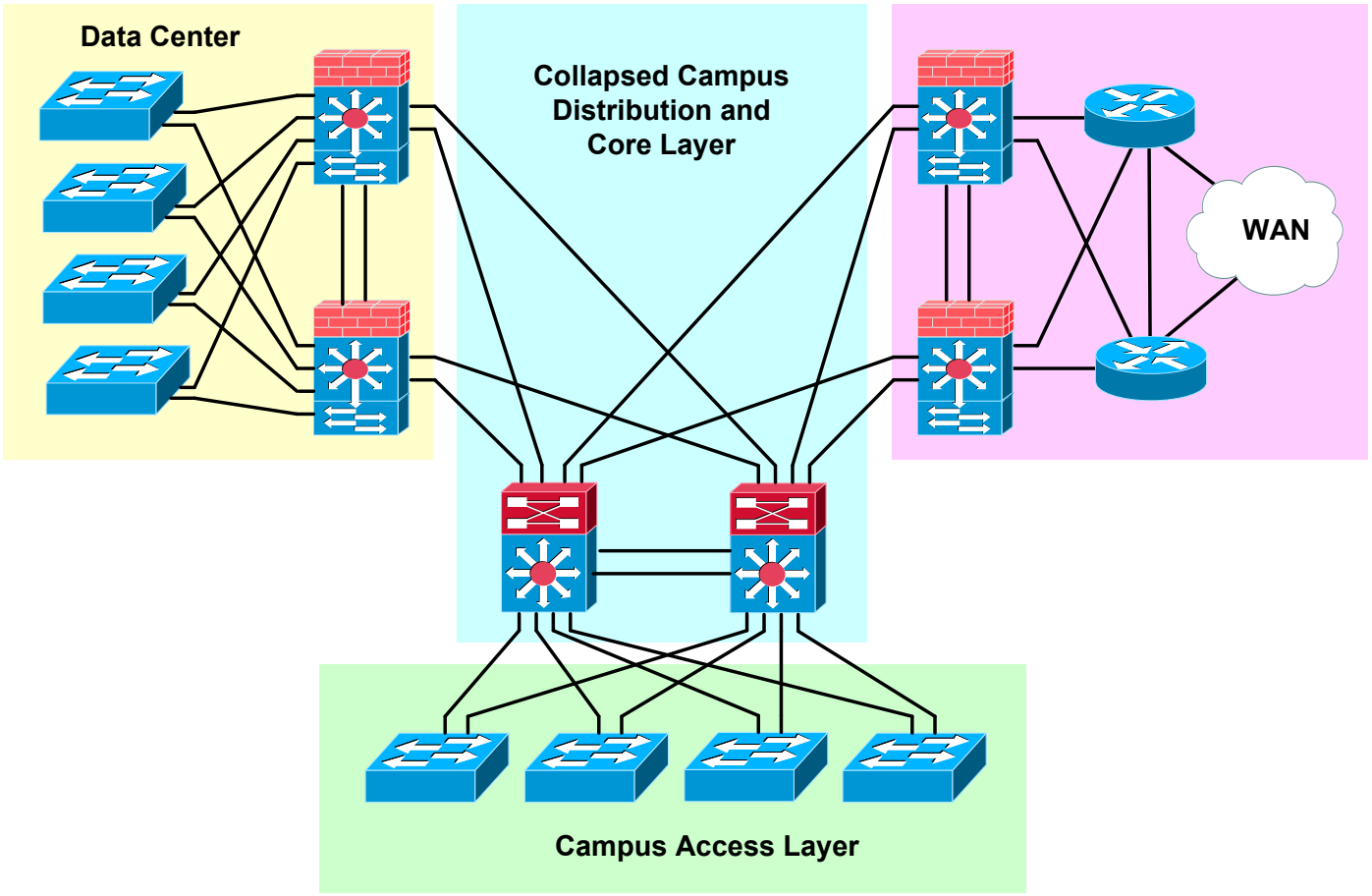
При три или повече сгради на фирмения комплекс препоръката е да се използва специално обособено ядро на мрежата. При много големи мрежи е желателно да се използват четири или повече двойки разпределителни комутатори. Специализираното ядро помага за по-лесното мащабиране на мрежата и за изпълнението на изискванията за плътност на Gigabit портовете, интеграцията на глас и данни, както и на конвергенцията между локалните, глобални и градски мрежи.




Фигура 4 Необходимо ли ни е ядро?

### 1.1.4 Модел със свито, деформирано ядро (Collapsed-Core Model)

В по-малките мрежи обединяването на слоя на ядрото с разпределителния слой елиминира нуждите от допълнителен комутиращ хардуер и улеснява изпълнението на мрежата. Това обаче премахва предимствата на многослойната архитектура и по-специално изолирането на възникнали неизправности. На Фигура 5 е показана топологията на мрежа със свито ядро. При този дизайн промените в разпределителния слой автоматично водят до промени в устройствата на ядрото. Ето защо изменения в някои части на разпределителния слой могат да се отразят на трафика в други части на мрежата. Например, в мрежата която е показана на Фигура 5, обновяването на софтуера, необходимо за въвеждане на нови функции в разпределителния слой, потенциално може да се отрази на трафика между центъра за данни и WAN модулите. Когато се използва отделно ядро и разпределителен слой, проблемите остават изолирани в рамките на дадения слой.




Фигура 5 Топология със свито ядро

**Забележка:** В комутаторите от серията Cisco Nexus 7000, може да се използват т.н. контексти на виртуални устройства (virtual device contexts - VDC), за комбиниране на слоя на ядрото и на разпределителния слой само в едно физическо устройство, като едновременно се поддържа функционално и административно разделение между слоевете. Въпреки това логическо разделение, проблемите с физическото изолиране остават.

## 1.2 Съображения при проектирането на високо достъпна мрежа

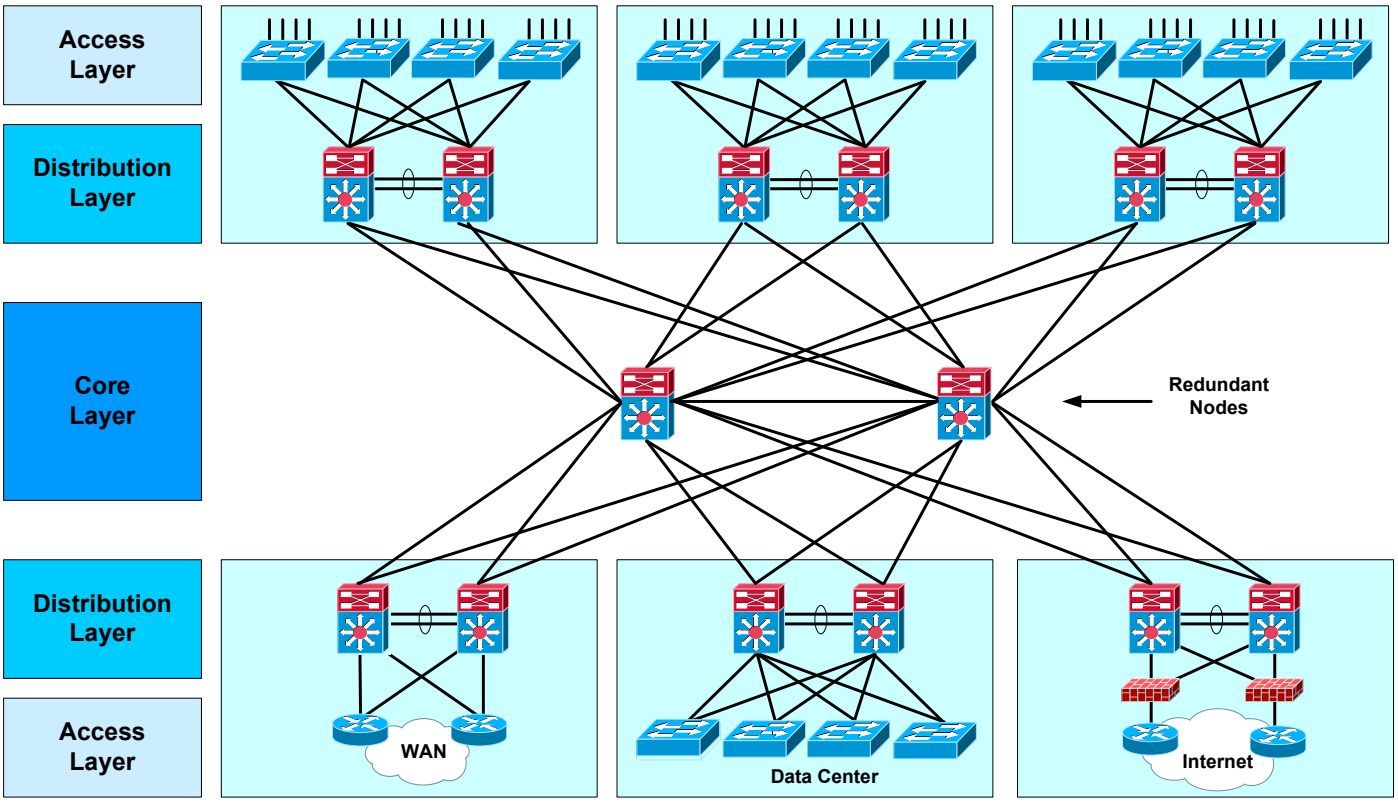
При проектирането на високо достъпна мрежа за фирмения комплекс, ние се стараем да сведем до минимум повредите в отделните възли и връзки, както и да оптимизираме времената за възстановяване на трафика. Целта е времето за престой на мрежата да бъде минимално.

### 1.2.1 Въвеждане на редундантност (Redundancy)

Препоръчителният дизайн включва използването на допълнителни, редундантни комутатори в разпределителния слой, редундантни връзки към ядрото, както и връзки в слой 3 между комутаторите в разпределителния слой. Комутаторите в слоя за достъп трябва да са свързани с редундантните комутатори в разпределителния слой, както е показано на Фигура 6.

Както вече беше казано, препоръката е в слоя на ядрото и в разпределителния слой да има допълнителни комутатори и връзките да са всеки с всеки. Това осигурява максимална редундантност и оптимална сходимост на протоколите. Честотната лента и капацитетът на мрежата трябва да са проектирани така, че при повреда в комутатор или връзка, в повечето случаи сходимостта на маршрутизиращите протоколи да варира в границите от 120 ms до 200 ms. Таймерите на маршрутизиращите протоколи Open Shortest Path First (OSPF) и

Enhanced Interior Gateway Routing Protocol (EIGRP) трябва да бъдат така настроени, че максимално бързо да пренасочат трафика по обходни пътища.




Фигура 6 Конфигурация с оптимален излишък

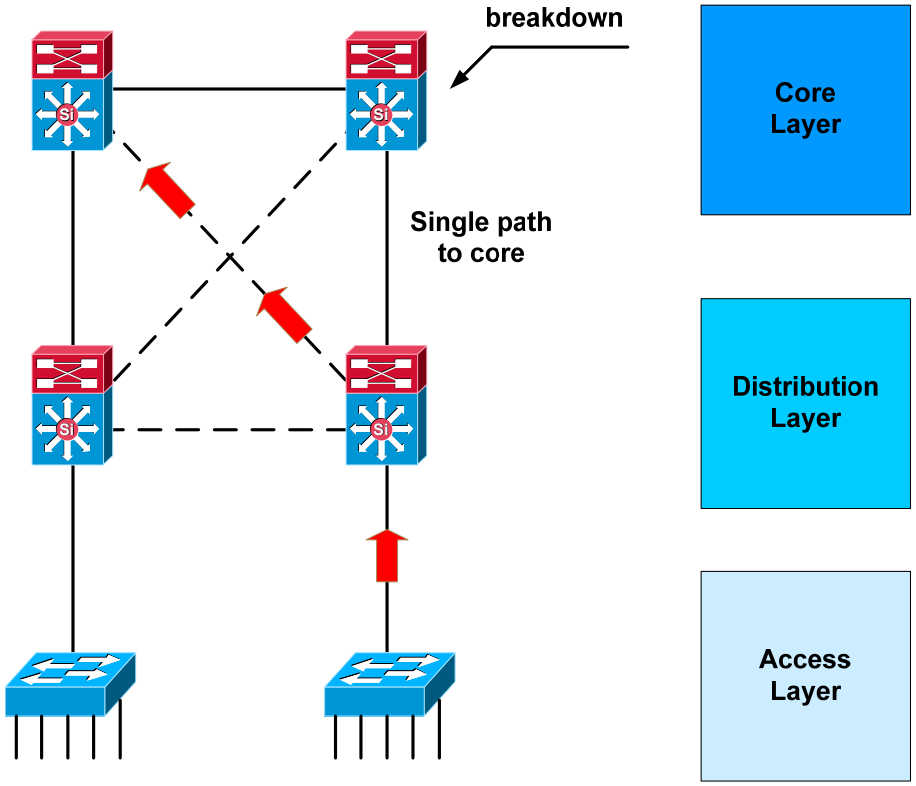
Да отбележим, че при напълно редундантна топология, добавянето на редундантни надзорни модули (supervisors) при работа с такива технологии на Cisco, като препращане без прекъсване (Nonstop Forwarding – NSF) и динамична защита на комутирането (Stateful Switchover - SSO), може да доведе до по-дълги времена на сходимост, отколкото ако използваме само един надзорен модул с добре настроени таймери на вътрешните маршрутизиращи протоколи (IGP). NSF се опитва да поддържа потока на трафика през маршрутизатор претърпял повреда. NSF заедно със SSO поддържат връзките от слой 3 до постигането на сходимост на маршрутизиращите протоколи. Въпреки това, взаимодействието между таймерите на IGP и на NSF може да доведе до объркване и да предизвика прекъсване на връзките от страна на съседните маршрутизатори.

Настройката на таймерите на OSPF и EIGRP се използва за подобряване на времето за сходимост в бродкастни мрежи с множествен достъп (например Ethernet). Първоначалният сценарий за използването на NSF със SSO предполага, че маршрутизаторите ще работят в модула за връзка с външния свят (Enterprise Edge), където връзките обикновено са от точка до точка (например към Интернет доставчика, или пък към Gigabit порт на комутатор в инфраструктурата на фирмения комплекс).

В нередундантни топологии използването на NSF със SSO, заедно с допълнителни редундантни надзорни модули, води до значителни подобрения и голяма гъвкавост. Технологиите NSF и SSO ще бъдат разгледани по-подробно в някой от следващите доклади.

## 1.2.2 Осигуряване на алтернативни пътища

Препоръчителния дизайн на разпределителния слой включва редундантни комутатори и редундантни връзки към ядрото, както и връзка в слой 3 между разпределителните комутатори, както е показано на Фигура 7.




Фигура 7 Осигуряване на алтернативни пътища

Въпреки че двата комутатора в разпределителния слой са свързани индивидуално към различни комутатори в ядрото (плътни линии) и това намалява броят на връзките и заетите портове в ядрото, този дизайн не предоставя достатъчно редундантност. При повреда на връзка или комутатор трафикът се прекъсва. За да осигурим алтернативни пътища за трафика, трябва да свържем всеки комутатор от разпределителния слой с някой друг комутатор от ядрото (пунктирани линии). Връзката между двата комутатора в разпределителния слой е необходима за обобщаването на информацията за маршрутизиране, предавана от разпределителния слой към слоя на ядрото. На Фигура 7 със стрелки е показан пътят на трафика при повреда на комутатор в ядрото.

### 1.2.3 Избягване на критични точки (Single Points of Failure)

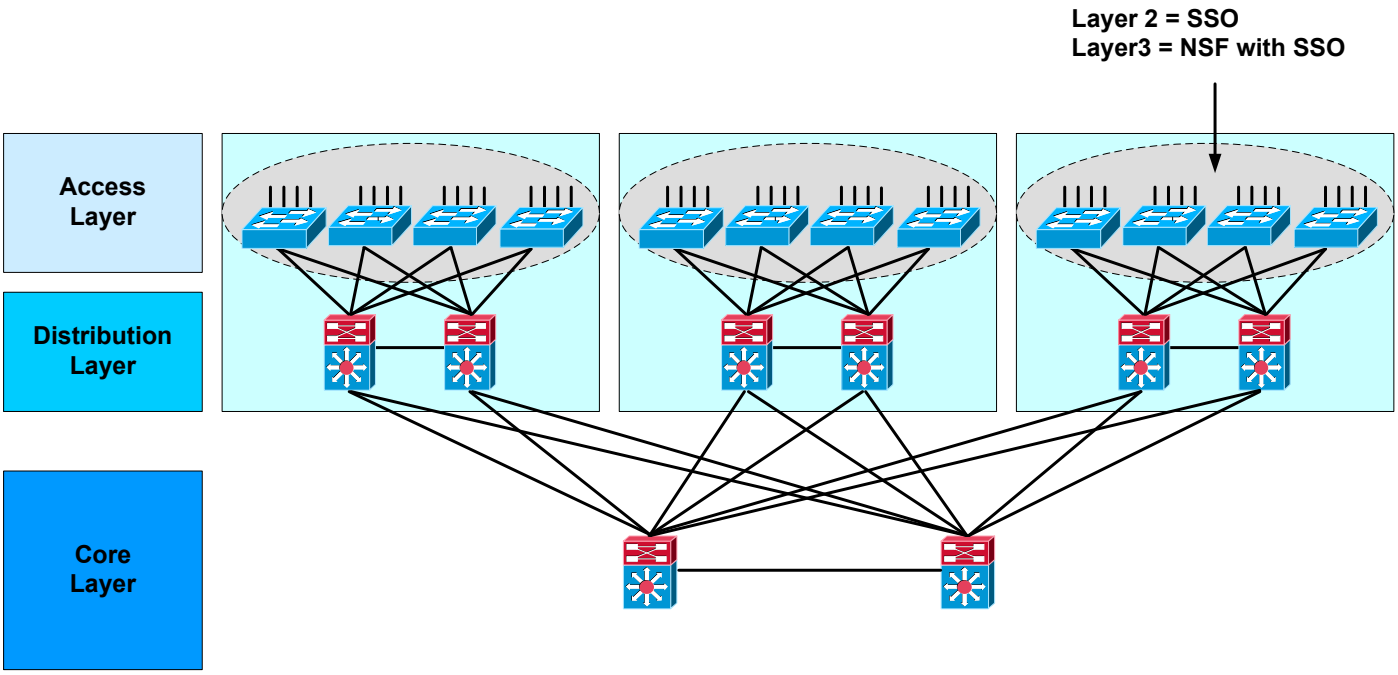
Технологията на едновременно използване на NSF, SSO и допълнителни (резервни) надзорни модули в устройствата, оказва най-голямо влияние върху мрежата на фирмения комплекс, ако се използва в слоя за достъп. Всеки комутатор в този слой всъщност се явява критична точка, и при повредата му, връзката на крайните устройства свързани чрез него с мрежата се прекъсва. Времето за прекъсване може да се намали до интервала от една до три секунди, ако се използва SSO в слой 2 и NSF със SSO в слой 3, както е показано на Фигура 8. Функцията SSO е имплементирана в Catalyst 4500, комутаторите от сериите 6500/7600, както и в маршрутизаторите за интегрирани услуги (Aggregation Services Routers - ASR).

### 1.2.4 Използване на NSF и SSO

Технологията за едновременно използване на NSF и SSO представлява механизъм на операционната система Cisco IOS, който механизъм позволява много бързо превключване между дублирани надзорни модули (supervisors) инсталирани в мрежовите устройства. Той обслужва нива от 2 до 4.

SSO позволява резервният маршрутизиращ процесор (route processor – RP) [2] да поеме управлението на устройството, след като в активния RP са настъпили хардуерни или софтуерни неизправности. SSO синхронизира всички такива параметри като: стартова

конфигурация, стойности на стартовите променливи, работна конфигурация, динамични стойности на работните променливи, състояние на протоколите от слой 2, на магистралите (trunks) и портовете, хардуерните таблици на слой 2 и слой 3 (MAC, FIB, Adjacency) [2], списъците за контрол на достъпа (Access Control Lists - ACL) и таблиците на QoS.




Фигура 8 Избягване на критични точки

NSF е функция на слой 3, която работи заедно със SSO, за да минимизира времето, през което мрежата е недостъпна, когато управлението на устройството се прехвърля от единия надзорен модул към другия. Основната цел на NSF е да продължи препращането на IP пакетите, докато това прехвърляне се осъществява. NSF се поддържа от такива маршрутизиращи протоколи като EIGRP, OSPF, IS-IS и BGP. Маршрутизаторът, който използва тези протоколи, може да открие това вътрешното превключване на модулите и да предприеме необходимите действия за да продължи препредаването на мрежовия трафик. Той използва метода за експресно пренасочване (Cisco Express Forwarding - CEF) [2] докато се възстанови маршрутната информация и се получи пълна сходимост.

Обикновено при рестартирането на един маршрутизатор, всички съседни маршрутизатори откриват, че в техните таблици за съседство (Adjacency table) са настъпили промени – маршрутизаторът изчезва, след което се появява. Този процес се нарича маршрутно объркване (routing flap), и той нарушава стабилното състояние на протокола. Маршрутното объркване създава маршрутни нестабилности, които са пагубни за общата производителност на мрежата. NSF помага за потискането на маршрутните объркванията.

NSF позволява препращането на пакетите от данни по известните маршрути за продължи, до възстановяването на маршрутната информация, нарушена вследствие на смяната на надзорните модули. Когато се използва NSF, съседните маршрутизатори не установяват маршрутно объркване, тъй като интерфейсите остават активни и таблиците за съседство не се променят. Предаването на данните продължава, както и поддържането на потребителските сесии, установени преди замяната на модулите.

Способността на интелигентните линейни модули (line cards) [2] да останат активни в процеса на смяната на надзорните модули като запазват текущото състояние на FIB от активния RP, е от решаващо значение за правилното функциониране на NSF. Докато плоскостта за контрол на мрежата изгражда новата база данни на маршрутизиращия протокол и рестартира договорките между възлите, плоскостта за предаване на данни разчита на маршрутната таблица от преди смяната за да продължи предаването на

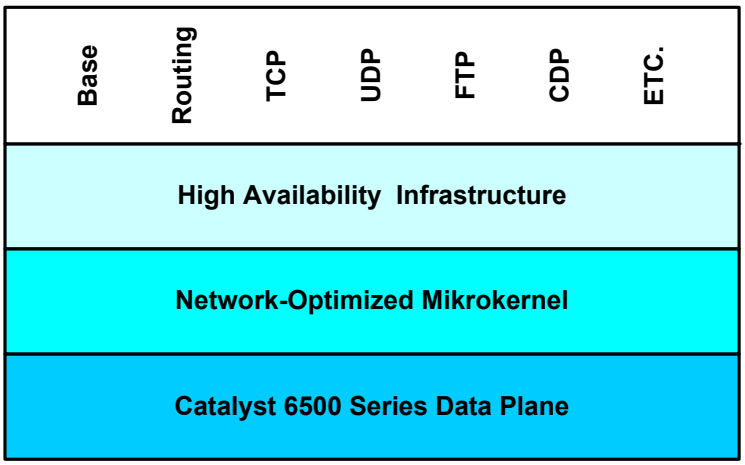
трафика [2]. След завършване на процеса на сходимост на маршрутизиращите протоколи, RP променя FIB таблицата, отстранява остарелите записи и зарежда линейните модули с актуализираната FIB таблица.

Смяната на надзорните модули трябва да завърши преди изтичането на таймерите (dead and hold timers) за елегантно рестартиране, в противен случай съседите променят своите таблици за съседство и пренасочват трафика.

Последните подобрения на NSF протокола позволяват на маршрутизатора да сигнализира на съседните NSF маршрутизатори, че в момента се извършва смяна на надзорните му модули.

### 1.2.5 Модулна архитектура на софтуера Cisco IOS

Комутаторите на Cisco от серията Catalyst 6500 са с операционна система IOS с модулна архитектура. Тази модулност подпомага високата достъпност на корпоративната мрежа. На Фигура 9 са показани основните елементи и компоненти на тази архитектура [3].




Фигура 9 Модулна архитектура на софтуера Cisco IOS

Когато операционната система IOS не е изградена на модулен принцип и трябва да се направят определени промени в нея (кръпки), след нейното компилиране тя се зарежда като ново изображение (имидж) в надзорните модули (активният и резервният). Това налага тяхното рестартиране. При Catalyst 6500 функциите на плоскостите за контрол и управление на мрежата (отговарящи за маршрутната актуализация и прокарването на управляващ трафик), се изпълняват от специални микропроцесори, намиращи се в комплекса от модули за пренасочване в многослойния комутатор (Multilayer Switch Forwarding Card - MSFC). Плоскостта за пренасочване на пакетите с данни от потребителския трафик е напълно отделена. Когато хардуерът е програмиран за непрекъсната работа, тази плоскост продължава пренасочването на пакетите, дори когато се случи срив в плоскостта за контрол на мрежата. Модулната архитектура на IOS дава възможност за изграждане на една по-устойчива плоскост за контрол на мрежата. Повече за отделните плоскости на трафика в IP маршрутизаторите можете да намерите в [2].

Комутаторите от серията Catalyst 6500 с модулна архитектура на операционната система позволяват различните подсистеми в равнината за контрол на мрежата да работят като независими процеси. Модулността на IOS подобрява оперативната ефективност и минимизира времето на престой. Най общо тя:

- Свежда до минимум непланираните престои чрез ограничаване на повредите и защитено динамично рестартиране на процесите.

- Опростява софтуерните промени чрез системата за обслужване на софтуерните подобрения (In-Service Software Upgrades - ISSU), а също така значително намалява сертифицирането на кода и на сроковете за внедряване.

- Дава възможност за автоматичен контрол на ниво процес като интегрира вградената в операционната система програма за управление на събитията (Cisco IOS Embedded Event Manager - EEM), разтоварването на времеемките задачи към мрежата и ускоряване на отстраняването на мрежовите проблеми. EEM е комбинация от процеси предназначени за наблюдение на ключови параметри на системата, такива като използването на процесора, интерфейсните броячи, събитията в протоколите syslog и SNMP. Тя реагира при появата на определени събития, например при превишаване на стойностите на някои броячи.

### 1.2.6 Предимства на модулната архитектура на софтуера

В този раздел щи опишем по-подробно предимствата на модулната архитектура на IOS на комутаторите от серията Cisco Catalyst 6500:

- **Оперативна съвместимост (Operational consistency):** Въвеждането на модулност в IOS не изменя нейната функционалност. Интерфейсът на командния ред (Command Line Interface – CLI) и интерфейсите за управление на протоколите SNMP и syslog са същите, както и преди. За да се реализират новите възможности са въведени само някои нови команди за конфигуриране и наблюдение.

- **Защитена памет (Protected memory):** Модулната IOS използва организация на паметта, при която процесите използват защитено адресно пространство. Всеки процес и свързаната с него подсистема се изпълнява в индивидуална, специално заделена за него памет. Използвайки този модел няма вмешателство на един процес в работата на друг чрез нерегламентирани промени в паметта.

- **Ограничаване на дефектите (Fault containment):** Ползата от използването на защитена памет се проявява в увеличение на наличността, тъй като проблемите възникнали в един процес не засягат други части на системата. Например ако един не толкова важен системен процес се повреди или не работи според очакванията, критично важните функции необходими за препращането на пакетите остават незасегнати.

- **Рестартиране на процеси (Process restartability):** Използването на защитена памет и ограничаването (херметизирането) на дефектите, позволява модулните процеси индивидуално да бъдат рестартирани. С цел тестване или при нереагиращи процеси можем да използваме командата *process restart process-name* за да рестартираме ръчно даден процес. Рестартирането на процес позволява бързо възстановяване от преходни грешки, без да е необходимо да се прекъсне комутацията на пакети. Интегрираната високо налична инфраструктура на комутатора непрекъснато проверява състоянието на процесите и следи колко пъти даден процес рестартира за определен период от време. Ако рестартирането на процеса не възстанови системата, инфраструктурата предприема по-драстични действия, като например започва процедура за превключване на надзорните модули или рестартиране на цялата система.

- **Модулни процеси (Modularized processes):** Редица функции са организирани като отделни модули, които покриват най-често срещаните особености. Като отделни модули са изградени например такива процеси като: маршрутизиращ процес (Routing process), Интернет услуга (Internet daemon), сурова IP обработка (Raw IP processing), TCP процес, UDP процес, CDP процес, Syslog услуга (Syslog daemon), всякакви EEM

компоненти, файлови системи, медийни драйвери и инсталационна програма (Install manager).

- **Подсистема ISSU (Subsystem ISSU):** Модулността позволява поддръжката на избрани подсистеми по време на нормалното функциониране на комутатора, като се правят индивидуални кръпки в софтуера. Чрез системата за проверка на версиите и номерата на кръпките, модулната IOS позволява тяхното сваляне, проверяване, инсталиране и активиране, без да е необходимо рестартирането на системата. Понеже пакетите в плоскостта за предаване на данни не са засегнати и продължават да бъдат препращани в процеса на въвеждане на кръпки в системата, мрежовият администратор разполага с голяма гъвкавост при въвеждането на софтуерни промени по всяко време. Кръпката засяга само софтуерните компоненти, свързани с актуализацията.

## 2. Проектиране на оптимален дизайн на слой 2

За изграждането на високо достъпна и детерминирана топология на слой 2, различните архитектури на този слой разчитат на следните технологии:

- Премахване на вредните затворени контури – Spanning Tree Protocol (STP) [4].
- Използване на магистрали за предаване на данни – Trunking (ISL/802.1Q) [5].
- Откриване на еднопосочни връзки – Unidirectional Link Direction (UDLD) [6]
- Групиране на връзки – EtherChannel [7].

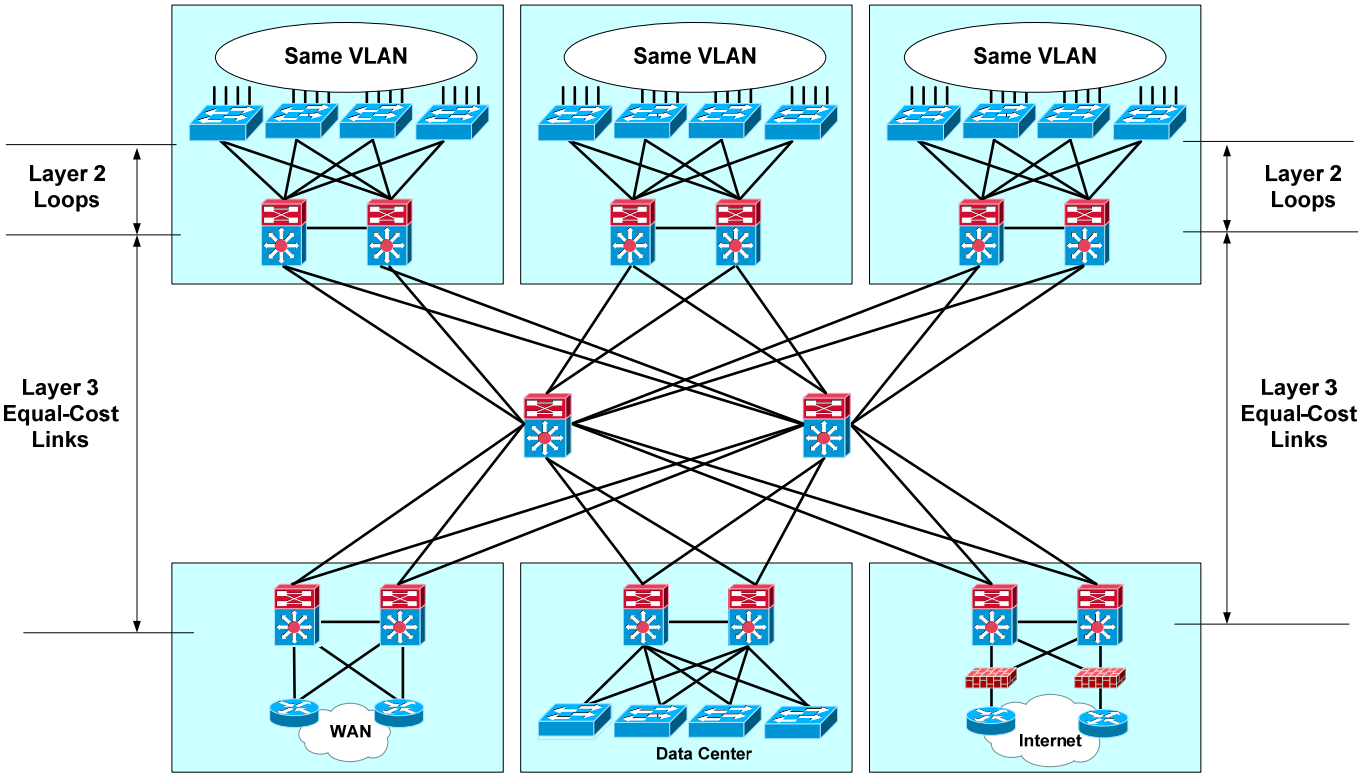
В този раздел е направен преглед на дизайна и на препоръчителните практики за оптимизиране на слой 2 на мрежата на фирмения комплекс.

## 2.1 Препоръчителни практики за конфигуриране на STP

Детерминираната и високо достъпна топология на мрежата на фирмения комплекс предвижда използването на редундантни връзки, както е показано на Фигура 10. Това води до появата на затворени контури в слой 2 (Layer 2 Loops) и на връзки с равни разходи в слой 3 (Layer 3 Equal-Cost Links).

Нуждата от използване на протокола STP възниква в следните ситуации:

- Когато потребителите на една виртуална локална мрежа (Virtual Local Area Network – VLAN) са свързани към различни комутатори в слоя за достъп (Фигура 10)
- Когато трябва да се защитаваме срещу възникването на затворени контури по вина на потребителя. Дори когато мрежата е проектирана така, че не разчитаме на STP, то може да се окаже, че този протокол е необходим за предпазване от определени действия на потребителя. Това могат да бъдат грешки в окабеляването, лошо конфигурирани работни станции или просто да има злонамерени потребители. STP е необходим, за да осигури топология свободна от затворени контури и да защити останалата част от мрежата от проблеми създадени в слоя за достъп.
- Когато имаме изграден изчислителен център оформен като група от сървъри (server farm) и бизнес приложенията налагат изисквания за използване на STP.




Фигура 10 Топология на високо достъпна мрежа изискваща STP

Винаги ще стои въпросът кога да използваме STP? Някои мрежови администратори от съображения за сигурност препоръчват блокирането на STP в модула за връзка с външния свят (Enterprise Edge) [1]. Такава практика не е за препоръчване, тъй като рискът от загуба на връзки без STP е много по-голям. Възможно е да възникне затворен контур (bridging loop), и той да доведе до загуба на свързаност и невъзможност за управление на мрежовите устройства. В този случай трябва да се използва протоколът Rapid Per-VLAN Spanning-Tree Plus (RPVST+) [8]. Можете да се възползвате и от направените подобрения на STP известни като Cisco STP Toolkit.

### 2.1.1 Инструментариум на Cisco за STP (Cisco STP Toolkit)

Подобренията на Cisco за STP включват:

- **PortFast*:** Тази функция предизвиква интерфейсът от слой 2, конфигуриран като порт за достъп, да премине в състояние forwarding незабавно, прескачайки състоянията listening и learning. Използвайте PortFast само при свързване на една работна станция към порт за достъп от слой 2.

- **UplinkFast:** Постига време на сходимост от три да пет секунди след прекъсване на директна връзка и осигурява балансиране на натоварването по редундантни връзки към разпределителния слой

- **BackboneFast:** Съкращава времето на сходимост чрез *max_age* при индиректна повреда. Тази функция стартира когато коренен порт (root port) или блокиран порт (blocked port) на мрежово устройство получи от назначения (designated) мост пакет BPDU (Bridge Protocol Data Unit) с по висок мостов идентификатор (Bridge ID). Такива пакети се наричат  inferior BPDU.

- **Loop guard*:** Предотвратява коренен порт или резервен коренен порт да станат назначени (designated), когато не получават BPDU пакети. Това предпазва от появата на затворени контури и зацикляния при възникване на еднопосочност на връзка от точка до точка.

- **Root guard*:** Запазва коренните функции на конкретен комутатор, като не позволява на друг, външен комутатор, да стане корен на дървото.

- **BPDU guard*:** Когато тази функция е конфигурирана на порт с разрешена функция PortFast, тя изключва порта когато той получи BPDU пакет.

- **UniDirectional Link Detection (UDLD):** Този протокол наблюдава физическата конфигурация на оптически или медни връзки и открива когато връзката е еднопосочна. Когато това стане, интерфейсът се изключва и се алармира системата.

- **Bridge assurance:** Когато тази функция е активирана, тя променя поведението на моста, като пакетите BPDU се изпращат по всички портове, а не както е в режим на нормална работа, когато BPDU пакетите се изпращат само към назначените (designated) портове. Когато порт с разрешена такава функция спре да получава BPDU пакети от свой съсед, той преминава в противоречиво (inconsistent) състояние, което не позволява потенциалното създаване на затворен контур [9].

Означените със * по-горе подобрения могат да се използват заедно с RPVST+. Инструментариумът на Cisco за STP поддържа също опция за филтриране на BPDU пакети (BPDU filter), която не позволява на порт с разрешена функция PortFast да изпраща и получава такива пакети. Тази възможност ефективно забранява STP в модула за връзка с външния свят. Не се препоръчва нейното използване.

### 2.1.2 STP стандарти и функции

STP позволява в мрежата да бъдат блокирани определени интерфейси, като по този начин се избягват затворените контури. Има няколко разновидности на STP:

- Оригиналната версия на STP е IEEE 802.1D. Тя осигурява дървовидна структура на връзките в мрежата, без затворени контури, и е специално разработена за мрежови топологии с редундантни връзки.

- Версията Common Spanning Tree (CST) поддържа една дървовидна структура обхващаща цялата мрежа, независимо от броя на използваните VLAN. Тази реализация намалява натоварването на процесора, тъй като се поддържа само една структура за цялата мрежа.

- Per VLAN Spanning Tree Plus (PVST+) е едно подобрение на Cisco на протокола STP, което изгражда отделна 802.1D дървовидна структура за всяка конфигурирана в мрежата VLAN. Всяка отделна структура поддържа PortFast, UplinkFast, BackboneFast, BPDU guard, BPDU filter, Root guard и Loop guard.

- Multiple Spanning Tree (MST) е IEEE стандарт, който е вдъхновен от имплементацията на по-ранната версия на патентования от Cisco протокол Multi-Instance Spanning Tree Protocol. При MST всяка VLAN се присъединява към една от многото дървовидни структури (instances), като всяка от тези структури е с различна топология от останалите. Такава архитектура предоставя множество пътища за пакетите с данни, което позволява балансиране на натоварването. Тя също така намалява броя на дървовидните структури необходими за поддържането на голям брой VLAN. MST подобрява отказоустойчивостта към откази на мрежата, тъй като повреда в една дървовидна структура не засяга другите дървовидни структури. Имплементацията на Cisco поддържа до 16 такива дървовидни структури, за всяка от които се използва протокола Rapid Spanning Tree Protocol (RSTP, 802.1W), и всяка поддържа PortFast, UplinkFast, BackboneFast, BPDU guard, BPDU filter, Root guard, и Loop guard. Някои

версии на софтуера поддържат до 65 структури за един MST домейн. Ето защо е необходимо да прочетете внимателно документацията на версията която използвате.

- RSTP или IEEE 802.1W представлява една еволюция на STP, която осигурява по-бърза сходимост.

- RPVST+ представлява подобрение на RSTP, което използва PVST+. То осигурява отделна дървовидна структура 802.1W за всеки VLAN. Всяка такава структура поддържа PortFast, UplinkFast, BackboneFast, BPDU guard, BPDU filter, Root guard и Loop guard.

Имплементацията на Cisco на RSTP превъзхожда 802.1D STP и PSVT+ по сходимост. Тя значително подобрява времето за възстановяване на всеки VLAN при промяна на топологията. Когато разполагаме с комутатори на различни производители, трябва или да се използва MST за постигане на максимална съвместимост, или чрез слой 3 да изградим различни STP домейни.

### 2.1.3 Препоръки за използване на инструментариума за STP

Когато искате един комутатор да изпълнява функциите на коренен мост (root bridge) за дадена VLAN, трябва да въведете командата *spanning-tree vlan vlan_ID root primary* за да смените приоритета по подразбиране на моста (32768) с някаква значително по-ниска стойност. В този случай приоритетът на моста става 8192, ако тази стойност е достатъчна за да го направи коренен мост за дадената VLAN. Ако тя не е достатъчна, т.е. съществува мост с по-нисък приоритет от 8192, комутаторът назначава приоритет на моста с 1 по-малък от приоритета на моста с най-нисък приоритет. Имате възможност и ръчно да определяте приоритетите на мостовете, което заедно с използването на опциите на STP Toolkit ви дава възможност да поддържате една детерминирана конфигурация, за която знаете кои портове трябва да бъдат активни и да препредават пакетите, и кои портове трябва да бъдат блокирани. Например на коренния мост можете да присвоите приоритет 0, а на резервния коренен мост – приоритет 1. Детерминираният избор на коренен мост увеличава мрежовата стабилност и намалява времето за сходимост.

За определяне на коренен мост на дървовидна структура в MST се използва командата *spanning-tree mst instance_id root primary*.

Фигура 11 илюстрира препоръките за използване на инструментариума за STP:

- Loop guard използваме на портовете от слой 2 между разпределителните комутатори, както и на портовете на комутаторите от слоя за достъп свързани с комутаторите в разпределителния слой (uplink ports).

- Root guard се конфигурира в разпределителните комутатори, на портовете свързани с комутаторите в слоя за достъп.

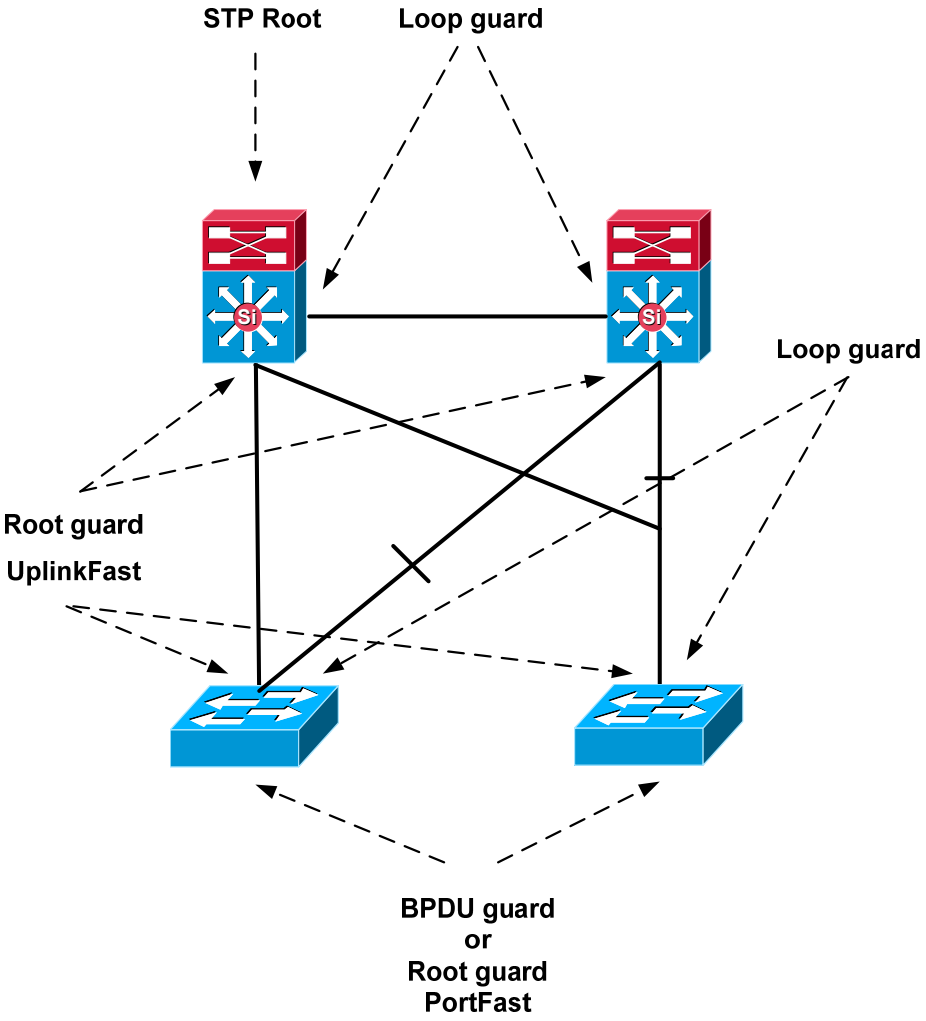
- UplinkFast използваме на портовете на комутаторите в слоя за достъп свързани с разпределителните комутатори (uplink ports). Да отбележим, че когато конфигурираме MST, функцията UplinkFast не трябва да се използва в комутаторите с резервирани връзки, т.н dual-homed switches.

- BPDU guard или Root guard се конфигурира на портовете на комутаторите в слоя за достъп свързани към потребителските устройства. Същото важи и за функцията PortFast.

- Протоколът UDLD позволява устройствата да наблюдават физическата конфигурация на кабелите и да откриват съществуването на еднопосочна връзка.

Когато настъпи такова събитие, UDLD изключва засегнатия порт. Конфигурираме този протокол обикновено на портовете, свързващи комутаторите помежду им.

- В зависимост от изискванията за сигурност на дадена организация, можем да използваме и функцията за сигурност на порт (port security), с която да ограничим входящия в него трафик. Това се постига чрез ограничаване на MAC адресите, на които е разрешено да изпращат пакети към дадения порт.




Фигура 11 Препоръки за използване на инструментариума за STP

## 2.2 Препоръчителни практики за конфигуриране на магистрални връзки

### 2.2.1 Магистрали връзки (Trunks)

Магистрална наричаме връзката от точка до точка между две мрежови устройства, по която преминава трафик на различни VLAN. Такива връзки обикновено се използват за свързване на слоя за достъп с разпределителния слой. Препоръчителната практика за тях е да се използва магистралния протокол IEEE 802.1Q. Имплементацията на 802.1Q на Cisco избягва проблемите със сигурността, свързани с използването в 802.1Q на един вътрешен, вроден (native), не маркиран с етикет (nontagged) VLAN. За магистрални връзки Cisco първоначално разработи собствен протокол Inter-Switch Link (ISL), който има структурни различия с 802.1Q, но теоретично постига същите резултати. ISL се среща вече рядко и то предимно в някои по-стари устройства, на които софтуера не позволява използването на 802.1Q.

VLAN Trunking Protocol (VTP) е протокол, който позволява на мрежовите администратори да управляват една централизирана база данни с информация за използваните VLAN. Препоръчителната практика е да използваме прозрачния режим (Transperant mode) на този

протокол, защото това намалява опасността от оперативна грешка. По подразбиране комутаторите на Cisco са конфигурирани като VTP сървъри без уточнено име на домейн. Ето защо се препоръчва при конфигурирането на комутаторите, заедно с определянето на режима като Transperant mode, да се зададе и име на VTP домейн. Това е особено важно в случаите, когато свързвате вашия комутатор към други домейни, например към комутатор на вашия доставчик на услуги. Неправилното конфигуриране на комутатора като сървър или клиент без име на VTP домейн, го кара да приеме името на домейна на съседен VTP сървър и да презапише локалната база данни

VTP работи само на магистрални връзки и предоставя следните четири режима:

- **Сървър (Server):** В този режим могат да се правят промени в базата данни. Комутаторът във VTP сървър режим разпространява VTP базата данни към комутаторите в режим на VTP клиент.

- **Клиент (Client):** Комутаторът в този режим получава актуализациите на базата данни, но на него не могат да се правят промени в нея.

- **Прозрачен (Transparent):** В този режим комутаторът не участва във VTP домейна. Той обаче позволява актуализациите на базата данни да преминават транзитно през него.

- **Изключен (Off):** Игнорира актуализациите на базата данни.

Когато използвате VTP, и конфигурирате нова VLAN в комутатор с режим Сървър, информацията за тази VLAN се разпространява до всички комутатори във VTP домейна. Така се избягва необходимостта да конфигурирате същата VLAN в другите комутатори.

В йерархичните мрежи при които VLAN не преминават през разпределителния слой, няма голям смисъл в съществуването на обща VLAN база данни. В препоръчания вече дизайн на фирмения комплекс, една и съща VLAN не би трябвало да се появява в два комутатора за достъп. Добавянето и премахването на VLAN по принцип не е обичайна практика в управлението на мрежите. В повечето случаи VLAN се дефинира еднократно по време на инсталирането на комутатора в слоя за достъп и се правят евентуално няколко промени, ако изобщо има такива. Ето защо за препоръчване е прозрачният режим на VTP.

Друга препоръчителна практика е ръчно да се отстраняват неизползваните VLAN от магистралните интерфейси.

## 2.2.2 Динамичен магистрален протокол (Dynamic Trunking Protocol)

Динамичният магистрален протокол (DTP) предоставя възможност на един порт да преговаря с отсрещния порт на дадена връзка за режима на работа на връзката, и когато е възможно тя автоматично да се преобразува в магистрална. При устройствата на Cisco съществуват 5 режима на работа на един порт:

- **Магистрален (Trunk, On):** Поставен в този режим портът остава постоянно в него. Протоколът не може да го промени автоматично. В този режим портът преговаря с отсрещния порт за да преобразува връзката в магистрална. В IOS този режим се нарича Trunk, а в операционната система на комутаторите CatOS - On.

- **Желателен (Desirable):** Портът активно се опитва да формира магистрална връзка при наличие на съгласие от отсрещния порт. Портът става магистрален, ако отсрещният порт е поставен в режим On, Desirable или Auto.

- **Авто (Auto):** В този режим портът е готов да превърне връзката в магистрална. Портът става магистрален, ако отсрещният порт е поставен в режим On или Desirable.

- **Достъп (Access):** Това е режимът на достъп на Cisco IOS, който уточнява, че портът никога не трябва да преминава в магистрален режим, дори ако отсрещния порт се опитва да го вкара в този режим. Портът постоянно остава в режим на достъп и преговаря с отсрещния порт връзката да не бъде магистрална.

- **Непреговарящ (Nonnegotiate):** В този режим портът не генерира DTP рамки. За да преобразувате връзката в магистрална връзка трябва отсрещния порт ръчно да бъде въведен в магистрален режим.

В устройствата на Cisco имаме три вида капсулиране за магистрална Ethernet връзка:

- **ISL:** Използва се капсулирането дефинирано в Inter-Switch Link протокола.

- **Dot1q:** Използва се капсулирането дефинирано в 802.1Q протокола.

- **Negotiate:** Определя, че портът преговаря с отсрещния порт да се използва ISL или 802.1Q протокол в зависимост от конфигурацията и възможностите на отсрещния порт. Както вече споменахме, ISL се разглежда като вече стар протокол и в много нови платформи не е имплементиран.

Магистралният режим, видът на използваното капсулиране и хардуерните възможности на два свързани LAN порта определят дали тази връзка ще стане ISL или 802.1Q магистрала.

Много често двата края на магистралната връзка се конфигурират в режим Desirable. Това в значителна степен опростява администрирането, с помощта на *show* командите получаваме точна индикация за функционирането на връзката, и в същност е обща препоръка за DTP.

Алтернативата е едната страна на връзката (обикновено в слоя за достъп) да се конфигурира в режим Auto, а другата страна на връзката (обикновено в разпределителния слой) да се конфигурира в режим Desirable. Такава настройка дава възможност да се формира автоматично магистрална връзка с работещ в нея DTP, което защитава от някои редки сценарии на хардуерен срив и неправилни софтуерни конфигурации.

Ако търсим по-бърза сходимост, можем да използваме трета конфигурация. Можем да конфигурираме DTP с On в двата края или с On и Nonnegotiate. С това спестяваме няколко секунди от престоя при възстановяването на дефектирала връзка или възел. В този случай обаче DTP не наблюдава активно състоянието на магистралната връзка и не е лесно идентифицирането на неправилна нейна конфигурация. Като пример за възможно използване на On с Nonnegotiate е когато връзката е между два различни VTP домейна. DTP включва името на домейна в своите съобщения, и когато имената на домейните не съвпадат, не е възможно да се изгради магистрална връзка в режим Desirable.

В крайна сметка препоръчителната практика при конфигурирането на магистрална връзка между комутаторите е да използвате протокола DTP с режим в двата края Desirable и капсулиране Negotiate.

## 2.3 Препоръчителни практики за конфигуриране на UDLD

UDLD позволява устройствата да наблюдават физическата конфигурация на кабелите и да откриват появата на еднопосочни връзки в случаите, когато не може да се осъществи двупосочна комуникация.

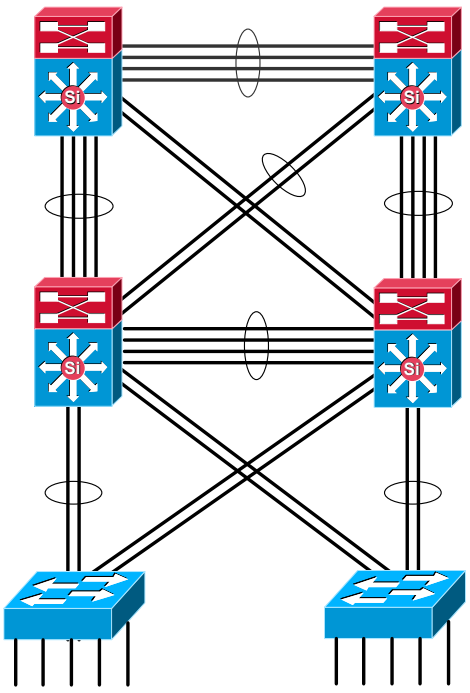
UDLD обикновено се използва при оптичните кабели, където могат да се появят физически несъответствия, например в двойките приемник/предавател. Протоколът поддържа и медни Ethernet кабели свързващи LAN портовете.

Всеки порт на комутатор, конфигуриран за UDLD, изпраща UDLD hello пакети в слой 2. Тези пакети съдържат идентификаторите на устройството и на порта, както и идентификаторите на отсрещните устройство и порт, така както се виждат те от UDLD протокола. Приемайки такъв пакет портът трябва да види в него идентификаторите на собственото си устройство и порт. Ако това не се случи за определен период от време, връзката се смята за еднопосочна и се прекъсва. Има два режима на UDLD – нормален и агресивен. При нормалния режим се деактивира само края на връзката, където UDLD е открил грешката. При агресивния режим се деактивират и двата края на връзката 8 секунди след като се установи, че тя не е двупосочна. И двата режима ползват едни и същи 15-секундни hello таймери.

Препоръчителната практика е да се разреши UDLD агресивния режим във всички среди, където се използват оптични междусистемни връзки. UDLD може да се разреши глобално, върху всички оптични портове, с командата на IOS *udld {enable | aggressive}*. За разрешаването на протокола в индивидуален LAN порт се използва командата *udld port [aggressive] interface*. Желателно е да разрешите протокола в глобален режим, за да не се налага да го разрешавате на всеки индивидуален оптичен интерфейс.

## 2.4 Препоръчителни практики за конфигуриране на EtherChannel

Протоколът EtherChannel обединява няколко (до осем) отделни Ethernet физически връзки в една логическа връзка. Както е показано на Фигура 12, той се използва най-често при свързването на устройствата от разпределителния слой с устройствата от ядрото или между отделните устройства в самото ядро, където се изисква увеличена достъпност и мащабируем трафик. Агрегирането на връзките се използва за осигуряване на редундантност и за избягване появата на единични точки на отказ. Протоколът също така намалява сложността на връзките, понеже използването на една логическа връзка намалява броя на съседите в слой 3 в сравнение със случая, когато използваме много паралелни връзки.



Фигура 12 Конфигурация с използване на EtherChannel

Да отбележим също така, че използването на EtherChannel води и до оптимизиране на протокола STP, понеже много физически връзки се третират като една и едновременно променят състоянията си.

EtherChannel създава канал от до 8 паралелни връзки между комутатори. Ако интерфейсите на отделните връзки се намират и в различни физически интерфейсни модули (cards), получаваме и увеличение на достъпността, тъй като повредата в един физически модул не води до пълна загуба на свързаност.

Разполагаме с два варианта на механизъм за управление на EtherChannel:

- Имплементацията на предварителния стандарт на Cisco, която използва Port Aggregation Protocol (PAgP) и

- Имплементацията на IEEE 802.3ad, която използва Link Aggregation Control Protocol (LACP).

Протоколите PAgP и LACP не си взаимодействат. Можете обаче ръчно да конфигурирате комутаторите с PAgP от едната страна на връзката и LACP от другата в режим On/On. По този начин се избягва диалога за договаряне на трафика между комутаторите. Тогава каналът е твърдо фиксиран.

Когато свързвате устройства на Cisco с различни операционни системи (IOS и CatOS) трябва да се убедите, че използваните настройки на PAgP са координирани. Настройките по подразбиране за различните операционни системи са различни. Когато свързваме две такива устройства, препоръчителната практика е протоколът PAgP в устройството с CatOS да бъде поставен в режим Off, когато няма да използваме EtherChannel. Когато ще използваме EtherChannel, и в двете устройства протоколът PAgP е хубаво да бъде поставен в режим Desirable.

Когато нямаме изрична нужда от него, агрегирането на портовете трябва да бъде забранено. Например то трябва да бъде забранено на портовете свързани с крайните потребители. Това можем да направим като използваме макросите *Set Port Host macro* на CatOS и *Switchport Host macro* на IOS. Горните макроси деактивират използването на тези портове в магистрални връзки и EtherChannel, като едновременно с това разрешават STP PortFast.

## 2.4.1 Port Aggregation Protocol (PAgP)

PAgP е един от механизмите за управление на EtherChannel. PAgP има четири режима свързани с автоматичното формиране на логически канал от много физически връзки:

- **On:** В този режим портовете се свързват безусловно в логически канал. Каналът реално функционира само когато всички негови портове на единия комутатор са в режим On и всички портове към които те са свързани на другия комутатор са в режим On. Няма трафик от преговори между портовете.

- **Desirable:** В този режим портът е поставен в активно състояние за преговори и започва да изпраща PAgP пакети за договаряне. Този режим не се поддържа когато членовете на групата на EtherChannel са от различни стекирани комутатори (cross-stack EtherChannel).

- **Auto:** В този режим портът е поставен в пасивно състояние за преговори. Той отговаря на PAgP пакетите които получава, но не започва по собствена инициатива да изпраща PAgP пакети за договаряне. Този режим не се поддържа когато

членовете на групата на EtherChannel са от различни стекирани комутатори (cross-stack EtherChannel).

- **Off:** Портът не се свързва в логически канал.

Както и при DTP, имаме дългогодишна практика за конфигуриране на EtherChannel с PAgP. От едната страна на връзката (обикновено в комутатора в слоя за достъп) портовете се поставят в режим Auto, а от другата страна на връзката (обикновено в комутатора в разпределителния слой) - в режим Desirable. Можем да поставим и двете страни в режим Desirable. Каналът в този случай става работоспособен при завършване на договарянето, но връзката с другата страна винаги е достъпна, дори когато каналът не е напълно установен. За EtherChannel в слой 2 конфигурирането в режим Desirable/Desirable е за препоръчване, тъй като тогава пакетите се предават по всички връзки и нарушаване на някоя от връзките няма да доведе до изменения в STP.

### 2.4.2 Link Aggregation Control Protocol

LACP е друг контролен механизъм за управление на EtherChannel. LACP има четири режима свързани с автоматичното формиране на логическия канал:

- **On:** В този режим портовете се свързват безусловно в логически канал. Каналът реално функционира само когато всички негови портове на единия комутатор са в режим On и всички портове към които те са свързани на другия комутатор са в режим On. Няма трафик от преговори между портовете.

- **Active** В този режим портът е поставен в активно състояние за преговори и започва да изпраща LACP пакети за договаряне.

- **Passive:** В този режим портът е поставен в пасивно състояние за преговори. Той отговаря на LACP пакетите които получава, но не започва по собствена инициатива да изпраща LACP пакети за договаряне.

- **Off:** Портът не се свързва в логически канал.

Препоръчителната практика е портовете от едната страна на връзката (обикновено в комутатора в слоя за достъп) да се поставят в режим Active, а от другата страна на връзката (обикновено в комутатора в разпределителния слой) - в режим Passive, или от двете страни да поставим портовете в режим Active. При такава конфигурация каналът става работоспособен след договарянето, но връзката с отдалечения комутатор е винаги достъпна, дори когато каналът не е напълно установен.

## 3. Литература


[1] Стоилов Емил, Проектиране на корпоративни мрежи. Част I Архитектура, Technical Report, Научен електронен архив на НБУ, 2014, http://eprints.nbu.bg/2219/

[2] Стоилов Емил, Общи архитектурни концепции използвани в IP маршрутизаторите, Technical Report. Научен електронен архив на НБУ, 2013. http://eprints.nbu.bg/1770/

[3] Cisco Catalyst 6500 Series with Cisco IOS Software Modularity, http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/prod_bulletin0900aecd80313e15.html

[4] Spanning Tree Protocol, http://www.cisco.com/c/en/us/tech/lan-switching/spanning-tree-protocol/index.html/index.html

[5] ISL and 802.1Q Trunking Between Catalyst Layer 2 Fixed Configuration Switches, http://www.cisco.com/c/en/us/support/docs/lan-switching/8021q/8758-43.html

[6] Understanding and Configuring the Unidirectional Link Detection Protocol Feature, http://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/10591-77.html

[7] Cisco EtherChannel, http://www.cisco.com/c/en/us/tech/lan-switching/etherchannel/index.html

[8] Cisco Networking Academy, Lab-Configuring Rapid PVST+, PortFast, and BPDU Guard, https://courses.cs.ut.ee/MTAT.08.004/2014_spring/uploads/Main/24_2.pdf

[9] Sean X Wang, bit and bytes: Understand Bridge Assurance, http://www.seanxwang.com/2010/06/understand-bridge-assurance.html