



pasja-informatyki.pl

Sieci komputerowe

Warstwa łączy danych

ARP, Ethernet

Damian Stelmach

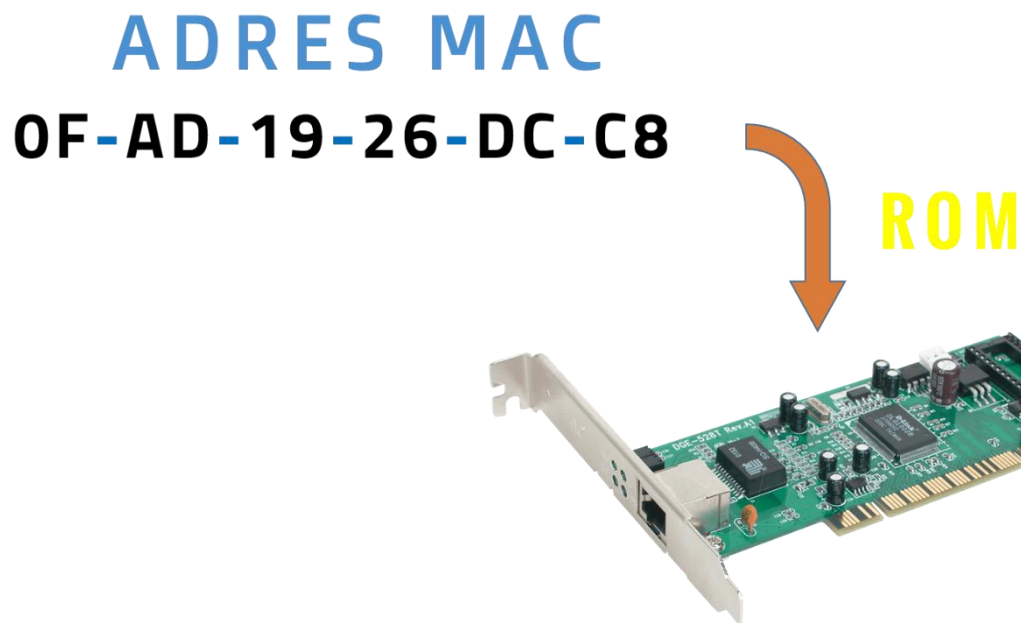
Spis treści

Zadania warstwy łącza danych.....	3
Ramka warstwy łącza danych i komunikacja.....	6
Protokół ARP.....	11
Ethernet.....	13

Główną i zasadniczą rolą warstwy łącza danych jest zapewnienie warstwom górnym dostępu do **medium transmisyjnego**. Dane, które wędrują w dół stosu, przechodząc przez poszczególne warstwy, muszą w pewnym momencie zostać dostarczone do **nośnika danych**, dzięki któremu dotrą do celu, czyli hosta odbierającego dane. To jest właśnie podstawowa funkcja warstwy łącza danych: **umieszcza w nośniku dane pochodzące z warstw wyższych**.



Omówiona w poprzednim odcinku kursu **warstwa sieciowa**, w procesie enkapsulacji opatrywała segmenty odebrane z warstwy transportowej **adresami IP** tworząc pakiety. Pakiety te, zanim zostaną wysłane do hosta docelowego trafiają do **warstwy łącza danych**, poprzez którą następnie przekazane zostają do **medium transmisyjnego**. Zanim to jednak nastąpi, pakiety opatrywane są kolejnymi informacjami sterującymi, tym razem są to **adresy fizyczne urządzeń**, czyli **48-bitowe adresy MAC**. Wówczas pakiety stają się **ramkami** i to właśnie te ramki trafiają dopiero do medium w celu ich dalszego przesłania do hosta docelowego. Adresy MAC nadawane są na **etapie produkcji** karty i zapisywane **pamięci ROM**. Pamięć ROM jest pamięcią **tylko do odczytu**, tak więc nie da się zmienić nadanego adresu na poziomie samej karty, na poziomie sprzętowym. Da się natomiast, taki adres zmienić na poziomie systemowym urządzenia, np. w **systemie operacyjnym**.



Sama warstwa łącza danych jest można powiedzieć **pośrednikiem** pomiędzy **mediami transmisyjnymi**, a **oprogramowaniem sieciowym**. W przypadku urządzeń końcowych, czyli komputerów, serwerów czy telefonów, jest to jedyna warstwa implementowana nie tylko w obszarze **programowym**, ale również w obszarze **sprzętowym**. Fizycznym odzwierciedleniem warstwy łącza danych są **karty sieciowe**, które mamy zainstalowane w naszych komputerach. Karty te stanowią interfejs pomiędzy oprogramowaniem sieciowym a medium transmisyjnym. W związku z tym, że warstwa łącza danych działa na dwóch płaszczyznach, na płaszczyźnie sprzętowej i programowej, jej funkcje i zadania również podzielone zostały na dwie mniejsze podwarstwy:

- **LLC** (ang. Logical Link Control),
- **MAC** (ang. Media Access Control).

Podwarstwa **LLC** umieszcza w ramach informacje o **stosowanym protokole warstwy sieci**, dzięki czemu możliwe jest korzystanie z tego samego medium transmisyjnego i karty sieciowej dla różnych protokołów warstwy sieci takich jak IPv4, IPv6 czy IPX, jej funkcje w komputerach pełnią **sterowniki kart sieciowych**. Podwarstwa **MAC** natomiast, określa **zasady dostępu do medium**, i **wykonuje funkcje adresowania**.

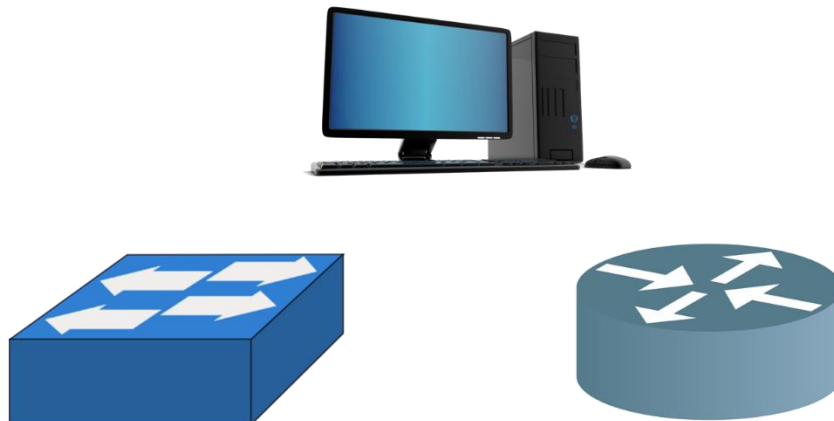
O metodach MAC mowa była w pierwszym odcinku z serii!

Podsumowując warstwa łącza danych:

- odbiera dane z warstwy sieci,

- tworzy ramki możliwe do przesłania przez medium,
- nadaje ramką adresy fizyczne,
- jest ona odpowiedzialna za kontrolę dostępu do medium.

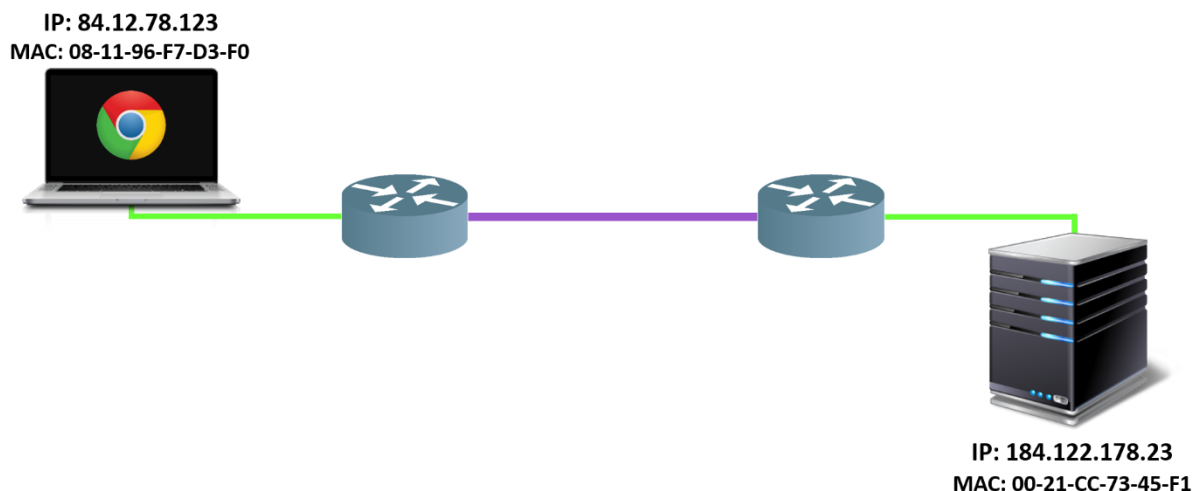
Implementacje tej warstwy znajdziemy na urządzeniach końcowych, takich jak komputery, ale również w ruterach i przełącznikach.



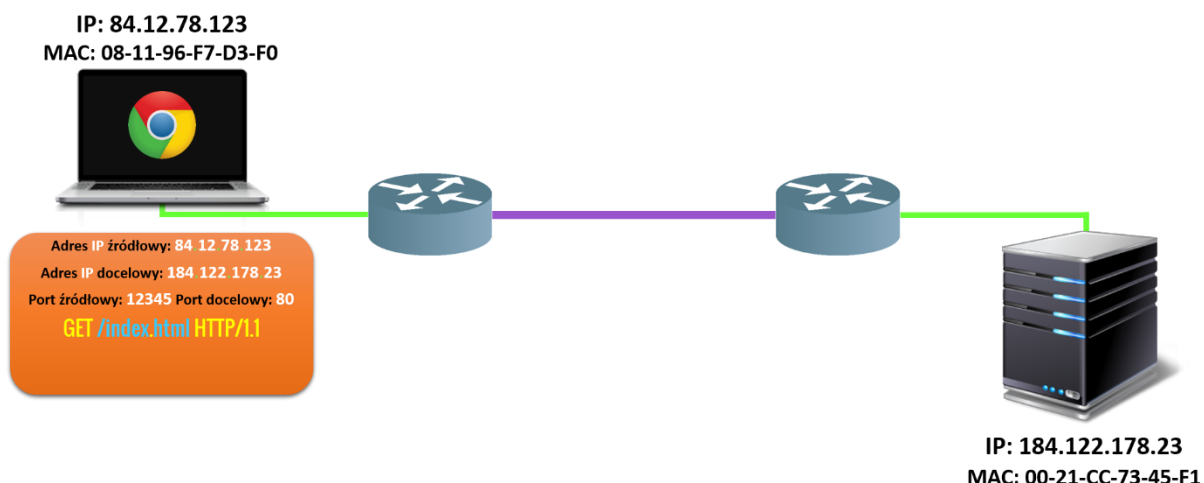
Istnieje wiele rozwiązań i wiele standardów sieciowych realizujących funkcje warstwy drugiej. Mamy standard **Ethernet**, mamy **sieci bezprzewodowe**, mamy w końcu wiele protokołów sieciowych działających w sieciach WAN, takich jak, chociażby protokół **Frame Relay**. Dlatego też nie istnieje coś takiego jak **uniwersalna ramka**. Każdy standard sieciowy dysponuje swoją ramką, która specyficzna jest dla jednego, konkretnego rozwiązania. Uogólniając temat, możemy przyjąć, że typowa ramka warstwy drugiej składa się z 3 głównych części:

Nagłówek	Dane	Stopka
<p>adresy MAC źródłowy i docelowy</p> <p>sygnał początku ramki</p>	<p>pakiety warstwy sieciowej / internetowej</p>	<p>sygnał końca ramki</p> <p>suma kontrolna</p>

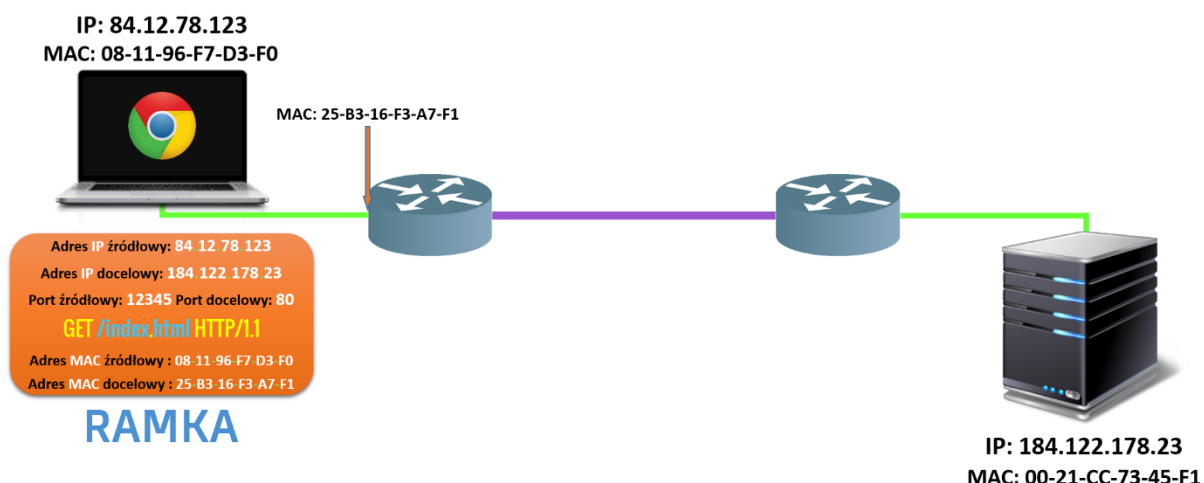
Prześledźmy teraz **proces komunikacji** pomiędzy urządzeniami, skupiając się na **funkcjach warstwy łącza danych**. Przyjmijmy, że nasz **komputer**, wysyła zapytanie do serwera **WWW** znajdującego się w odległej sieci.



Dane wysyłające takie zapytanie, wcześniej zostały już w **procesie enkapsulacji**, wyposażone w **numery portów aplikacji**, a także w **adresy logiczne**, czyli **adresy IP** komputera oraz serwera, no i stały się **pakiety**.

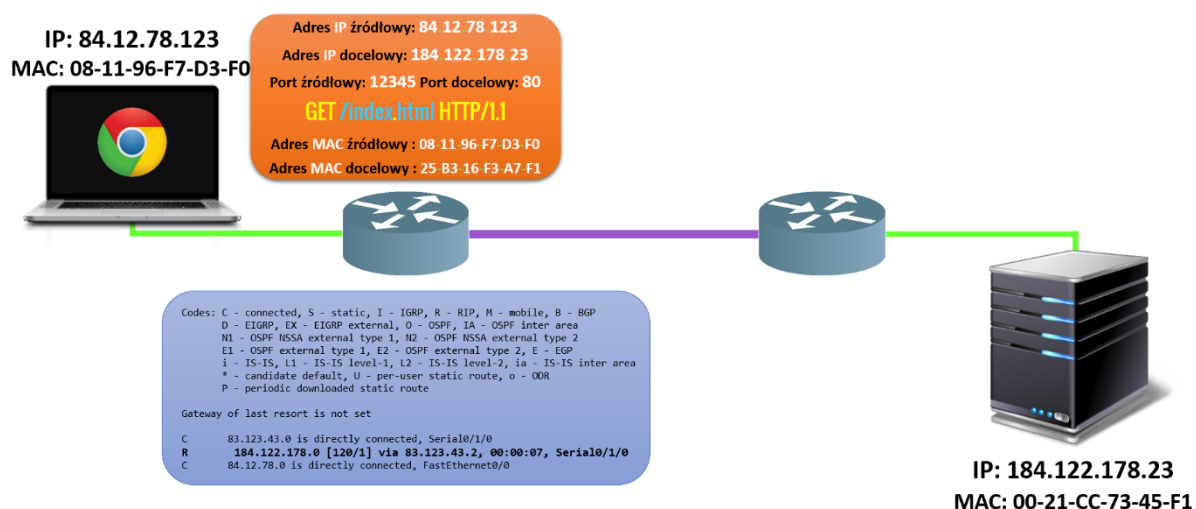


Zanim pakiet trafi do medium transmisyjnego, **warstwa łącza danych** musi utworzyć **ramkę** z odpowiednimi **adresami MAC nadawcy i odbiorcy ramki**. W przypadku adresu MAC **nadawcy** sprawa jest oczywista, będzie to po prostu **adres MAC** komputera, ale co z **adresem hosta docelowego**? **Komputer**, jeśli nie jest w tej samej sieci, co **serwer WWW**, nie jest w stanie dowiedzieć się jaki **adres MAC** ma jego karta sieciowa, nie ma takiej możliwości, jest to technicznie niewykonalne. Dlaczego? No dlatego, że **adresy MAC** służą do komunikacji **tylko w obrębie jednej, danej sieci i nie wychodzą poza jej obszar**. W związku w polu ramki zawierającej **docelowy adres MAC** zapisany zostanie **adres MAC interfejsu rutera**, do którego podłączony jest nasz komputer.

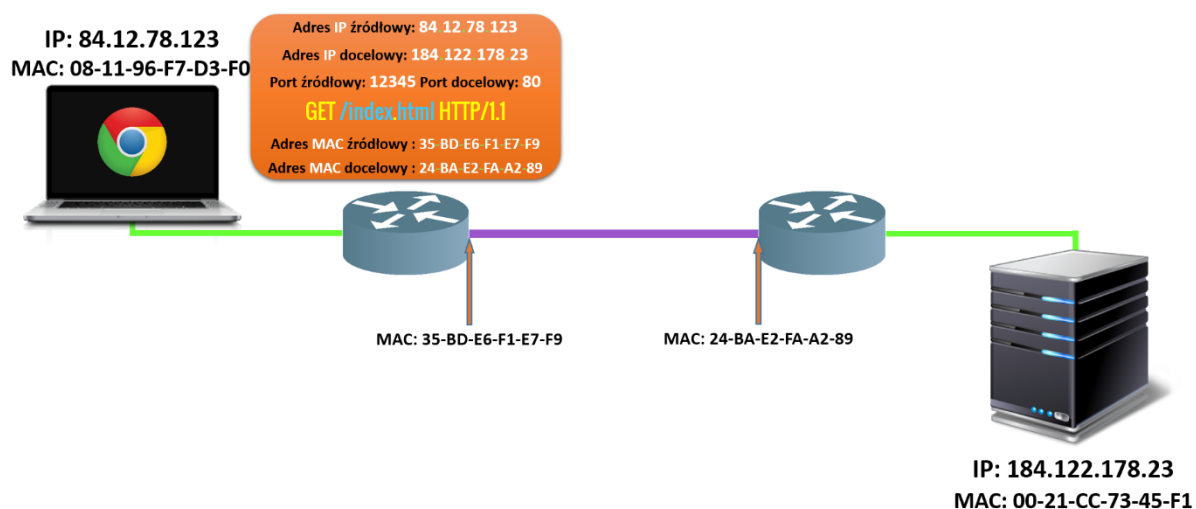


Ramka zostaje wysłana poprzez medium transmisyjne właśnie do pierwszego rutera. Ten następnie, po otrzymaniu ramki dokona jej **dekapsulacji**, po to, aby móc odczytać **adres IP urządzenia**, do którego ma trafić pakiet. **Adresu IP nie może odczytać bezpośrednio z ramki warstwy drugiej, stąd ta dekapulacja**. Po odczytaniu **adresu IP z pakietu** (po dekapulacji ramki, dane znowu stają się

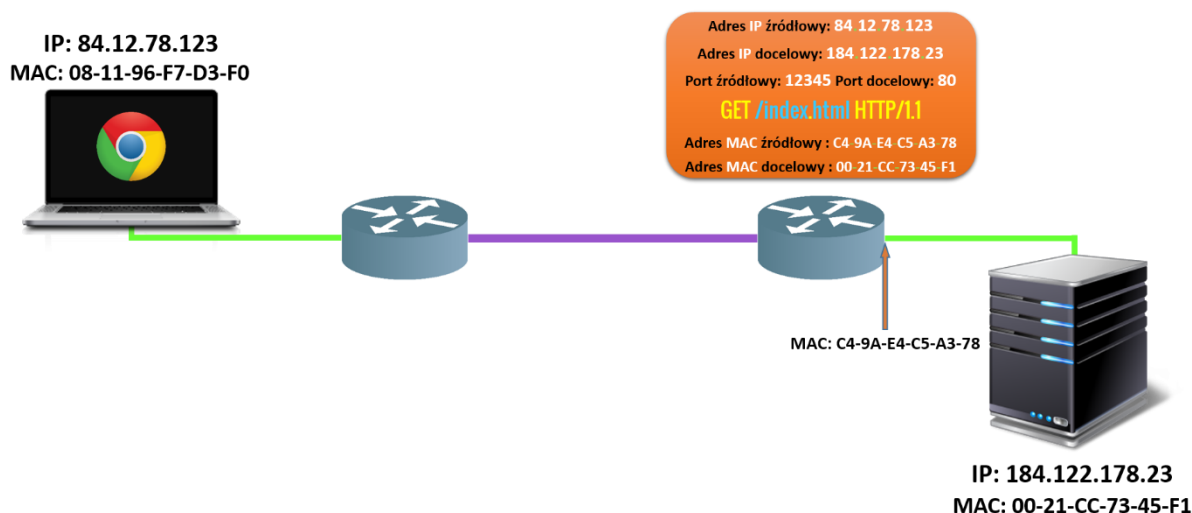
pakiem), porówna go ze swoimi wpisami w **tablicy routingu** i odnajdzie wpis informujący, że do sieci, w której pracuje serwer, prowadzi trasa, przez ten drugi ruter.



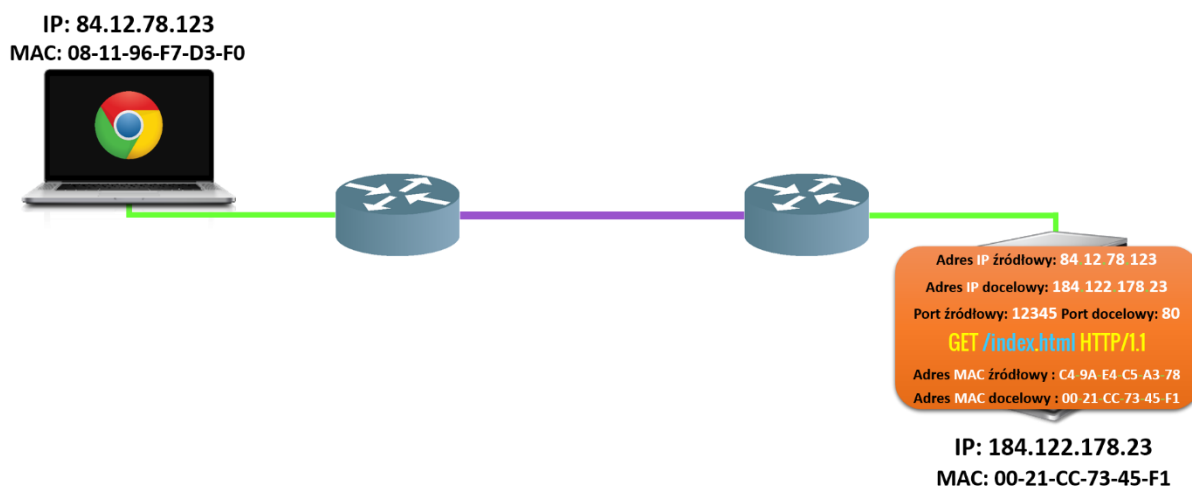
Wówczas utworzy **nową ramkę**, w której **adresem źródłowym** będzie już **adres MAC** jego interfejsu, poprzez który łączy się z drugim ruterem, a **docelowym**, **adres MAC** tego właśnie rutera.



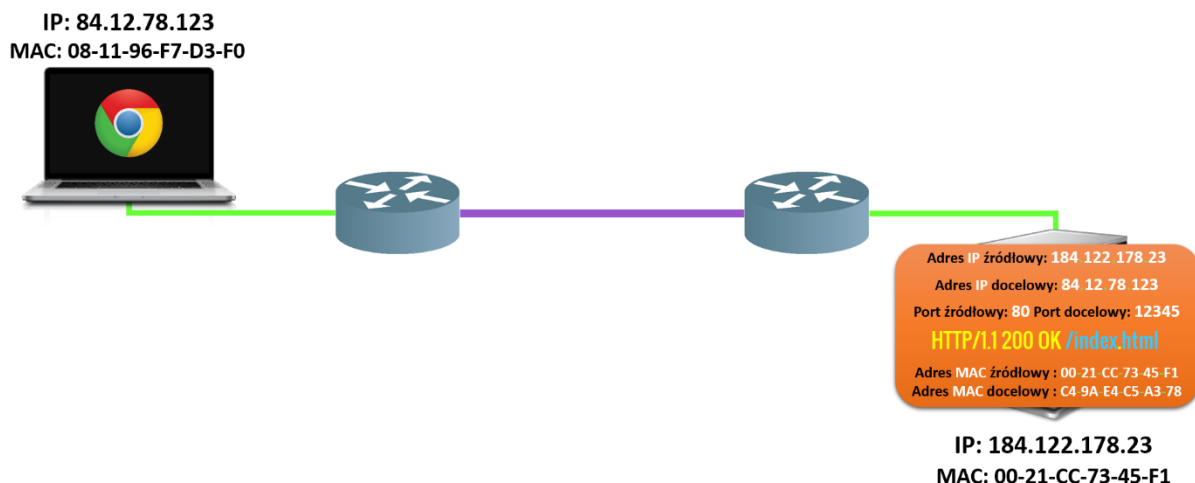
Ramka następnie trafia poprzez medium do **drugiego rutera**, który ponownie dokonuje **dekapsulacji ramki**, aby odczytać **adres IP** z pakietu. Stwierdza, że adresatem danych, jest urządzenie, które pracuje w sieci, bezpośrednio do niego podłączonej, tak więc ponownie następuje proces **enkapsulacji**, realizowany przez drugi ruter, tym razem w polach **adresów MAC** umieszcza on **adres MAC swojego drugiego interfejsu** jako adres źródłowy, oraz **adres MAC serwera** jako **adres docelowy**.



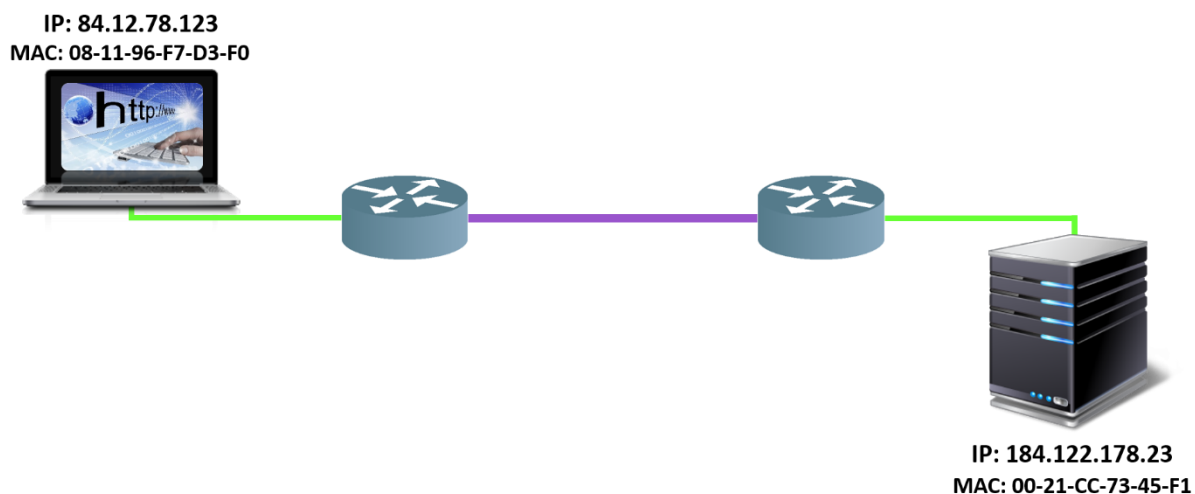
Ramka w ten sposób przygotowana przesyłana jest do **serwera**, który również dokonuje jej **dekapsulacji**. Tym razem jednak jest to urządzenie, do którego są **kierowane dane**, tak więc dokonuje **pełnej ich dekapsulacji**, czyli odczytuje dodatkowo **numery portów aplikacji**, tak aby wysłać danej do właściwej, konkretnej aplikacji, w tym przypadku do **usługi WWW**.



Usługa WWW przygotowuje następnie **dane odpowiedzi**. Dane trafiają najpierw do **warstwy transportowej**, gdzie nadawane są **numery portów aplikacji**, następnie do **warstwy sieciowej** gdzie tworzone są **pakiety** z odpowiednimi **adresami IP**, a na koniec do **warstwy łącza danych**, która przygotowuje z pakietów **ramki**, oznaczone **adresami MAC serwera i rutera**, do którego serwer jest podłączony.

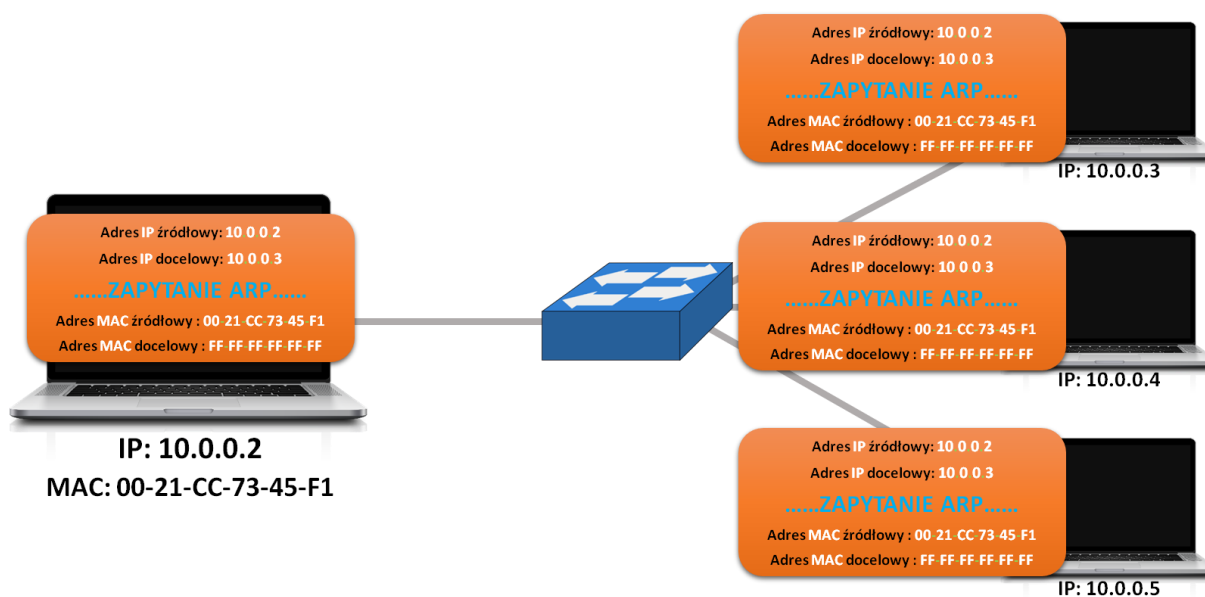


Odpowiedź następnie przekazywana jest do **nośnika** i następuje proces **przesyłania jej do klienta**. Po drodze, przechodzi przez **dwa routery**, które wykonują procesy **dekapsulacji** i ponownej **enkapsulacji**, no bo muszą odczytywać **adresy IP**, dzięki którym mogą przesyłać odpowiedź dalej. Na koniec odpowiedź trafia do **klienta**. Ten dokonuje **dekapsulacji danych**, co w konsekwencji pozwala za pomocą przeglądarki wyświetlić **stronę WWW**.

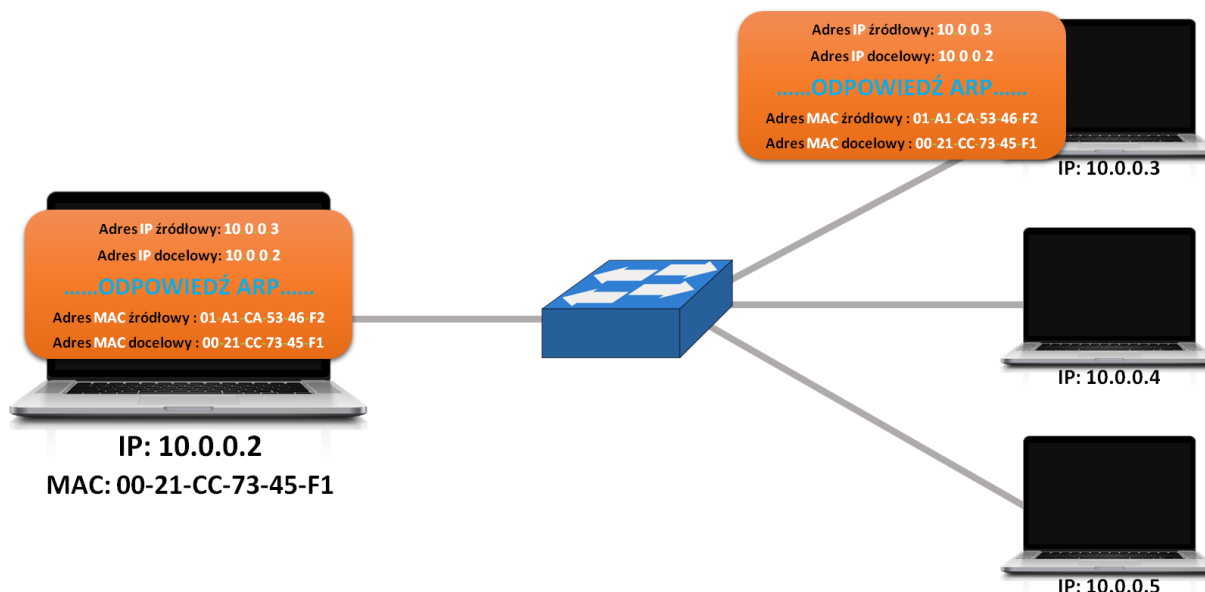


Kiedy jako **użytkownicy sieci**, wysyłamy dane z **jednego urządzenia do drugiego** to albo znamy jego **adres IP**, albo **nazwę domenową**. Gorzej jest już z **adresami MAC**, na ich podstawie, my użytkownicy sieci, nie definiujemy odbiorców danych, to dzieje się poza nami. W sieciach komputerowych, **opartych na protokole IPv4**, do uzyskiwania informacji o **adresie MAC** danego urządzenia służy protokół zwany **ARP** (ang. Address Resolution Protocol).

ARP to mechanizm pozwalający na **odzworowanie adresu logicznego**, czyli **IP** na **adres fizyczny** czyli **MAC**. Załóżmy, że komputer chcąc przesłać dane do innego urządzenia zna jego **adres IP**, ale nie zna **adresu MAC**. Aby ten adres poznać, **komputer będący nadawcą danych**, zanim te konkretne dane wyśle, tworzy **rozgłoszeniową ramkę ARP**, która rozsyłana jest do wszystkich urządzeń w tej samej sieci. W polu adresu źródłowego takiej ramki zapisywany jest **adres komputera**, który przygotował taką ramkę, a w polu adresu docelowego, **rozgłoszeniowy adres MAC: FF-FF-FF-FF-FF-FF**.



Każde z urządzeń, które odbierze **ramkę**, **dekapsuluje ją do postaci pakietu** i sprawdza, czy w polu docelowym **adres IP** jest jego adres. Jeśli w polu docelowym adres IP będzie inny adres niż jego, to zignoruje pakiet, jeśli natomiast to jego **adres IP**, **utworzy nową ramkę**, w której zapisany będzie **jego adres MAC** i przekaże ją do przesłania.



Teraz już komputer, **który wysłał rozgłoszeniową ramkę** wie jaki **adres fizyczny** ma urządzenie, z którym chce się skomunikować i taką komunikację może rozpocząć. Informacje o odwzorowaniu **adresu IP** na **adres MAC** zapisywane są w **tablicy ARP** każdego urządzenia, tak aby można je było wykorzystać w późniejszym czasie. Domyślnie, w systemach Windows wpis taki utrzymuje się maksymalnie do **10 minut**, po tym czasie zostaje usunięty. Aby wyświetlić tablicę ARP, należy w konsoli wykonać polecenie **arp -a**. Jak widać znajdują się tutaj jakieś wpisy, co oznacza, że w **ciągu ostatnich 10 minut** odbywała się komunikacja pomiędzy moim komputerem a innym urządzeniem.

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Wszelkie prawa zastrzeżone.

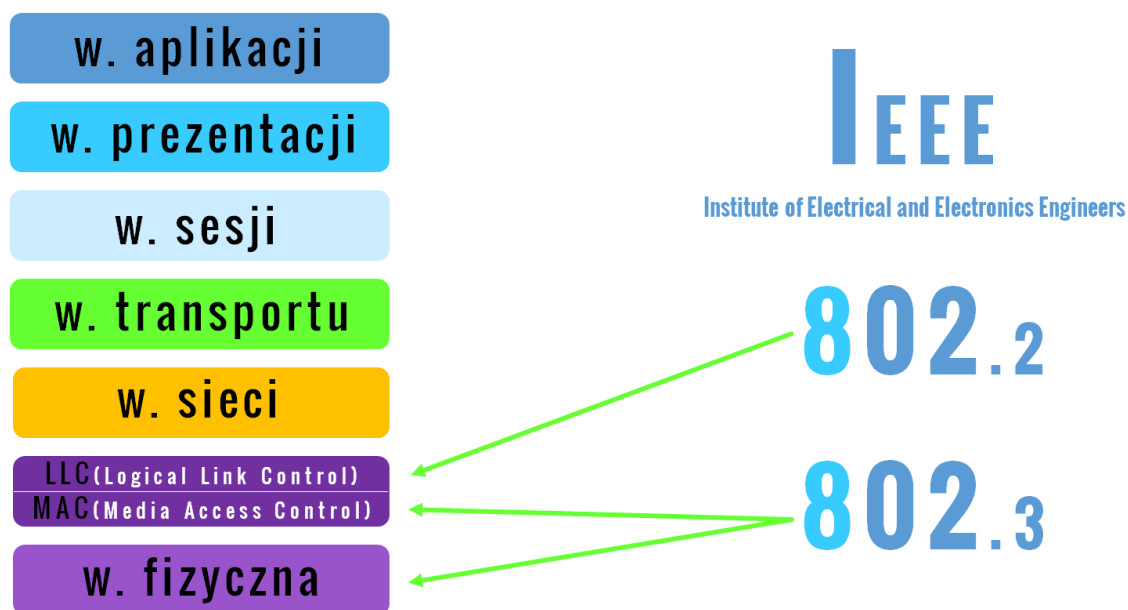
C:\Users\damian>arp -a

Interface: 192.168.0.103 --- 0x12
Internet Address      Physical Address      Type
5.5.5.5               a3-3e-51-45-e1-e2    static
192.168.0.1           64-66-b3-5b-ae-3a    dynamic
192.168.0.100         08-11-96-f7-d3-f0    dynamic
192.168.0.102         e8-5b-5b-3f-fe-24    dynamic
192.168.0.255         ff-ff-ff-ff-ff-ff    static
224.0.0.2             01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
224.0.0.253           01-00-5e-00-00-fd    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\damian>
    
```

Początki pracy nad tym standardem to odległe **lata 70**. Firma **Xerox** będąca wówczas jedną z większych firm technologicznych obrała sobie za cel zaprojektowanie **otwartego standardu komunikacji sieciowej**, który będzie służył ludziom przez wiele lat. Pod koniec lat 70 opracowała standard lokalnych sieci komputerowych, który stał się pierwowzorem **Ethernetu**. Obecnie Ethernet to standard, który spotkamy w **większości lokalnych sieci komputerowych** na świecie, a ponadto dzięki wielu swoim zaletom stał się również standardem **sieci miejski**, a w niektórych przypadkach nawet **rozległych**.

Ethernet to cały zbiór **rozwiązań sieciowych**, które implementowane są zarówno w **warstwie łącza danych**, jak również w **warstwie fizycznej**. Pieczę nad rozwojem tej technologii sprawuje obecnie organizacja **IEEE** (ang. Institute of Electrical and Electronics Engineers), która w 1985 roku opublikowała jej standardy i opisała je numerem **802.2** oraz **802.3**. Standard **802.2** odnosi się do funkcji związanych z podwarstwą **LLC**, ten drugi natomiast do podwarstwy **MAC** oraz do **warstwy fizycznej** modelu OSI.



Na sukces rozwiązań opartych na standardzie **Ethernet** składa się wiele czynników, m. in.:

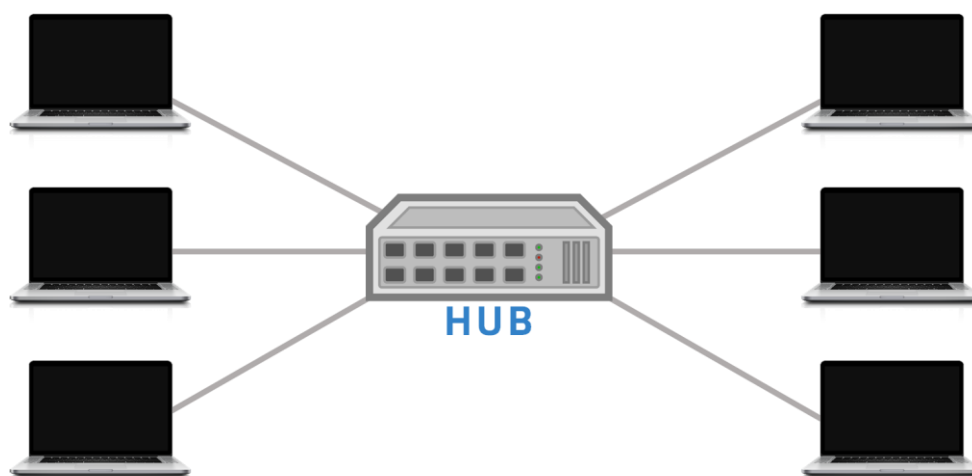
- łatwość implementacji,
- niezawodność,
- zdolność do przyjmowania nowych technologii,
- stosunkowo niewielki koszt implementacji.

Rozwój Ethernetu

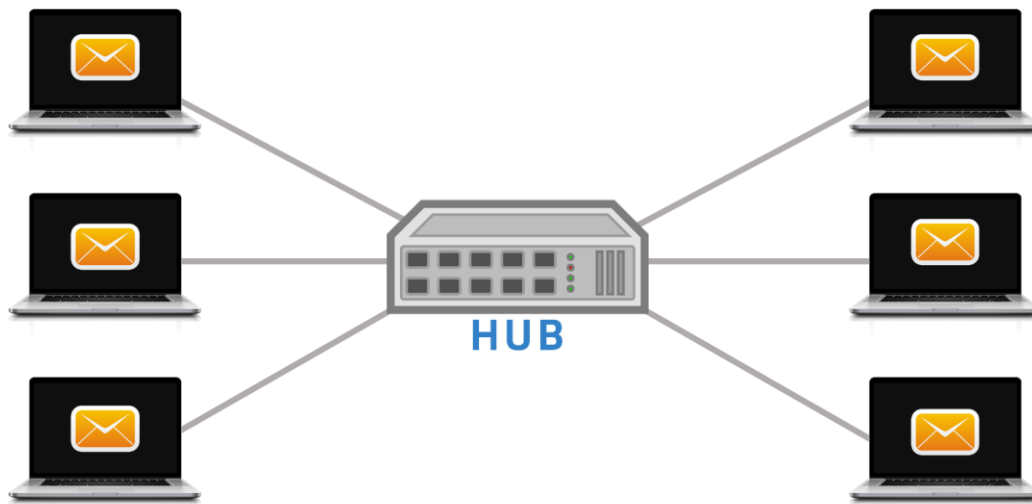
Początkowe wersje Ethernetu zwane **Thicknet** (tzw. **gruby Ethernet**) oraz **Thinnet** (tzw. **cieńki Ethernet**) oferowały niewiele możliwości względem tego czym dysponujemy obecnie. Stare wersje pracowały w oparciu o miedziane medium transmisyjne jakim był **kabel koncentryczny**. Stosowały one fizyczną **topologię magistrali**, charakteryzującą się tym, że wszystkie urządzenia podłączone są do wspólnego medium. Takie rozwiązanie wymagało **sterowania dostępem do nośnika**, które realizowane było przez metodę **CSMA/CD**



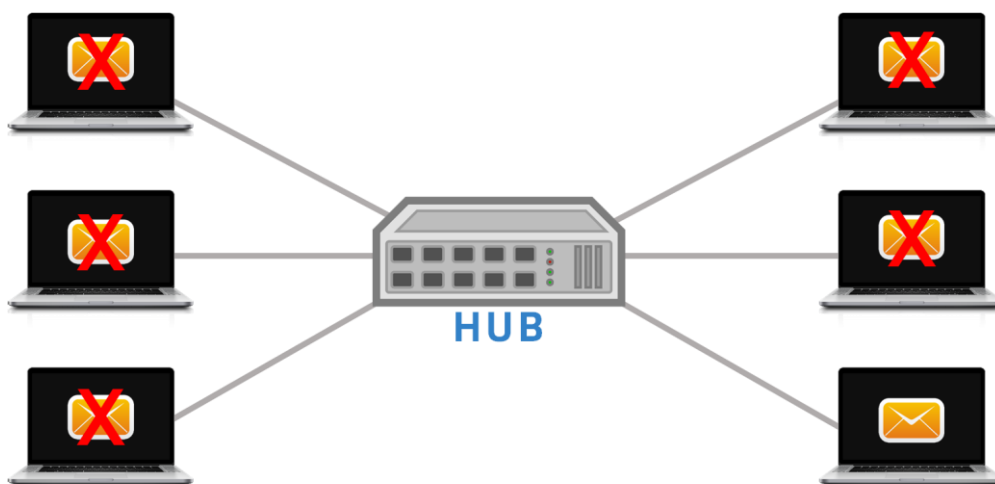
Po latach stosowania rozwiązania opartego na **topologii magistrali i koncentryka** jako medium transmisyjne okazało się, że to rozwiązanie nie jest już wystarczająco efektywne. Szybki rozwój sieci spowodował **zwiększenie wymagań użytkowników** co do jej przepustowości i niezawodności. Zamiast **kabla koncentrycznego** powszechnie zaczęto używać **kabla typu skrętka**, **kabla UTP**, oraz nowej topologii. Pojawiła się **topologia gwiazdy**, ta sama, która stosowana jest obecnie, jednak zamiast przełączników jako centralnych punktów sieci stosowano **koncentratory** (ang. **HUB**). O przełącznikach jeszcze wówczas nikt nie słyszał.



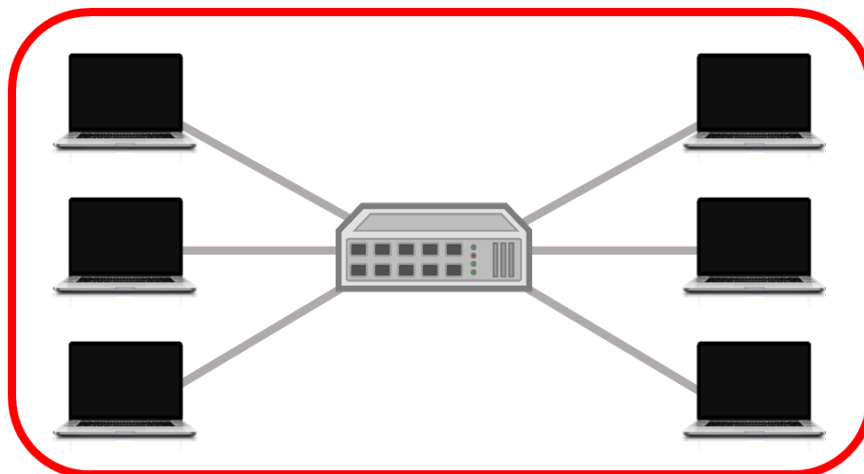
Zastosowanie koncentratorów, usprawniło w pewnym stopniu działania sieci komputerowych, ale szybko okazało się, że i to rozwiązanie **nie jest idealne**. Podstawową cechą koncentratora jest to, że **wysyła on dane do wszystkich urządzeń**, które są do niego podłączone. Działa to tak, że komputer, który chce przesłać dane do **innego urządzenia**, realizując to komunikację za pośrednictwem koncentratora. Ten z kolei, nie jest tak inteligentny, aby dostarczyć dane to właściwego urządzenia tylko **wysyła je do wszystkich**, które są do niego podłączone.



Dopiero urządzenia, do których trafiły dane **decydują czy są ich adresatem czy nie**, poprzez analizę adresacji. Jeśli **nie są adresatami**, ignorują dane, jeśli są natomiast, no to je **interpretują**.



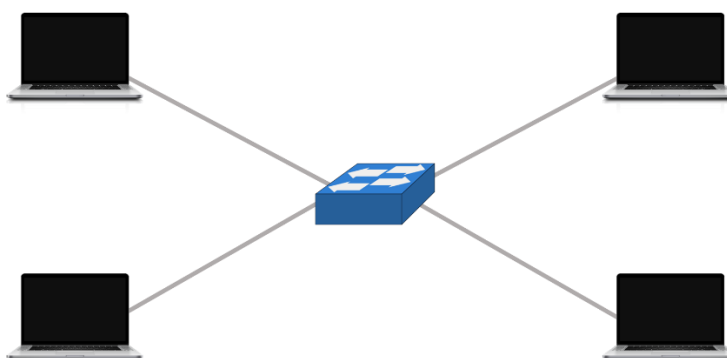
Tego typu rozwiązanie powodowało, że chociaż **fizyczną topologią** była **topologii gwiazdy**, to logicznie dalej było to podobne rozwiązanie do tego stosowanego w **poprzedniej generacji Ethernetu**. Również tutaj stosowano metodę dostępu do łącza opartą o **CSMA/CD** co przy szybkim rozwoju sieci stało się nieefektywne. Ponadto każdy koncentrator tworzył tzw. **domenę kolizyjną**.



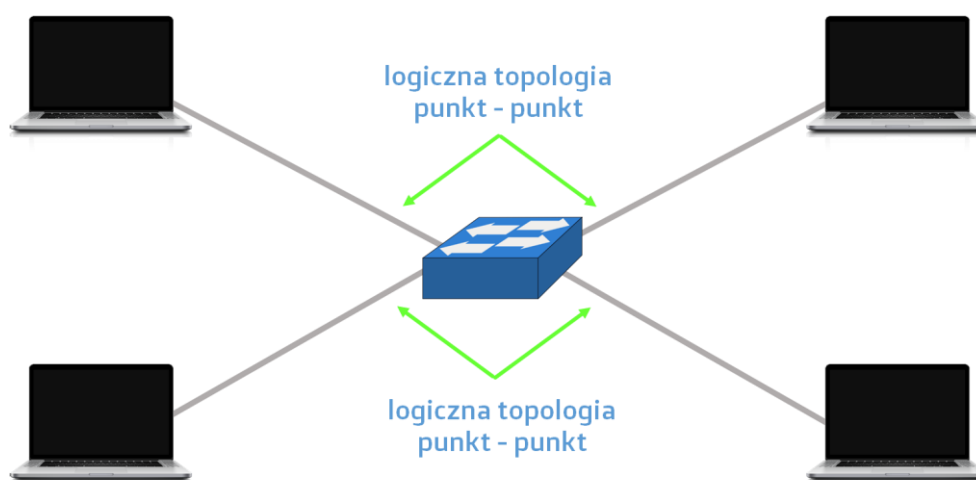
domena kolizyjna

Im więcej urządzeń podłączonych do koncentratora tym większa domena kolizyjna, a czym większa domena kolizyjna tym większe prawdopodobieństwo wystąpienia kolizji, co w konsekwencji ogranicza przepustowość i generuje częstą konieczność retransmisji danych. Większa ilość kolizji to nie jedyny problem związany z zastosowaniem koncentratorów. Do innych wad takich urządzeń zaliczyć musimy jeszcze **ograniczoną skalowalność**, a także **zwiększone opóźnienie** w dostarczaniu danych, powodowane m.in. tymi kolizjami.

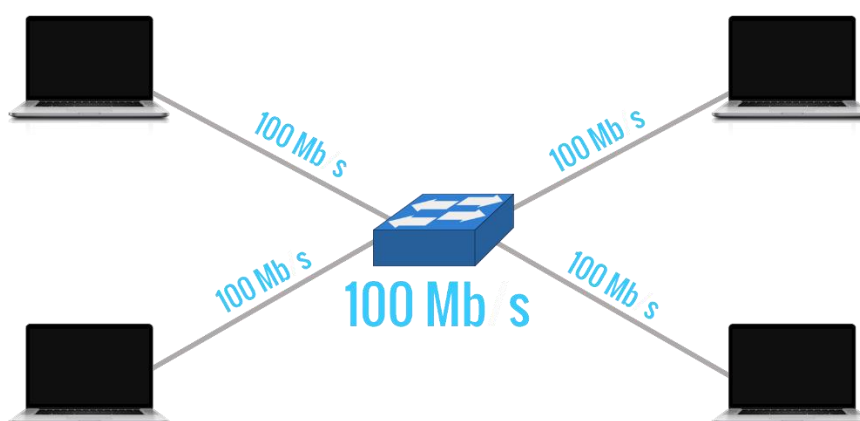
Tak mijały kolejne lata, trwały prace nad wyeliminowaniem **słabych stron Ethernetu** opartego na koncentratorach, aż wymyślono inteligentne urządzenie sieciowe zwane **przełącznikiem** (ang. **SWITCH**), który rozwiązywał problemy trapiące wcześniejsze wersje Ethernetu.



Przełączniki pracują w sieciach komputerowych do dziś i nic nie wskazuje na to, że szybko się to zmieni. Skąd taka popularność tych urządzeń i dlaczego są takie inteligentne? No w odróżnieniu od koncentratora, przełącznik **nie wysyła danych do wszystkich urządzeń do niego podłączonych**, ale tylko do **jednego, konkretnego**, do którego te dane są adresowane, pomijając oczywiście transmisję rozgłoszeniową, takie jak chociażby omówione wcześniej rozgłaszanie ARP. Pomiędzy portem przełącznika, do którego podłączone jest urządzenie, a samym urządzeniem występuje **logiczna topologia punkt - punkt**. Dane, które są adresowane do konkretnego urządzenia, **trafiają do niego i tylko do niego**.



Zastosowanie przełączników **praktycznie w całości eliminuje ryzyko wystąpienia kolizji**, gdyż urządzenia nie muszą rywalizować ze sobą o dostęp do medium. Jednocześnie **ogranicza się wielkość domeny kolizyjnej**, bo wówczas na taką domenę składa się tylko port przełącznika i urządzenie do niego podłączone. Zalet przełączników jest o **wiele więcej**. Każde urządzenie podłączone do portu przełącznika ma do dyspozycji **dedykowaną przepustowość**. Jeśli przykładowo przełącznik oferuje transmisję z szybkością **100 Mb/s**, to taką przepustowość będzie miało do dyspozycji **każde urządzenie do niego podłączone**.



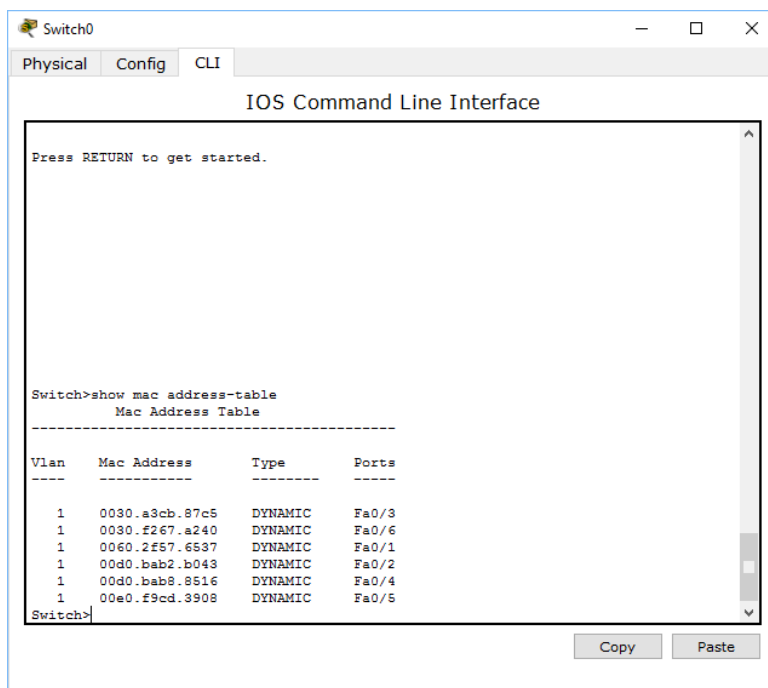
W przypadku **koncentratorów** przepustowość ta **dzielona była na wszystkie urządzenia**. Dzięki wykorzystaniu przełączników możliwa stała się też transmisja danych w trybie **pełnego duplexu**, co oznacza, że urządzenie do niego podłączone może jednocześnie odbierać dane i je wysyłać.

Obecnie stosowanych jest kilka, wersji **standardu Ethernet** wykorzystujących przełączniki. Najpopularniejszy z nich to standard oferujący nominalną przepustowość sięgającą **100 Mb/s**, zwany standardem **Fast Ethernet**. Transmisja w tym standardzie odbywa się tylko poprzez **2 pary żył miedzianych**, a nie 4, które dostępne są w skrętce. To powszechne rozwiązanie, stosowane w wielu sieciach komputerowych. W większości przypadków spełnia ono wymagania stawiane sieciom komputerowym.

Kiedy **zapotrzebowanie na przepustowość sieci wzrasta** wraz z ilością przesyłanych danych wówczas można zastosować standard **Gigabit Ethernet**. Nominalnie oferuje przepustowość rzędu jednego **1 Gb/s**. Jeśli wykorzystywany jest standard **1000BASE-T**, to wówczas do transmisji wykorzystuje się **wszystkie pary żył miedzianych w skrętce**. Duże sieci lokalne, w których wykorzystywana jest **telefonia internetowa VoIP**, a także przesyłane są duże ilości różnego typu **multimediów** stosują właśnie tą wersję Ethernetu. Przy wykorzystaniu standardu Ethernet istnieje również możliwość przesyłania danych za pomocą **łączy światłowodowych**. Wówczas gigabitowy standard Ethernetu nazywa się **1000BASE-SX** lub **LX**. Istnieją również standardy Ethernetu oferujące komunikację z przepustowością sięgającą **10**, a nawet **100 Gb/s**. Są one stosowane głównie w **sieciach miejskich i rozległych**, gdyż ich implementacja jest bardzo, ale to bardzo kosztowna i mało kogo stać na stosowanie tego typu rozwiązań w sieciach lokalnych. Poniższa tabela zawiera spis najpopularniejszych wersji standardu Ethernet, wraz z wykorzystywanymi przez nie mediami transmisyjnymi:

Standard Ethernet	Maksymalna przepustowość	Stosowane medium transmisyjne	Maksymalna odległość
100BASE-TX (fastEthernet)	100 Mb/s	UTP (kat. 5/5e)	100 metrów
100BASE-FX (fastEthernet)	100 Mb/s	Światłowód jednomodowy/wielomodowy	400/2 000 metrów
1000BASE-T (gigabitEthernet)	1 Gb/s	UTP (kat. 5e)	100 metrów
1000BASE-TX (gigabitEthernet)	1 Gb/s	UTP (kat. 6)	100 metrów
1000BASE-SX (gigabitEthernet)	1 Gb/s	Światłowód wielomodowy	550 metrów
1000BASE-LX (gigabitEthernet)	1 Gb/s	Światłowód jednomodowy	2000 metrów
10GBASE-T (10gigabitEthernet)	10 Gb/s	UTP (kat. 6/7)	100 metrów
10GBASE-LX4 (10gigabitEthernet)	10 Gb/s	Światłowód jednomodowy/wielomodowy	300/10 000 metrów

Wspomniane wcześniej przełączniki **stosują adresy MAC** do przesyłania danych pomiędzy urządzeniami podłączonymi do **portów przełącznika**. Każdy przełącznik posiada coś takiego co nazywa się **tablicą MAC adresów**. Jest to nic innego jak zbiór informacji określających jakie urządzenie, a właściwie **jaki MAC adres urządzenia** podłączony jest do konkretnego portu. Zrzut z przykładowej tablicy adresów MAC, dla przełącznika CISCO widzicie poniżej:



The screenshot shows a Cisco Switch CLI window titled "Switch0". The "CLI" tab is selected. The prompt is "Switch>". The command "show mac address-table" has been entered, and the output is displayed as follows:

```
Switch>show mac address-table
Mac Address Table
-----
Vlan    Mac Address      Type      Ports
-----
1       0030.a3cb.87c5    DYNAMIC   Fa0/3
1       0030.f267.a240    DYNAMIC   Fa0/6
1       0060.2f57.6537    DYNAMIC   Fa0/1
1       00d0.bab2.b043    DYNAMIC   Fa0/2
1       00d0.bab8.8516    DYNAMIC   Fa0/4
1       00e0.f9cd.3908    DYNAMIC   Fa0/5
```

At the bottom of the window, there are "Copy" and "Paste" buttons.

Wpisy w takiej tablicy dodane zostały dynamicznie, a nie przez administratora. Przełącznik zdobywa informacje zapisane w tablicy w procesie uczenia się. Z odebranej ramki, przełącznik odczytuje źródłowy adres MAC i taki dopisuje do swojej tablicy, przypisując jednocześnie numer portu na którym taką ramkę odebrał. Jeśli z kolei, nie wie do kogo dalej wysłać taką ramkę, bo nie ma wpisu w tablicy dotyczącego adresu MAC odbiorcy wówczas następuje proces zwany zalewaniem. Można go porównać do rozgłaszania, ponieważ ramka przesyłana jest do wszystkich urządzeń, za wyjątkiem nadawcy. Urządzenia, do których ramka adresowana nie jest, odrzucają ją, natomiast urządzenie będące adresatem, odpowiada i przesyła ramkę do przełącznika. Przełącznik odczyta z ramki adres MAC nadawcy, i zapisze go w swojej tablicy. Cały proces **uczenia się i zalewania** pokazany został w tutorialu.

Skoro standard Ethernet pracuje w **drugiej warstwie modelu OSI**, no to już pewnie się domyślacie, że również on tworzy **swoje ramki**. Oczywiście tak jest, Ethernet w procesie enkapsulacji tworzy własną ramkę, zwaną **ramką Ethernetową**. Przykładową ramkę widzicie poniżej:

Rozmiar pola w bajtach	7	1	6	6	2	46-1500	4
Nazwa pola	Preambuła	Znacznik początku ramki	Adres MAC odbiorcy	Adres MAC nadawcy	Długość/Typ	Dane i wypełnienie	Kod kontrolny ramki (FCS)

- **Preambuła** oraz **znacznik początku ramki** – te pola służą do poinformowania urządzenia docelowego, aby przygotował się na odbiór ramki;
- **Docelowy adres MAC**, czyli adres fizyczny odbiorcy ramki;
- **Źródłowy adres MAC**, czyli adres fizyczny hosta wysyłającego dane;
- **Długość / Typ** - pole **długość** określa wielkość ramki, natomiast **typ** określa, jaki został wykorzystany **protokół warstwy wyższej**, najczęściej jest to **IPv4**;
- **Dane** – to pakiet, który odebrany został z **warstwy sieciowej**. Minimalna wielkość tego pola to **46**, a maksymalna **1500 bajtów**. Jeśli pakiet jest mniejszy niż **46 bajtów**, to dopełnia się go losowymi danymi, tak aby rozmiar całej ramki został zwiększony do wymaganego minimum, czyli do **64 bajtów**.
- **Kod kontrolny ramki** – pole zawierające **sumę kontrolną ramki**, służącą do wykrywania ewentualnych błędów ramki. Urządzenie wysyłające dane **oblicza sumę kontrolną** i umieszcza ją w ramce, odbiorca danych, po jej otrzymaniu **również taką sumę oblicza, jeśli obydwie sumy się zgadzają ramka jest akceptowana, jeśli się różnią, ramkę traktuje się jako uszkodzoną i odrzuca**.

Całkowita wielkość ramki może wynieść maksymalnie **1518 bajtów** (przy obliczaniu jej wielkości, **nie brana jest pod uwagę preambuła i sygnał początku ramki**). Istnieje jeszcze jeden rodzaj ramek ethernetowych, których maksymalna wielkość może wynosić do **1522 bajtów**. Takie ramki stosuje się w **wirtualnych sieciach LAN**, w tzw. **VLAN-ach**.