Sieci komputerowe

Rafał J. Wysocki

Instytut Fizyki Teoretycznej, Wydział Fizyki UW

13 lutego 2011

Charakterystyczne cechy sieci komputerowej

Przesyłanie informacji w postaci binarnej

W sieci komputerowej wszystkie informacje (dane) są przesyłane w postaci binarnej (cyfrowej), czyli jako ciągi cyfr binarnych (bitów).

Przesyłanie po jednym bicie na raz jest mało efektywne (zabiera stosunkowo dużo czasu).

Symbole

Jeśli mamy do dyspozycji 2^n różnych rozróżnialnych postaci (stanów) sygnału, to każdej z nich można przypisać inne słowo n-bitowe, zwane symbolem.

Przesyłanie informacji z użyciem symboli

Symbolom można przypisywać różne wzory graficzne, takie jak znaki alfabetu.

Przesyłanie informacji na odległość z użyciem różnych zbiorów symboli, przeważnie w formie wizualnej, ma długą historię (miało miejsce już w starożytności).

Zostało usystematyzowane pod koniec XVIII i na początku XIX wieku (przekazywanie rozkazów we flotach wojennych z pomocą różnych zestawów flag sygnałowych, telegraf optyczny we Francji).

W I połowie XIX wieku skonstruowano telegraf elektryczny.

Telegraf

Charles Wheatstone

Symbolami są litery alfabetu angielskiego, wykorzystuje 5 przewodów i urządzenie wskazujące właściwą literę na podstawie kombinacji napięć.

Telegraf

Charles Wheatstone

Symbolami są litery alfabetu angielskiego, wykorzystuje 5 przewodów i urządzenie wskazujące właściwą literę na podstawie kombinacji napięć.

Samuel Morse

Dwa symbole, kropka i kreska (oraz dodatkowy stan „brak sygnału"), dwa przewody do przesyłania sygnałów. Linie telegraficzne proste w konstrukcji i duża niezawodność, standard w II połowie XIX wieku.

Telegraf

Charles Wheatstone

Symbolami są litery alfabetu angielskiego, wykorzystuje 5 przewodów i urządzenie wskazujące właściwą literę na podstawie kombinacji napięć.

Samuel Morse

Dwa symbole, kropka i kreska (oraz dodatkowy stan „brak sygnału"), dwa przewody do przesyłania sygnałów. Linie telegraficzne proste w konstrukcji i duża niezawodność, standard w II połowie XIX wieku.

Pozwalał na dostarczenie informacji do miejsca przeznaczenia zanim przestawały one być użyteczne.

Alfabet Morse'a

Alfabet Morse'a

System kodowania znaków alfabetu oraz cyfr z pomocą kombinacji kropek i kresek, wykorzystywany przy przesyłaniu informacji z pomocą telegrafu, a także z wykorzystaniem sygnalizacji świetlnej i (później) fal radiowych.

Alfabet Morse'a

Alfabet Morse'a

System kodowania znaków alfabetu oraz cyfr z pomocą kombinacji kropek i kresek, wykorzystywany przy przesyłaniu informacji z pomocą telegrafu, a także z wykorzystaniem sygnalizacji świetlnej i (później) fal radiowych.

W powszechnym użyciu w II połowie XIX i prawie całym XX wieku.

W czasie II Wojny Światowej (i jakiś czas po niej) wykorzystywany do przesyłania zaszyfrowanych informacji.

Komunikacja telefoniczna

Sieci telefoniczne, rozwijające się od początku XX wieku, pozwalają na komunikację głosową praktycznie bez opóźnień (opóźnienia są zaniedbywalnie małe).

Komunikacja telefoniczna

Sieci telefoniczne, rozwijające się od początku XX wieku, pozwalają na komunikację głosową praktycznie bez opóźnień (opóźnienia są zaniedbywalnie małe).

W I połowie XX wieku w sieciach telefonicznych wykorzystywano przełączanie obwodów (*ang. circuit switching*), czyli na potrzeby każdego połączenia telefonicznego zestawiany był oddzielny obwód elektryczny umożliwiający komunikację.

Komunikacja telefoniczna

Sieci telefoniczne, rozwijające się od początku XX wieku, pozwalają na komunikację głosową praktycznie bez opóźnień (opóźnienia są zaniedbywalnie małe).

W I połowie XX wieku w sieciach telefonicznych wykorzystywano przełączanie obwodów (*ang. circuit switching*), czyli na potrzeby każdego połączenia telefonicznego zestawiany był oddzielny obwód elektryczny umożliwiający komunikację.

Ograniczało to znacząco liczbę rozmów telefonicznych, które można było prowadzić jednocześnie (zwłaszcza długodystansowych).

Komunikacja radiowa

Błyskawiczny rozwój w okresie XX Wojny Światowej i późniejszym.

Komunikacja radiowa

Błyskawiczny rozwój w okresie XX Wojny Światowej i późniejszym.

Początkowo wykorzystywano komunikację z użyciem fali o jednej podstawowej częstotliwości, nazywanej falą nośną (*ang. carrier*), która była odpowiednio modulowana w celu przekazania informacji na odległość.

Komunikacja radiowa

Błyskawiczny rozwój w okresie XX Wojny Światowej i późniejszym.

Początkowo wykorzystywano komunikację z użyciem fali o jednej podstawowej częstotliwości, nazywanej falą nośną (*ang. carrier*), która była odpowiednio modulowana w celu przekazania informacji na odległość.

Nadawca i wszyscy odbiorcy informacji musieli korzystać z tej samej częstotliwości fali nośnej, więc łatwo było podsłuchiwać rozmowy radiowe.

Komunikacja radiowa

Błyskawiczny rozwój w okresie XX Wojny Światowej i późniejszym.

Początkowo wykorzystywano komunikację z użyciem fali o jednej podstawowej częstotliwości, nazywanej falą nośną (*ang. carrier*), która była odpowiednio modulowana w celu przekazania informacji na odległość.

Nadawca i wszyscy odbiorcy informacji musieli korzystać z tej samej częstotliwości fali nośnej, więc łatwo było podsłuchiwać rozmowy radiowe.

Liczba dostępnych częstotliwości fal nośnych jest ograniczona.

Koncepcja przełączania pakietów (*ang. packet switching*)

Leonard Kleinrock

- 1961 – Badania w dziedzinie teorii kolejkowania (*ang. queuig theory*) i praca na temat przełączania wiadomości (*ang. message switching*).
- 1962 – Praca doktorska poświęcona koncepcji przełączania pakietów (opublikowana jako książka w roku 1964).

Koncepcja przełączania pakietów (*ang. packet switching*)

Leonard Kleinrock

- 1961 – Badania w dziedzinie teorii kolejkowania (*ang. queuig theory*) i praca na temat przełączania wiadomości (*ang. message switching*).
- 1962 – Praca doktorska poświęcona koncepcji przełączania pakietów (opublikowana jako książka w roku 1964).

Paul Baran

- 1964 – Praca na temat ogólnej architektury wielkoskalowej, rozporoszonej sieci komunikacyjnej zdolnej przetrwać zniszczenie części węzłów.

Koncepcja sieci z przełączaniem pakietów

Paul Baran, podstawowe założenia

- Zdecentralizowana sieć z wieloma ścieżkami między dwoma dowolnymi punktami.
- Podział wiadomości użytkownika (*ang. user message*) na części, zwane początkowo *blokami wiadomości* (*ang. message block*), a później pakietami (*ang. packet*).
- Dostarczanie wiadomości z wykorzystaniem techniki przełączania store and forward.

Koncepcja sieci z przełączaniem pakietów

Paul Baran, podstawowe założenia

- Zdecentralizowana sieć z wieloma ścieżkami między dwoma dowolnymi punktami.
- Podział wiadomości użytkownika (*ang. user message*) na części, zwane początkowo *blokami wiadomości* (*ang. message block*), a później pakietami (*ang. packet*).
- Dostarczanie wiadomości z wykorzystaniem techniki przełączania store and forward.

Donald Davies

- 1965 – Praca rozwijająca koncepcje podobne do przedstawionych przez Paula Barana, wprowadzenie terminu „przełączanie pakietów”.

ARPANET *(Advanced Research Projects Agency Network)*

Lawrence Roberts

- Należał do zespołu Donalda Daviesa.
- Od roku 1966 w *ARPA Information Processing Techniques Office*, kierował zespołem projektującym ARPANET.

ARPANET *(Advanced Research Projects Agency Network)*

Lawrence Roberts

- Należał do zespołu Donalda Daviesa.
- Od roku 1966 w *ARPA Information Processing Techniques Office*, kierował zespołem projektującym ARPANET.

Pierwsze węzły ARPANET

- *University of California*, Los Angeles (UCLA), SDS Sigma 7
- *Stanford Research Institute's Augmentation Research Center*, SDS 940
- *University of California*, Santa Barbara, IBM 360/75
- *University of Utah's Computer Science Department*, DEC PDP-10

Pierwsza wiadomość przesłana o 10:30, 29 października 1969 r.

Pakiety i transmisja głosu

Przypuśćmy, że rejestrujemy natężenie dźwięku używając wartości $0 \ldots 255$ (8-bitowych) i zapisujemy próbki z częstotliwością 8 kHz (w celu wiernego odtworzenia składowych do 4 kHz włącznie).

Pakiety i transmisja głosu

Przypuśćmy, że rejestrujemy natężenie dźwięku używając wartości $0 \ldots 255$ (8-bitowych) i zapisujemy próbki z częstotliwością 8 kHz (w celu wiernego odtworzenia składowych do 4 kHz włącznie).

Wtedy 1 sekunda transmisji wymaga przesłania 64 kb ($64 \cdot 10^3$ b), ale nie musimy transmitować całości w sposób ciągły.

Pakiety i transmisja głosu

Przypuśćmy, że rejestrujemy natężenie dźwięku używając wartości $0 \ldots 255$ (8-bitowych) i zapisujemy próbki z częstotliwością 8 kHz (w celu wiernego odtworzenia składowych do 4 kHz włącznie).

Wtedy 1 sekunda transmisji wymaga przesłania 64 kb ($64 \cdot 10^3$ b), ale nie musimy transmitować całości w sposób ciągły.

Można podzielić 1 s transmisji na 8 pakietów po 1000 B (1 KB). Podczas przesyłania i odtwarzania (u odbiorcy) zawartości pierwszego pakietu rejestrujemy zawartość drugiego pakietu itd.

Pakiety i transmisja głosu

Przypuśćmy, że rejestrujemy natężenie dźwięku używając wartości $0 \ldots 255$ (8-bitowych) i zapisujemy próbki z częstotliwością 8 kHz (w celu wiernego odtworzenia składowych do 4 kHz włącznie).

Wtedy 1 sekunda transmisji wymaga przesłania 64 kb ($64 \cdot 10^3$ b), ale nie musimy transmitować całości w sposób ciągły.

Można podzielić 1 s transmisji na 8 pakietów po 1000 B (1 KB). Podczas przesyłania i odtwarzania (u odbiorcy) zawartości pierwszego pakietu rejestrujemy zawartość drugiego pakietu itd.

Jeżeli czas przesyłania pojedynczego pakietu jest (w przybliżeniu) stały i krótszy od 1/8 s, odbiorca nie powinien zauważyć różnicy.

Pakiety i transmisja głosu c. d.

Załóżmy, że przesyłanie jednego pakietu zajmuje czas $\Delta t = 1/8\,s$.

Pakiety i transmisja głosu c. d.

Załóżmy, że przesyłanie jednego pakietu zajmuje czas $\Delta t = 1/8\,s$.

Przebieg zdarzeń podczas transmisji

czas	**rejestracja**	**przesyłanie (sieć)**	**odtwarzanie**
t_0	pakiet 1		
$t_0 + \Delta t$	pakiet 2	pakiet 1	
$t_0 + 2\Delta t$	pakiet 3	pakiet 2	pakiet 1
$t_0 + 3\Delta t$	pakiet 4	pakiet 3	pakiet 2
$t_0 + 4\Delta t$	pakiet 5	pakiet 4	pakiet 3
$t_0 + 4\Delta t$	pakiet 6	pakiet 5	pakiet 4
	...		

Początki ARPANet – hardware

Budowa pierwszej sieci

Łącza – linie dzierżawione i modemy

IMP – urządzenia spełniające rolę współczesnych ruterów (*ang. router*), przesyłające pakiety (*ang. Interface Message Processor*)

Hosty – komputery wymieniające dane, łączone z IMP z pomocą złączy szeregowych

Początki ARPANet – hardware

Budowa pierwszej sieci

Łącza – linie dzierżawione i modemy

IMP – urządzenia spełniające rolę współczesnych ruterów (*ang. router*), przesyłające pakiety (*ang. Interface Message Processor*)

Hosty – komputery wymieniające dane, łączone z IMP z pomocą złączy szeregowych

Jako IMP początkowo używane były komputery Honeywell DDP-516 z 24 KiB pamięci (można ją było rozszerzać).

Jeden IMP mógł obsługiwać do 4 hostów i mógł łączyć się z 6 innymi IMP za pośrednictwem linii dzierżawionych, używając modemów.

Początki ARPANet – protokół 1822

Protokół sieciowy (*ang. network protocol*)

Zbiór reguł określających zasady działania elementów sieci (na ogół w pewnym ograniczonym zakresie lub na ustalonym poziomie).

Początki ARPANet – protokół 1822

Protokół sieciowy (*ang. network protocol*)

Zbiór reguł określających zasady działania elementów sieci (na ogół w pewnym ograniczonym zakresie lub na ustalonym poziomie).

Komunikacja (przesyłanie danych) w sieci jest możliwa tylko wtedy, gdy wszystkie elementy sieci realizujące tę samą funkcję zachowują się zgodnie z tym samym (ustalonym) protokołem sieciowym.

Początki ARPANet – protokół 1822

Protokół sieciowy (*ang. network protocol*)

Zbiór reguł określających zasady działania elementów sieci (na ogół w pewnym ograniczonym zakresie lub na ustalonym poziomie).

Komunikacja (przesyłanie danych) w sieci jest możliwa tylko wtedy, gdy wszystkie elementy sieci realizujące tę samą funkcję zachowują się zgodnie z tym samym (ustalonym) protokołem sieciowym.

W sieci ARPANet początkowo wykorzystywany był protokół 1822, zgodnie z którym wiadomość (pakiet) składała się z 3 części (pól):

- Typu wiadomości
- Numerycznego adresu hosta
- Pola danych

Początki ARPANet – protokół NCP

Przesyłanie wiadomości (pakietu) zgodnie z 1822

1. Tworzenie wiadomości (host-nadawca).
2. Przekazywanie wiadomości do IMP (host-nadawca).
3. Wyznaczanie odbiorcy wiadomości (IMP).
4. Dla odbiorców obsługiwanych przez dany IMP (lokalnych), przekazywanie wiadomości do odbiorcy.
5. Dla odbiorców obsługiwanych przez inny IMP, przekazywanie wiadomości do następnego IMP.

Początki ARPANet – protokół NCP

Przesyłanie wiadomości (pakietu) zgodnie z 1822

1. Tworzenie wiadomości (host-nadawca).
2. Przekazywanie wiadomości do IMP (host-nadawca).
3. Wyznaczanie odbiorcy wiadomości (IMP).
4. Dla odbiorców obsługiwanych przez dany IMP (lokalnych), przekazywanie wiadomości do odbiorcy.
5. Dla odbiorców obsługiwanych przez inny IMP, przekazywanie wiadomości do następnego IMP.

Protokół 1822 nie był przystosowany do obsługi komunikacji między różnymi programami w obrębie tego samego hosta i dlatego został zastąpiony bardziej zaawansowanym protokołem NCP (*ang. Network Control Program*).

Początki ARPANet – usługi

Podstawowe usługi w sieci ARPANet

Poczta elekrtoniczna (*ang. e-mail*), Ray Tomlinson, 1971 (stanowi 75% ruchu w roku 1973).

Transfer plików od 1973 z użyciem FTP (*File Transfer Protocol*).

Transmisja głosu od 1977 z użyciem NVP (*Network Voice Protocol*), zarzucona z powodu kłopotów technicznych.

Początki ARPANet – usługi

Podstawowe usługi w sieci ARPANet

Poczta elekrtoniczna (*ang. e-mail*), Ray Tomlinson, 1971 (stanowi 75% ruchu w roku 1973).

Transfer plików od 1973 z użyciem FTP (*File Transfer Protocol*).

Transmisja głosu od 1977 z użyciem NVP (*Network Voice Protocol*), zarzucona z powodu kłopotów technicznych.

Nie wiadomo jaka była treść pierwszego listu przesłanego pocztą elektroniczną. Prawdopodobnie był to pojedynczy ciąg znaków bez znaczenia (przesłany między komputerami stojącymi obok siebie).

Rozwój ARPANet

październik 1969 – pierwsza transmisja, 4 IMP

marzec 1970 – 5 IMP, dołączono IMP w firmie BBN (Bolt, Beranek and Newman), Wschodnie Wybrzeże USA

czerwiec 1970 – 9 IMP

grudzień 1970 – 13 IMP

wrzesień 1971 – 18 IMP (23 hosty rządowe i uniwersyteckie)

sierpień 1972 – 29 IMP

wrzesień 1973 – 40 IMP

1973 – do sieci dołączono IMP w Londynie

czerwiec 1974 – 46 IMP

lipiec 1975 – 57 IMP

1981 – 213 hostów, nowe hosty dołączane co ok. 20 dni

Powstanie NSFNet

Dalsze losy ARPANet

1975 – przejście pod kontrolę Defense Communications Agency

1983 – odłączenie MILNET (redukcja o 68 ze 113 IMP)

Powstanie NSFNet

Dalsze losy ARPANet

1975 – przejście pod kontrolę Defense Communications Agency

1983 – odłączenie MILNET (redukcja o 68 ze 113 IMP)

Powstanie i rozwój NSFNet

1985 – National Science Foundation (NSF) podejmuje budowę sieci szkieletowej w oparciu o łącza 56 kb/s (linie dzierżawione). Ma ona wykorzystywać protokoły opracowane na potrzeby ARPANet.

1986 – przebudowa szkieletu NFSNet (łącza 1,5 Mb/s).

1988 – NSFNet staje się główną siecią szkieletową Internetu.

NSFNet i Internet

Dalszy rozwój Internetu

1989 – postępująca komercjalizacja, podłączanie coraz większej liczby indywidualnych sieci do szkieletu NSFNet.

1991 – publikacja projektu World Wide Web (WWW) przez CERN.

lata 1990 – Internet rozwija się w tempie 100% rocznie (jeszcze szybciej w latach 1996/97). Powstają nowe usługi i protokoły.

NSFNet i Internet

Dalszy rozwój Internetu

1989 – postępująca komercjalizacja, podłączanie coraz większej liczby indywidualnych sieci do szkieletu NSFNet.

1991 – publikacja projektu World Wide Web (WWW) przez CERN.

lata 1990 – Internet rozwija się w tempie 100% rocznie (jeszcze szybciej w latach 1996/97). Powstają nowe usługi i protokoły.

Kluczowe protokoły sieciowe wykorzystywane w NSFNet i później w całym Internecie pochodzą z sieci ARPANet. Jest to rodzina protokołów TCP/IP wywodząca się z protokołu NCP.

Rodzina protokołów TCP/IP

Vinton Cerf i Robert Kahn

W 1973 roku po raz pierwszy opisali TCP jako bardziej elastyczne rozwinięcie protokołu NCP wykorzystywanego w sieci ARPANet (skrót TCP pochodził od nazwy Transmission Control Program).

Rodzina protokołów TCP/IP

Vinton Cerf i Robert Kahn

W 1973 roku po raz pierwszy opisali TCP jako bardziej elastyczne rozwinięcie protokołu NCP wykorzystywanego w sieci ARPANet (skrót TCP pochodził od nazwy Transmission Control Program).

TCP/IP

Wkrótce z pierwotnego TCP wydzielono protokół IP (*ang. Internet Protocol*), który opisywał zasady konstruowania i przesyłania pakietów oraz „nowy" TCP (*ang. Transmission Control Protocol*), opisujący zasady tworzenia logicznych połączeń między aplikacjami w sieci z wykorzystaniem pakietów IP.

Rodzina protokołów TCP/IP c. d.

UDP i ICMP

Rodzinę protokołów TCP/IP uzupełniają UDP (*ang. User Datagram Protocol*), pozwalający na wykorzystywanie pakietów IP do komunikacji bez logicznych połączeń oraz ICMP (*ang. Internet Control Message Protocol*), definiujący zasady przesyłania informacji kontrolnych (np. komunikatów o błędach) w sieci.

Rodzina protokołów TCP/IP c. d.

UDP i ICMP

Rodzinę protokołów TCP/IP uzupełniają UDP (*ang. User Datagram Protocol*), pozwalający na wykorzystywanie pakietów IP do komunikacji bez logicznych połączeń oraz ICMP (*ang. Internet Control Message Protocol*), definiujący zasady przesyłania informacji kontrolnych (np. komunikatów o błędach) w sieci.

Sieci rozległe

Sieci, w których początkowo używana była rodzina protokołów TCP/IP (ARPANet, NSFNet), były zbudowane w oparciu o łącza punkt-punkt (*ang. point-to-point*) między ruterami oddalonymi jeden od drugiego o wiele kilometrów, więc dziś nazwalibyśmy je sieciami rozległymi, czyli WAN (*ang. Wide Area Network*).

PARC Ethernet

Koncepcja przełączania pakietów sprawdza się nie tylko w sieciach, w których dane przesyłane są na wielkie odległości.

PARC Ethernet

Koncepcja przełączania pakietów sprawdza się nie tylko w sieciach, w których dane przesyłane są na wielkie odległości.

W latach 1973 – 1975 w laboratoriach firmy Xerox w Palo Alto skonstruowano sieć, w której komputery, zwane stacjami (*ang. station*), mogły przesyłać dane na niewielkie odległości korzystając ze wspólnego nośnika sygnału (*ang. signal carrier*).

PARC Ethernet

Koncepcja przełączania pakietów sprawdza się nie tylko w sieciach, w których dane przesyłane są na wielkie odległości.

W latach 1973 – 1975 w laboratoriach firmy Xerox w Palo Alto skonstruowano sieć, w której komputery, zwane stacjami (*ang. station*), mogły przesyłać dane na niewielkie odległości korzystając ze wspólnego nośnika sygnału (*ang. signal carrier*).

Nośnikiem sygnału był kabel koncentryczny, a sieć nazwano Ethernet, ponieważ jej działanie przypominało komunikację radiową z użyciem jednej częstotliwości fali nośnej (skrót PARC pochodzi od nazwy miejsca, Palo Alto Research Center).

Pakiety w sieci PARC Ethernet

Robert Metcalfe, David Boggs

Opisali sieć PARC Ethernet w 1976 r. (należeli do jej twórców).

Pakiety w sieci PARC Ethernet

Robert Metcalfe, David Boggs

Opisali sieć PARC Ethernet w 1976 r. (należeli do jej twórców).

Ramki

W sieci PARC Ethernet dane były przesyłane w pakietach zwanych ramkami (*ang. frame*).

Pakiety w sieci PARC Ethernet

Robert Metcalfe, David Boggs

Opisali sieć PARC Ethernet w 1976 r. (należeli do jej twórców).

Ramki

W sieci PARC Ethernet dane były przesyłane w pakietach zwanych ramkami (*ang. frame*).

Struktura ramki PARC Ethernet

1. Preambuła (*ang. preamble*) – 8 B
2. Adres miejsca przeznaczenia (*ang. destination*) – 1 B
3. Adres nadawcy (*ang. source*) – 1 B
4. Typ (*ang. type*) – 2 B
5. Dane

Zasada działania sieci Ethernet

CSMA/CD

CSMA/CD (*ang. Carrier Sense, Multiple Access with Collision Detection*) jest techniką pozwalającą na wykorzystywanie wspólnego nośnika sygnału i wykrywanie kolizji sygnałów.

Zasada działania sieci Ethernet

CSMA/CD

CSMA/CD (*ang. Carrier Sense, Multiple Access with Collision Detection*) jest techniką pozwalającą na wykorzystywanie wspólnego nośnika sygnału i wykrywanie kolizji sygnałów.

Wysyłanie ramki

1. Czekamy, aż nośnik będzie wolny (nikt nie nadaje).
2. Jeżeli nośnik jest wolny, nadajemy.
3. Sprawdzamy, czy doszło do kolizji sygnałów.
4. Jeżeli doszło do kolizji sygnałów, czekamy (przez losowy czas, zależny od liczby dotychczasowych powtórzeń) i powtarzamy.

Zasada działania sieci Ethernet

CSMA/CD

CSMA/CD (*ang. Carrier Sense, Multiple Access with Collision Detection*) jest techniką pozwalającą na wykorzystywanie wspólnego nośnika sygnału i wykrywanie kolizji sygnałów.

Wysyłanie ramki

1. Czekamy, aż nośnik będzie wolny (nikt nie nadaje).
2. Jeżeli nośnik jest wolny, nadajemy.
3. Sprawdzamy, czy doszło do kolizji sygnałów.
4. Jeżeli doszło do kolizji sygnałów, czekamy (przez losowy czas, zależny od liczby dotychczasowych powtórzeń) i powtarzamy.

Każda ramka dociera do wszystkich stacji.

DIX Ethernet i projekt 802.3

DEC, Intel, Xerox

Trzy liczące się firmy zaangażowane w rozwój sieci Ethernet, zmieniony format ramki i przepustowość 10 Mb/s.

DIX Ethernet i projekt 802.3

DEC, Intel, Xerox

Trzy liczące się firmy zaangażowane w rozwój sieci Ethernet, zmieniony format ramki i przepustowość 10 Mb/s.

Luty 1980

IEEE (*ang. Institute of Electrical and Electronics Engineers*) uruchamia projekt 802 w celu standaryzacji sieci lokalnych, czyli LAN (*ang. Local Area Network*). Grupa robocza nr 3 zajmowała się siecią Ethernet.

DIX Ethernet i projekt 802.3

DEC, Intel, Xerox

Trzy liczące się firmy zaangażowane w rozwój sieci Ethernet, zmieniony format ramki i przepustowość 10 Mb/s.

Luty 1980

IEEE (*ang. Institute of Electrical and Electronics Engineers*) uruchamia projekt 802 w celu standaryzacji sieci lokalnych, czyli LAN (*ang. Local Area Network*). Grupa robocza nr 3 zajmowała się siecią Ethernet.

1981, 3Com (Robert Metcalfe)

Pierwszy adapter sieciowy zgodny z nowym formatem ramki DIX (Ethernet II).

Specyfikacja sieci Ethernet

Podział specyfikacji na dwie części, zwane warstwami (*ang. layer*):

- MAC (*ang. Media Access Control*) – dostęp do nośnika.
- PHY (*ang. Physical Layer Interface*) – interfejs sprzętowy.

Specyfikacja sieci Ethernet

Podział specyfikacji na dwie części, zwane warstwami (*ang. layer*):

- MAC (*ang. Media Access Control*) – dostęp do nośnika.
- PHY (*ang. Physical Layer Interface*) – interfejs sprzętowy.

Warstwa MAC specyfikacji jest wspólna dla wszystkich rodzajów sieci Ethernet:

- Adresowanie
- Format ramki
- Technika CSMA/CD

Specyfikacja sieci Ethernet

Podział specyfikacji na dwie części, zwane warstwami (*ang. layer*):

- MAC (*ang. Media Access Control*) – dostęp do nośnika.
- PHY (*ang. Physical Layer Interface*) – interfejs sprzętowy.

Warstwa MAC specyfikacji jest wspólna dla wszystkich rodzajów sieci Ethernet:

- Adresowanie
- Format ramki
- Technika CSMA/CD

Warstwa PHY obejmuje specyfikację nośnika i urządzenia nadawczo-odbiorczego, zwanego nadbiornikiem (*ang. transciever*).

Adresy i format ramki dla sieci Ethernet

Adresy MAC

Adresy stacji w sieci Ethernet, zwane adresami MAC (*ang. MAC address*), są słowami 48-bitowymi (6 B).

Adresy i format ramki dla sieci Ethernet

Adresy MAC

Adresy stacji w sieci Ethernet, zwane adresami MAC (*ang. MAC address*), są słowami 48-bitowymi (6 B).

Ramka Ethernet

1. Preambuła (*ang. preamble*) – 7 B
2. Znacznik początku (*ang. start delimiter*) – 1 B
3. Adres MAC miejsca przeznaczenia (*ang. destination*) – 6 B
4. Adres MAC nadawcy (*ang. source*) – 6 B
5. Typ (*ang. type*) lub długość (*ang. length*) – 2 B
6. (Opcjonalny nagłówek 802.2 LLC – 3 B lub 4 B)
7. Dane – do 46 do 1500 B
8. Suma kontrolna – 4 B

Rodzaje sieci Ethernet

Przykłady PHY

- 10Base2 – 10 Mb/s, „cienki” kabel koncentryczny, 1985
- 10Base-T – 10 Mb/s, skrętka dwyżyłowa kategorii 3, 1990
- 100Base-TX – 100 Mb/s, skrętka dwyżyłowa kategorii 5, 1995
- 100Base-FX – 100 Mb/s, światłowód, 1995
- 1000Base-X – 1000 Mb/s, światłowód, 1998
- 1000Base-T – 1000 Mb/s, skrętka dwyżyłowa kategorii 5e, 1999

Rodzaje sieci Ethernet

Przykłady PHY

- 10Base2 – 10 Mb/s, „cienki” kabel koncentryczny, 1985
- 10Base-T – 10 Mb/s, skrętka dwyżyłowa kategorii 3, 1990
- 100Base-TX – 100 Mb/s, skrętka dwyżyłowa kategorii 5, 1995
- 100Base-FX – 100 Mb/s, światłowód, 1995
- 1000Base-X – 1000 Mb/s, światłowód, 1998
- 1000Base-T – 1000 Mb/s, skrętka dwyżyłowa kategorii 5e, 1999

Sieci wykorzystujące skrętkę dwużyłową wymagają koncentratorów (*ang. hub*) lub przełączników (*ang. switch*).

Sieci światłowodowe wymagają stosowania przełączników.

Koncentratory i przełączniki

Dla skrętki i światłowodów łącza są punkt-punkt, ale nie bezpośrednio między stacjami.

Koncentratory i przełączniki

Dla skrętki i światłowodów łącza są punkt-punkt, ale nie bezpośrednio między stacjami.

Koncentrator

- Urządzenie wieloportowe (można podłączyć wiele stacji).
- Sygnał odbierany przez jeden port jest wzmacniany i wysyłany przez pozostałe porty (replikacja).
- Spełnia rolę wspólnego nośnika.

Koncentratory i przełączniki

Dla skrętki i światłowodów łącza są punkt-punkt, ale nie bezpośrednio między stacjami.

Koncentrator

- Urządzenie wieloportowe (można podłączyć wiele stacji).
- Sygnał odbierany przez jeden port jest wzmacniany i wysyłany przez pozostałe porty (replikacja).
- Spełnia rolę wspólnego nośnika.

Przełącznik

- Urządzenie wieloportowe (można podłączyć wiele stacji).
- „Uczy się" adresów MAC stacji.
- Ramka adresowana do stacji jest przesyłana tylko przez port, przez który ta stacja jest dostępna.

Domeny kolizji i rozgłaszania

Domena kolizji (*ang. collision domain*), segment

Obejmuje stacje (w sieci Ethernet), dla których może dojść do kolizji sygnałów wysłanych przez dowolną parę z nich (tzn. sygnał wysłany przez jedną stację dociera do wszystkich pozostałych).

Domeny kolizji i rozgłaszania

Domena kolizji (*ang. collision domain*), segment

Obejmuje stacje (w sieci Ethernet), dla których może dojść do kolizji sygnałów wysłanych przez dowolną parę z nich (tzn. sygnał wysłany przez jedną stację dociera do wszystkich pozostałych).

Domena rozgłaszania (*ang. broadcast domain*)

Obejmuje stacje (w sieci Ethernet), dla których ramka wysłana przez jedną z nich może dotrzeć do dowolnej innej stacji. Powstaje poprzez połączenie wielu domen kolizji z pomocą przełączników.

Domeny kolizji i rozgłaszania

Domena kolizji (*ang. collision domain*), segment

Obejmuje stacje (w sieci Ethernet), dla których może dojść do kolizji sygnałów wysłanych przez dowolną parę z nich (tzn. sygnał wysłany przez jedną stację dociera do wszystkich pozostałych).

Domena rozgłaszania (*ang. broadcast domain*)

Obejmuje stacje (w sieci Ethernet), dla których ramka wysłana przez jedną z nich może dotrzeć do dowolnej innej stacji. Powstaje poprzez połączenie wielu domen kolizji z pomocą przełączników.

Ramka rozgłoszeniowa (*ang. broadcast frame*)

Ramka, dla której adres miejsca przezaczenia jest słowem o wszystkich bitach równych 1, rozsyłana do wszystkich stacji.

Standard 802.11

Zastosowanie przełączników na dużą skalę w II połowie lat dziewięćdziesiątych XX wieku spowodowało, że (przełączane) sieci Ethernet praktycznie wyparły z rynku inne standardy LAN, poza bezprzewodowymi (*ang. wireless*).

Standard 802.11

Zastosowanie przełączników na dużą skalę w II połowie lat dziewięćdziesiątych XX wieku spowodowało, że (przełączane) sieci Ethernet praktycznie wyparły z rynku inne standardy LAN, poza bezprzewodowymi (*ang. wireless*).

Standard 802.11, 1997

Pierwsza specyfikacja bezprzewodowej sieci LAN:

- Adresowanie jak dla warstwy MAC sieci Ethernet.
- Podobny format ramki, ale wiele rodzajów ramek (także kontrolne).
- CSMA/CA (*ang. Carrier Sense, Multiple Access with Collision Avoidance*).
- Możliwość „przezroczystego" łączenia z sieciami Ethernet.

PHY dla sieci bezprzewodowych

802.11a, 1999 – pasmo 5 GHz, maksymalna „surowa" przepustowość 54 Mb/s

802.11b, 1999 – pasmo 2,4 GHz, maksymalna „surowa" przepustowość 11 Mb/s, 14 kanałów po 22 MHz

802.11g, 2003 – pasmo 2,4 GHz, maksymalna „surowa" przepustowość 54 Mb/s, 14 kanałów po 22 MHz

802.11n, 2009 – pasmo 2,4 GHz albo 5 GHz, maksymalna „surowa" przepustowość 600 Mb/s, kanały 40 MHz, wiele anten

PHY dla sieci bezprzewodowych

802.11a, 1999 – pasmo 5 GHz, maksymalna „surowa" przepustowość 54 Mb/s

802.11b, 1999 – pasmo 2,4 GHz, maksymalna „surowa" przepustowość 11 Mb/s, 14 kanałów po 22 MHz

802.11g, 2003 – pasmo 2,4 GHz, maksymalna „surowa" przepustowość 54 Mb/s, 14 kanałów po 22 MHz

802.11n, 2009 – pasmo 2,4 GHz albo 5 GHz, maksymalna „surowa" przepustowość 600 Mb/s, kanały 40 MHz, wiele anten

Wi-Fi

Znak firmowy i symbol marketingowy odpowiadający rodzinie standardów 802.11. Wykorzystywany przez organizację Wi-Fi Alliance zajmującą się certyfikacją sprzętu.

Budowa sieci bezprzewodowych

Punkt dostępowy

AP (*ang. Access Point*) spełnia rolę koncentratora w sieci bezprzewodowej.

Budowa sieci bezprzewodowych

Punkt dostępowy

AP (*ang. Access Point*) spełnia rolę koncentratora w sieci bezprzewodowej.

Komórka (*ang. cell, service set*)

Zespół stacji korzystających z jednego AP. Odpowiada segmentowi (domenie kolizji) w sieci Ethernet.

Budowa sieci bezprzewodowych

Punkt dostępowy

AP (*ang. Access Point*) spełnia rolę koncentratora w sieci bezprzewodowej.

Komórka (*ang. cell, service set*)

Zespół stacji korzystających z jednego AP. Odpowiada segmentowi (domenie kolizji) w sieci Ethernet.

ESS

Wiele komórek może tworzyć większą strukturę, nazywaną ESS (*ang. Extended Service Set*), odpowiadającą domenie rozgłaszania w sieci Ethernet. Połączenia między AP mogą być bezprzewodowe lub zbudowane z segmentów sieci Ethernet.

Zastosowanie sieci lokalnych do uwspólniania zasobów

Uwspólnianie zasobów (*ang. sharing of resources*)

Wykorzystywanie jednego zasobu (*ang. resource*), takiego jak twardy dysk o dużej pojemności, wydajna drukarka itp., w sposób umożliwiający korzystanie z niego wielu komputerom (najczęściej za pośrednictwem sieci).

Zastosowanie sieci lokalnych do uwspólniania zasobów

Uwspólnianie zasobów (*ang. sharing of resources*)

Wykorzystywanie jednego zasobu (*ang. resource*), takiego jak twardy dysk o dużej pojemności, wydajna drukarka itp., w sposób umożliwiający korzystanie z niego wielu komputerom (najczęściej za pośrednictwem sieci).

Serwer (*ang. server*)

1. Komputer w sieci, za pośrednictwem którego inne komputery korzystają ze wspólnych zasobów, np. serwer plikowy (*ang. file server*), sewer drukarkowy (*ang. print serwer*).
2. Oprogramowanie realizujące określoną usługę dostępną w sieci, np. serwer WWW, serwer FTP.
3. Komputer, na którym uruchamiane są programy-serwery.

NetWare

Klient (*ang. client*)

1. Komputer w sieci, korzystający z zasobów udostępnianych przez inne komputery (serwery).
2. Oprogramowanie pozwalające na korzystanie z określonej usługi dostępnej w sieci, np. klient WWW, klient FTP.
3. Komputer, na którym uruchamiane są programy-klienci.

NetWare

Klient (*ang. client*)

1. Komputer w sieci, korzystający z zasobów udostępnianych przez inne komputery (serwery).
2. Oprogramowanie pozwalające na korzystanie z określonej usługi dostępnej w sieci, np. klient WWW, klient FTP.
3. Komputer, na którym uruchamiane są programy-klienci.

Novell NetWare

System operacyjny dla serwerów, pozwalający na udostępnianie zasobów, takich jak przestrzeń dyskowa i drukarki, w sieciach lokalnych. Spowodował pojawienie się dedykowanych serwerów.

Rodzina protokołów IPX/SPX

IPX (*ang. Internetwork Packet Exchange*)

Protokół sieciowy spełniający rolę podobną do IP, wykorzystywany (początkowo) przez NetWare.

SPX (*ang. Sequenced Packet Exchange*)

Protokół sieciowy spełniający rolę podobną do TCP, wykorzystywany (początkowo) przez NetWare.

Rodzina protokołów IPX/SPX

IPX (*ang. Internetwork Packet Exchange*)

Protokół sieciowy spełniający rolę podobną do IP, wykorzystywany (początkowo) przez NetWare.

SPX (*ang. Sequenced Packet Exchange*)

Protokół sieciowy spełniający rolę podobną do TCP, wykorzystywany (początkowo) przez NetWare.

W II połowie lat 1980 i I połowie lat 1990 rodzina protokołów IPX/SPX była faktycznym standardem dla sieci lokalnych (z powodu wielkiej popularności NetWare).

LAN Manager i Windows NT Advanced Server

LAN Manager

- System operacyjny dla serwerów, umożliwiający udostępnianie zasobów w sieci lokalnej.
- Miał stanowić odpowiedź Microsoftu na NetWare.
- Wykorzystywał protokoły NBF (*ang. NetBIOS Frames*) i SMB (*ang Signal Message Block*).
- Wersja 2.2 z 1990 roku pozwalała używać TCP/IP (zamiast NBF).

LAN Manager i Windows NT Advanced Server

LAN Manager

- System operacyjny dla serwerów, umożliwiający udostępnianie zasobów w sieci lokalnej.
- Miał stanowić odpowiedź Microsoftu na NetWare.
- Wykorzystywał protokoły NBF (*ang. NetBIOS Frames*) i SMB (*ang Signal Message Block*).
- Wersja 2.2 z 1990 roku pozwalała używać TCP/IP (zamiast NBF).

Windows NT Advanced Server, 1993

- System operacyjny dla serwerów, umożliwiający udostępnianie zasobów w sieci lokalnej.
- Pełne wsparcie dla TCP/IP w wersji 3.5 (1994).

Rywalizacja między NetWare i Windows NT Server

W II połowie lat 1990 NetWare zaczyna tracić udział w rynku na rzecz Windows NT Server.

Rywalizacja między NetWare i Windows NT Server

W II połowie lat 1990 NetWare zaczyna tracić udział w rynku na rzecz Windows NT Server.

Przewaga Windows NT

- Graficzny interfejs użytkownika.
- Serwer nie musi być dedykowany (można na nim pracować, jak na kliencie).
- Większa łatwość podłączania sieci lokalnej do Internetu.

Rywalizacja między NetWare i Windows NT Server

W II połowie lat 1990 NetWare zaczyna tracić udział w rynku na rzecz Windows NT Server.

Przewaga Windows NT

- Graficzny interfejs użytkownika.
- Serwer nie musi być dedykowany (można na nim pracować, jak na kliencie).
- Większa łatwość podłączania sieci lokalnej do Internetu.

Windows 98, 1998

Windows 98 zawiera oprogramowanie „internetowe” (Internet Explorer, Outlook Express) oraz oprogramowanie umożliwiające korzystanie z serwerów Windows NT bez instalowania dodakowych komponentów.

Powstanie Linuksa

Richard M. Stallman

W 1983 rozpoczyna Projekt GNU (*ang. GNU Project*), mający na celu stworzenie systemu operacyjnego zbudowanego wyłącznie z oprogramowania dostępnego wraz z kodem źródłowym i na licencji gwarantującej możliwość korzystania z niego bez ograniczeń (*ang. Free Software*). Do 1992 gotowe są wszystkie główne składniki systemu oprócz jądra (*ang. kernel*).

Powstanie Linuksa

Richard M. Stallman

W 1983 rozpoczyna Projekt GNU (*ang. GNU Project*), mający na celu stworzenie systemu operacyjnego zbudowanego wyłącznie z oprogramowania dostępnego wraz z kodem źródłowym i na licencji gwarantującej możliwość korzystania z niego bez ograniczeń (*ang. Free Software*). Do 1992 gotowe są wszystkie główne składniki systemu oprócz jądra (*ang. kernel*).

Powstanie Linuksa

Richard M. Stallman

W 1983 rozpoczyna Projekt GNU (*ang. GNU Project*), mający na celu stworzenie systemu operacyjnego zbudowanego wyłącznie z oprogramowania dostępnego wraz z kodem źródłowym i na licencji gwarantującej możliwość korzystania z niego bez ograniczeń (*ang. Free Software*). Do 1992 gotowe są wszystkie główne składniki systemu oprócz jądra (*ang. kernel*).

Linus Torvalds

W 1991 zaczyna rozwijać jądro systemu operacyjnego, które nazywa Linux (od połączenia imienia Linus z nazwą systemu operacyjnego UNIX). Linus traktuje swój projekt jako badawczy i wybiera dla niego licencję wykorzystywaną w Projekcie GNU.

GNU/Linux

W 1992, przez połączenie jądra systemu zwanego Linux z Projektem GNU, powstaje w pełni funkcjonalny system operacyjny na licencji typu *Open Source*.

GNU/Linux

W 1992, przez połączenie jądra systemu zwanego Linux z Projektem GNU, powstaje w pełni funkcjonalny system operacyjny na licencji typu *Open Source*.

W II połowie lat 1990 GNU/Linux zaczyna być zaawansowany technicznie i znajduje liczne zastosowania:

- hosty internetowe
- rutery (m. in. łączenie sieci lokalnych z Internetem)
- serwery (w tym plikowe i drukarkowe)
- stacje robocze (obliczenia, modelowanie, wizualizacja danych)
- komputery-klienci usług internetowych (WWW, FTP, e-mail)

Linux i sieci lokalne

Pod koniec lat 1990 Linux był najbardziej wszechstronnym systemem. Mógł być wykorzystywany praktycznie do wszystkiego, zawierał wsparcie dla praktycznie wszystkich wykorzystywanych protokołów sieciowych (w tym IPX/SPX) i był bezpłatny.

Linux i sieci lokalne

Pod koniec lat 1990 Linux był najbardziej wszechstronnym systemem. Mógł być wykorzystywany praktycznie do wszystkiego, zawierał wsparcie dla praktycznie wszystkich wykorzystywanych protokołów sieciowych (w tym IPX/SPX) i był bezpłatny.

Łączenie sieci lokalnych z Internetem bez pomocy Linuksa było bardzo kosztowne.

Linux i sieci lokalne

Pod koniec lat 1990 Linux był najbardziej wszechstronnym systemem. Mógł być wykorzystywany praktycznie do wszystkiego, zawierał wsparcie dla praktycznie wszystkich wykorzystywanych protokołów sieciowych (w tym IPX/SPX) i był bezpłatny.

Łączenie sieci lokalnych z Internetem bez pomocy Linuksa było bardzo kosztowne.

Mając komputer z Linuksem w sieci można było zrobić z niego serwer (kosztem większego nakładu pracy, niż w przypadku systemów „komercjalnych").

Linux i sieci lokalne

Pod koniec lat 1990 Linux był najbardziej wszechstronnym systemem. Mógł być wykorzystywany praktycznie do wszystkiego, zawierał wsparcie dla praktycznie wszystkich wykorzystywanych protokołów sieciowych (w tym IPX/SPX) i był bezpłatny.

Łączenie sieci lokalnych z Internetem bez pomocy Linuksa było bardzo kosztowne.

Mając komputer z Linuksem w sieci można było zrobić z niego serwer (kosztem większego nakładu pracy, niż w przypadku systemów „komercjalnych").

Linux wydajnie działał nawet na kiepskim sprzęcie.

Zmiana charakteru sieci lokalnych w XXI wieku

Po roku 2000 jednym z głównych zadań sieci lokalnych staje się zapewnienie dostępu do Internetu komputerom w sieci (łącze internetowe staje się wspólnym zasobem udostępnianym komputerom w sieci).

Zmiana charakteru sieci lokalnych w XXI wieku

Po roku 2000 jednym z głównych zadań sieci lokalnych staje się zapewnienie dostępu do Internetu komputerom w sieci (łącze internetowe staje się wspólnym zasobem udostępnianym komputerom w sieci).

Maleje znaczenie udostępniania zasobów dyskowych i drukarek (przynajmniej w małych sieciach). Zasoby są udostępniane bardziej w celu wymiany danych, niż w celu przechowywania ich.

Zmiana charakteru sieci lokalnych w XXI wieku

Po roku 2000 jednym z głównych zadań sieci lokalnych staje się zapewnienie dostępu do Internetu komputerom w sieci (łącze internetowe staje się wspólnym zasobem udostępnianym komputerom w sieci).

Maleje znaczenie udostępniania zasobów dyskowych i drukarek (przynajmniej w małych sieciach). Zasoby są udostępniane bardziej w celu wymiany danych, niż w celu przechowywania ich.

Dedykowane serwery stają się hostami internetowymi udostępniającymi usługi. Wszędzie wykorzystuje się rodzinę protokołów TCP/IP.

Zmiana charakteru sieci lokalnych w XXI wieku

Po roku 2000 jednym z głównych zadań sieci lokalnych staje się zapewnienie dostępu do Internetu komputerom w sieci (łącze internetowe staje się wspólnym zasobem udostępnianym komputerom w sieci).

Maleje znaczenie udostępniania zasobów dyskowych i drukarek (przynajmniej w małych sieciach). Zasoby są udostępniane bardziej w celu wymiany danych, niż w celu przechowywania ich.

Dedykowane serwery stają się hostami internetowymi udostępniającymi usługi. Wszędzie wykorzystuje się rodzinę protokołów TCP/IP.

Pojawiają się wydajne sieci bezprzewodowe (standard 802.11).

Transmisja z użyciem fal elemtromagnetycznych (EM)

Nadawca

Wytwarza fale EM w ściśle określony sposób, aby po dotarciu do miejsca przeznaczenia można było odczytać z nich zakodowane (*ang. encoded*) informacje.

Transmisja z użyciem fal elemtromagnetycznych (EM)

Nadawca

Wytwarza fale EM w ściśle określony sposób, aby po dotarciu do miejsca przeznaczenia można było odczytać z nich zakodowane (*ang. encoded*) informacje.

Odbiorca

Przeprowadza pomiary własności fal EM docierających do niego od nadawcy i na podstawie ich wyników stara się odtworzyć oryginalny sygnał (mogą występować błędy).

Transmisja z użyciem fal elemtromagnetycznych (EM)

Nadawca

Wytwarza fale EM w ściśle określony sposób, aby po dotarciu do miejsca przeznaczenia można było odczytać z nich zakodowane (*ang. encoded*) informacje.

Odbiorca

Przeprowadza pomiary własności fal EM docierających do niego od nadawcy i na podstawie ich wyników stara się odtworzyć oryginalny sygnał (mogą występować błędy).

Ośrodek (*ang. medium*)

Układ fizyczny, w którym rozchodzą się fale EM wytwarzane przez nadawcę i docierające do odbiorcy. Jego własności mogą wpływać na własności fal EM.

Wąskopasmowe i szerokopasmowe techniki transmisji

Protokół transmisji

Umowa między nadawcą i odbiorcą dotycząca tego, w jaki sposób mają być wytwarzane fale EM w celu zakodowania określonych informacji i jak należy interpretować ich (zmierzone) własności.

Wąskopasmowe i szerokopasmowe techniki transmisji

Protokół transmisji

Umowa między nadawcą i odbiorcą dotycząca tego, w jaki sposób mają być wytwarzane fale EM w celu zakodowania określonych informacji i jak należy interpretować ich (zmierzone) własności.

Transmisja wąskopasmowa (*ang. baseband*)

Wykorzystywana jest jedna sinusoidalna fala nośna (*ang. carrier*), której własności (amplituda, częstotliwość, przesunięcie fazowe) są zmieniane w czasie w celu zakodowania informacji.

Wąskopasmowe i szerokopasmowe techniki transmisji

Protokół transmisji

Umowa między nadawcą i odbiorcą dotycząca tego, w jaki sposób mają być wytwarzane fale EM w celu zakodowania określonych informacji i jak należy interpretować ich (zmierzone) własności.

Transmisja wąskopasmowa (*ang. baseband*)

Wykorzystywana jest jedna sinusoidalna fala nośna (*ang. carrier*), której własności (amplituda, częstotliwość, przesunięcie fazowe) są zmieniane w czasie w celu zakodowania informacji.

Transmisja szerokopasmowa (*ang. broadband*)

Wykorzystywane są sygnały zbudowane z wielu fal sinusoidalnych o różnych własnościach.

Fala nośna

Techniki wąskopasmowe wykorzystują pojedynczą falę nośną (formuła dla ustalonego punktu przestrzeni):

$$\psi_c(t) = A_c \sin(2\pi f_c t + \phi_c)$$

gdzie

- A_c – aplituda (*ang. amplitude*)
- f_c – częstotliwość (*ang. frequency*)
- ϕ_c – przesunięcie fazowe (*ang. phase shift*) lub faza

Fala nośna

Techniki wąskopasmowe wykorzystują pojedynczą falę nośną (formuła dla ustalonego punktu przestrzeni):

$$\psi_c(t) = A_c \sin(2\pi f_c t + \phi_c)$$

gdzie

- A_c – aplituda (*ang. amplitude*)
- f_c – częstotliwość (*ang. frequency*)
- ϕ_c – przesunięcie fazowe (*ang. phase shift*) lub faza

Kodowanie informacji odbywa się poprzez dokonywanie zmian A_c, f_c lub ϕ_c w czasie w sposób uzgodniony z odbiorcą.

Modulowanie amplitudy

AM (*ang. Amplitude Modulation*)

Technika polegająca na manipulowaniu amplitudą fali nośnej:

$$\psi(t) = [A_c + m(t)] \sin(2\pi f_c t + \phi_c)$$

Modulowanie amplitudy

AM (*ang. Amplitude Modulation*)

Technika polegająca na manipulowaniu amplitudą fali nośnej:

$$\psi(t) = [A_c + m(t)] \sin(2\pi f_c t + \phi_c)$$

W przypadku danych (tzn. informacji w postaci binarnej) $m(t)$ jest (zwykle) funkcją schodkową.

Modulowanie amplitudy

AM (*ang. Amplitude Modulation*)

Technika polegająca na manipulowaniu amplitudą fali nośnej:

$$\psi(t) = [A_c + m(t)] \sin(2\pi f_c t + \phi_c)$$

W przypadku danych (tzn. informacji w postaci binarnej) $m(t)$ jest (zwykle) funkcją schodkową.

ASK (*ang. Amplitude-Shift Keying*)

Forma AM, w której $m(t)$ przyjmuje 2^k różnych (dyskretnych) wartości i każda z tych wartości reprezentuje ustalony ciąg k bitów, zwany symbolem.

Modulowanie częstotliwości

FM (*ang. Frequency Modulation*)

Technika polegająca na manipulowaniu częstotliwością fali nośnej:

$$\psi(t) = A_c \sin\left\{2\pi\left[f_c + \int_0^t m(\tau)d\tau\right]t + \phi_c\right\}$$

Modulowanie częstotliwości

FM (*ang. Frequency Modulation*)

Technika polegająca na manipulowaniu częstotliwością fali nośnej:

$$\psi(t) = A_c \sin\left\{2\pi\left[f_c + \int_0^t m(\tau)d\tau\right]t + \phi_c\right\}$$

FSK (*ang. Frequency-Shift Keying*)

Forma FM, w której $m(t)$ przyjmuje 2^k różnych wartości reprezentujących k-bitowe symbole.

Modulowanie częstotliwości

FM (*ang. Frequency Modulation*)

Technika polegająca na manipulowaniu częstotliwością fali nośnej:

$$\psi(t) = A_c \sin\left\{2\pi\left[f_c + \int_0^t m(\tau)d\tau\right]t + \phi_c\right\}$$

FSK (*ang. Frequency-Shift Keying*)

Forma FM, w której $m(t)$ przyjmuje 2^k różnych wartości reprezentujących k-bitowe symbole.

Okazuje się, że dla FM moc transmitowanego (tzn. modulowanego) sygnału jest skoncentrowana w przedziale częstotliwości węższym, niż w przypadku AM.

Modulowanie fazy

PM (*ang. Phase Modulation*)

Technika polegająca na manipulowaniu przesunięciem fazowym (fazą) fali nośnej:

$$\psi(t) = A_c \sin[2\pi f_c t + \phi_c + m(t)]$$

Modulowanie fazy

PM (*ang. Phase Modulation*)

Technika polegająca na manipulowaniu przesunięciem fazowym (fazą) fali nośnej:

$$\psi(t) = A_c \sin[2\pi f_c t + \phi_c + m(t)]$$

PSK (*ang. Phase-Shift Keying*)

Forma PM, w której $m(t)$ przyjmuje 2^k różnych wartości reprezentujących k-bitowe symbole.

Modulowanie fazy

PM (*ang. Phase Modulation*)

Technika polegająca na manipulowaniu przesunięciem fazowym (fazą) fali nośnej:

$$\psi(t) = A_c \sin[2\pi f_c t + \phi_c + m(t)]$$

PSK (*ang. Phase-Shift Keying*)

Forma PM, w której $m(t)$ przyjmuje 2^k różnych wartości reprezentujących k-bitowe symbole.

Fale kodujące różne symbole można reprezentować z pomocą kombinacji liniowych funkcji $cos(2\pi f_c t)$ oraz $sin(2\pi f_c t)$, które można przedstawiać jako liczby zespolone lub punkty na płaszczyźnie z kartezjańskim układem współrzędnych.

Kwadraturowe modulowanie amplitudy

QAM (*ang. Quadrature Amplitude Modulation*)

Technika polegająca na składaniu dwóch fal nośnych o tej samej częstotliwości, ale przesuniętych w fazie o $\pi/2$ jedna względem drugiej:

$$\psi(t) = A_c[1 + x(t)]\cos(2\pi f_c t) + A_c[1 + y(t)]\sin(2\pi f_c t)$$

Kwadraturowe modulowanie amplitudy

QAM (*ang. Quadrature Amplitude Modulation*)

Technika polegająca na składaniu dwóch fal nośnych o tej samej częstotliwości, ale przesuniętych w fazie o $\pi/2$ jedna względem drugiej:

$$\psi(t) = A_c[1 + x(t)]\cos(2\pi f_c t) + A_c[1 + y(t)]\sin(2\pi f_c t)$$

Do transmisji danych stosuje się warianty QAM, w których każda z funkcj $x(t)$ oraz $y(t)$ przyjmuje pewną liczbę dyskretnych wartości, $x_1, x_2, \ldots, x_n$ oraz $y_1, y_2, \ldots, y_n$. Wówczas symbole są reprezentowane przez punkty (x_i, y_j) na płaszczyźnie z kartezjańskim układem współrzędnych (lub odpowiadające im liczby zespolone).

Kodowanie typu Manchester (*ang. Manchester encoding*)

Wariant BPSK dla częstotliwości fali nośnej równej 0, w którym zmiana poziomu sygnału (w danym cyklu) z niskiego na wysoki oznacza 0, a zmiana poziomu sygnału z wysokiego na niski oznacza 1.

Kodowanie typu Manchester (*ang. Manchester encoding*)

Wariant BPSK dla częstotliwości fali nośnej równej 0, w którym zmiana poziomu sygnału (w danym cyklu) z niskiego na wysoki oznacza 0, a zmiana poziomu sygnału z wysokiego na niski oznacza 1.

Dzięki takiemu rozwiązaniu poziom sgnału zmienia się w czasie niezależnie od tego, jakie kombinacje bitów są transmitowane (odbiorca może łatwo synchronizować zegar z nadawcą).

Kodowanie typu Manchester (*ang. Manchester encoding*)

Wariant BPSK dla częstotliwości fali nośnej równej 0, w którym zmiana poziomu sygnału (w danym cyklu) z niskiego na wysoki oznacza 0, a zmiana poziomu sygnału z wysokiego na niski oznacza 1.

Dzięki takiemu rozwiązaniu poziom sgnału zmienia się w czasie niezależnie od tego, jakie kombinacje bitów są transmitowane (odbiorca może łatwo synchronizować zegar z nadawcą).

Efektywna częstotliwość sygnału jest rzędu BR (*ang. bit rate*), czyli liczby bitów przesyłanych w jednostce czasu.

Kodowanie typu Manchester (*ang. Manchester encoding*)

Wariant BPSK dla częstotliwości fali nośnej równej 0, w którym zmiana poziomu sygnału (w danym cyklu) z niskiego na wysoki oznacza 0, a zmiana poziomu sygnału z wysokiego na niski oznacza 1.

Dzięki takiemu rozwiązaniu poziom sgnału zmienia się w czasie niezależnie od tego, jakie kombinacje bitów są transmitowane (odbiorca może łatwo synchronizować zegar z nadawcą).

Efektywna częstotliwość sygnału jest rzędu BR (*ang. bit rate*), czyli liczby bitów przesyłanych w jednostce czasu.

Możliwe dla BR co najwyżej rzędu 10^7/s ze względu na przepisy.

Kodowanie z wieloma poziomami napięcia

MLT-3 (*ang. Multi-Level Transition 3*)

Kodowanie z wykorzystaniem 3 poziomów napięcia, V, 0, $-V$, 0, aplikowanych cyklicznie:

- jeśli następnym transmitowanym bitem jest 1, zmieniamy poziom napięcia na kolejny w cyklu,
- w przeciwnym wypadku nie zmieniamy poziomu napięcia.

Kodowanie z wieloma poziomami napięcia

MLT-3 (*ang. Multi-Level Transition 3*)

Kodowanie z wykorzystaniem 3 poziomów napięcia, V, 0, $-V$, 0, aplikowanych cyklicznie:

- jeśli następnym transmitowanym bitem jest 1, zmieniamy poziom napięcia na kolejny w cyklu,
- w przeciwnym wypadku nie zmieniamy poziomu napięcia.

Efektywna częstotliwość sygnału jest rzędu BR/4, ale pojawia się problem z synchronizacją zegarów przy przesyłaniu ciągów zer.

Kodowanie z wieloma poziomami napięcia

MLT-3 (*ang. Multi-Level Transition 3*)

Kodowanie z wykorzystaniem 3 poziomów napięcia, V, 0, $-V$, 0, aplikowanych cyklicznie:

- jeśli następnym transmitowanym bitem jest 1, zmieniamy poziom napięcia na kolejny w cyklu,
- w przeciwnym wypadku nie zmieniamy poziomu napięcia.

Efektywna częstotliwość sygnału jest rzędu BR/4, ale pojawia się problem z synchronizacją zegarów przy przesyłaniu ciągów zer.

4B/5B

Każde 4 bity danych kodujemy jako 5-bitowy symbol tak, aby w każdym symbolu występowały co najmniej dwie jedynki.

Zastosowania technik wąskopasmowych

Sieć Ethernet

- 10Base-T – kodowanie typu Manchester
- 100Base-TX – MLT-3 z 4B/5B
- 1000Base-T – PAM-5

Zastosowania technik wąskopasmowych

Sieć Ethernet

10Base-T – kodowanie typu Manchester

100Base-TX – MLT-3 z 4B/5B

1000Base-T – PAM-5

Sieci 802.11

Różne warianty PSK i QAM, ale w „ramach” technik szerokopasmowych.

Transmisja w szerokim zakresie częstotliwości

Techniki szerokopasmowe charakteryzują się tym, że moc transmitowanego sygnału jest rozłożona na przedział częstotliwości.

Transmisja w szerokim zakresie częstotliwości

Techniki szerokopasmowe charakteryzują się tym, że moc transmitowanego sygnału jest rozłożona na przedział częstotliwości.

OFDM (*ang. Orthogonal Frequency-Division Multiplexing*)

- Przedział częstotliwości dzielony jest na „podpasma".
- W każdym „podmasmie" mamy falę nośną, której częstotliwość odpowiada środkowi „podpasma". Są to pomocnicze fale nośne (*ang. subcarrier*).
- Pomocnicze fale nośne są dobrane tak, aby były wzajemnie ortogonalne.
- Każda pomocnicza fala nośna jest modulowana oddzielnie.
- Sygnały wynikowe są sumowane z użyciem odwrotnej transformaty Fouriera.

OFDM

Część rzeczywista i urojona wyjściowego sygnału modulują składowe fali nośnej (o częstotliwości odpowiadającej środkowi przedziału, który mamy do dyspozycji) przesunięte w fazie o $\pi/2$.

OFDM

Część rzeczywista i urojona wyjściowego sygnału modulują składowe fali nośnej (o częstotliwości odpowiadającej środkowi przedziału, który mamy do dyspozycji) przesunięte w fazie o $\pi/2$.

Odbiór synału OFDM

- Modulowane składowe fali nośnej są poddawane transformacji, z której można odzyskać rzeczywistą i urojoną część sygnału reprezentującego dane.
- Są one dostarczane do układu przeprowadzającego transformatę Fouriera.
- W wyniku otrzymujemy szereg sgnałów odpowiadających (modulowanym) pomocniczym falom nośnym.
- Z każdego z nich można „odzyskać" zakodowane bity danych.

CDM (*ang. Code Division Multiplexing*)

Technika transmisji wykorzystująca ortogonalne wektory:

- Każdy nadawca otrzymuje unikatowy wektor o współrzędnych równych 1 i -1 (liczba współrzędnych zależy od liczby nadawców).
- Wektory są tak dobrane, aby były wzajemnie ortogonalne.
- Aby wysłać bit (danych) równy 1, nadawca używa współrzędnych swojego wektora do modulowania fali nośnej.
- Aby wysłać bit równy 0, nadawca używa współrzędnych swojego wektora pomnożonego przez -1 do modulowania fali nośnej.
- Sygnały od różnych nadawców nie interferują destrukcyjnie.

CDMA (*ang. Code Division Multiple Access*)

Technika podobna do CDM, wykorzystująca pseudolosowe sekwencje liczb (zwykle 1 i -1) zamiast wzajemnie ortogonalnych wektorów:

- Pseudolosowa sekwencja liczb jest różna dla każdego nadawcy.
- Odbiorcy wiedzą która sekwencja odpowiada danemu nadawcy.
- Okazuje się, że to wystarcza do rozróżnienia sygnałów od różnych nadawców.
- Dodatkowym efektem jest poszerzenie widma mocy sygnału.

CDMA (*ang. Code Division Multiple Access*)

Technika podobna do CDM, wykorzystująca pseudolosowe sekwencje liczb (zwykle 1 i -1) zamiast wzajemnie ortogonalnych wektorów:

- Pseudolosowa sekwencja liczb jest różna dla każdego nadawcy.
- Odbiorcy wiedzą która sekwencja odpowiada danemu nadawcy.
- Okazuje się, że to wystarcza do rozróżnienia sygnałów od różnych nadawców.
- Dodatkowym efektem jest poszerzenie widma mocy sygnału.

DSSS (*ang. Direct-Sequence Spread Spectrum*)

Technika modulacji wykorzystywana w sieciach 802.11 i 802.11b opracowana w oparciu o CDMA.

Zastosowania technik szerokopasmowych

DSL (*ang. Digital Subscriber Line*)

Technologia, dzięki której można uzyskać duże BR na zwykłych liniach telefonicznych (miedziane, 2 lub 4 przewody). Wykorzystuje OFDM jako technikę modulacji.

Zastosowania technik szerokopasmowych

DSL (*ang. Digital Subscriber Line*)

Technologia, dzięki której można uzyskać duże BR na zwykłych liniach telefonicznych (miedziane, 2 lub 4 przewody). Wykorzystuje OFDM jako technikę modulacji.

ADSL (*ang. Asymmetric Digital Subscriber Line*)

Wariant DSL, w którym przepustowość łącza w kierunku do abonenta (*ang. downstream*) jest większa, niż w kierunku przeciwnym (*ang. upstream*). Może być wykorzystywany na liniach o niskiej jakości.

Zastosowania technik szerokopasmowych c. d.

Sieci 802.11

Wykorzystują DSSS (802.11, 802.11b, 802.11g, 802.11n) oraz OFDM (802.11a, 802.11g, 802.11n).

Zastosowania technik szerokopasmowych c. d.

Sieci 802.11

Wykorzystują DSSS (802.11, 802.11b, 802.11g, 802.11n) oraz OFDM (802.11a, 802.11g, 802.11n).

Sieci światłowodowe

Wykorzystują warianty OFDM znane jako WDM (*ang. Wavelength-Division Multiplexing*) i DWDM (*ang. Dense WDM*).

Adresy IP

Zgodnie z protokołem IP komputery (i inne urządzenia) w sieci, czyli węzły sieci (*ang. network node*), są identyfikowane z pomocą numerycznych adresów.

Adresy IP

Zgodnie z protokołem IP komputery (i inne urządzenia) w sieci, czyli węzły sieci (*ang. network node*), są identyfikowane z pomocą numerycznych adresów.

Adresy IP (*ang. IP address*)

Są słowami 32-bitowymi, jednak nie każde takie słowo może być adresem węzła sieci (w szczególności adresem węzła sieci IP nie może być słowo, w którym trzy najbardziej znaczące bity są jedynkami lub którego najbardziej znaczący bajt ma wartość 0).

Adresy IP

Zgodnie z protokołem IP komputery (i inne urządzenia) w sieci, czyli węzły sieci (*ang. network node*), są identyfikowane z pomocą numerycznych adresów.

Adresy IP (*ang. IP address*)

Są słowami 32-bitowymi, jednak nie każde takie słowo może być adresem węzła sieci (w szczególności adresem węzła sieci IP nie może być słowo, w którym trzy najbardziej znaczące bity są jedynkami lub którego najbardziej znaczący bajt ma wartość 0).

DQN (*ang. Dotted-Quad Notation*)

Tradycyjny sposób zapisu adresów IP, w którym poszczególne bajty słowa są zapisywane oddzielnie, jako liczby dziesiętne i oddzielane kropkami, np. `193.0.80.28` (adres serwera *www.fuw.edu.pl*).

Podsieci IP

Adresy IP służą m. in. do identyfikacji źródła (*ang. source*) oraz miejsca przeznaczenia (*ang. destination*) pakietów. Adresy te są przydzielane w grupach zwanych podsieciami.

Podsieci IP

Adresy IP służą m. in. do identyfikacji źródła (*ang. source*) oraz miejsca przeznaczenia (*ang. destination*) pakietów. Adresy te są przydzielane w grupach zwanych podsieciami.

Podsieć IP (*ang. IP subnet*)

Zbiór adresów IP zawierających ten sam ciąg bitów, czyli wzór bitowy (*ang. bit pattern*), na pewnej (ustalonej) liczbie najbardziej znaczących pozycji.

Podsieci IP

Adresy IP służą m. in. do identyfikacji źródła (*ang. source*) oraz miejsca przeznaczenia (*ang. destination*) pakietów. Adresy te są przydzielane w grupach zwanych podsieciami.

Podsieć IP (*ang. IP subnet*)

Zbiór adresów IP zawierających ten sam ciąg bitów, czyli wzór bitowy (*ang. bit pattern*), na pewnej (ustalonej) liczbie najbardziej znaczących pozycji.

Przykład: podsieć Wydziału Fizyki

Podsieć Wydziału Fizyki zawiera adresy IP, w których 22 najbardziej znaczące pozycje bitowe zawierają wzór bitowy 1100000100000000010100.

Podsieci IP – symboliczne oznaczenie

Prefiks (*ang. prefix*)

Część adresu IP wspólna dla wszystkich adresów w danej podsieci.

Podsieci IP – symboliczne oznaczenie

Prefiks (*ang. prefix*)

Część adresu IP wspólna dla wszystkich adresów w danej podsieci.

Długość prefiksu

Liczba pozycji bitowych zajmowanych przez prefiks w adresie IP.

Podsieci IP – symboliczne oznaczenie

Prefiks (*ang. prefix*)

Część adresu IP wspólna dla wszystkich adresów w danej podsieci.

Długość prefiksu

Liczba pozycji bitowych zajmowanych przez prefiks w adresie IP.

Symboliczne oznaczenie podsieci IP

Składa się z prefiksu, uzupełnionego do pełnego adresu IP (w notacji DQN) poprzez wstawienie zer na pozycje bitowe poza prefiksem, znaku / oraz długości prefiksu.

Podsieci IP – symboliczne oznaczenie

Prefiks (*ang. prefix*)

Część adresu IP wspólna dla wszystkich adresów w danej podsieci.

Długość prefiksu

Liczba pozycji bitowych zajmowanych przez prefiks w adresie IP.

Symboliczne oznaczenie podsieci IP

Składa się z prefiksu, uzupełnionego do pełnego adresu IP (w notacji DQN) poprzez wstawienie zer na pozycje bitowe poza prefiksem, znaku / oraz długości prefiksu.

Przykład: podsieć Wydziału Fizyki

Symboliczne oznaczenie dla podsieci IP Wydziału Fizyki ma postać `193.0.80.0/22`.

Podział podsieci IP na części

Zasada podziału podsieci IP

1. Podsieć IP z prefiksem o długości k można podzielić na dwie podsieci z prefiksami o długości $k + 1$.
2. Dla każdej z nowych podsieci k najbardziej znaczących bitów prefiksu pokrywa się z prefiksem oryginalnej podsieci.
3. Zatem prefiksy nowych podsieci różnią się jednym, najmniej znaczącym bitem.

Podział podsieci IP na części

Zasada podziału podsieci IP

1. Podsieć IP z prefiksem o długości k można podzielić na dwie podsieci z prefiksami o długości $k + 1$.
2. Dla każdej z nowych podsieci k najbardziej znaczących bitów prefiksu pokrywa się z prefiksem oryginalnej podsieci.
3. Zatem prefiksy nowych podsieci różnią się jednym, najmniej znaczącym bitem.

Przykład: podsieć Wydziału Fizyki

Podsieć `193.0.80.0/22` można podzielić na podsieci `193.0.80.0/23` i `193.0.82.0/23`. Z kolei każdą z nich można podzielić na 2 podsieci, otrzymując 4 podsieci: `193.0.80.0/24`, `193.0.81.0/24`, `193.0.82.0/24`, `193.0.83.0/24`.

Podsieci IP – maski podsieci

Problem

Jak rozróżnić prefiksy podsieci `193.0.80.0/22` i `193.0.80.0/24` uzupełnione zerami do pełnych adresów IP?

Podsieci IP – maski podsieci

Problem

Jak rozróżnić prefiksy podsieci `193.0.80.0/22` i `193.0.80.0/24` uzupełnione zerami do pełnych adresów IP?

Maska podsieci (*ang. subnet mask*)

Dla danej podsieci IP jest ciągiem (32) bitów, w którym na pozycjach bitowych odpowiadających prefiksowi znajdują się jedynki, a na pozostałych pozycjach – zera.

Podsieci IP – maski podsieci

Problem

Jak rozróżnić prefiksy podsieci `193.0.80.0/22` i `193.0.80.0/24` uzupełnione zerami do pełnych adresów IP?

Maska podsieci (*ang. subnet mask*)

Dla danej podsieci IP jest ciągiem (32) bitów, w którym na pozycjach bitowych odpowiadających prefiksowi znajdują się jedynki, a na pozostałych pozycjach – zera.

Przykład: podsieć Wydziału Fizyki

Maska podsieci dla podsieci Wydziału Fizyki, `193.0.80.0/22`, ma (w notacji DQN) postać `255.255.252.0`. Dla podsieci `193.0.80.0/24` ma ona postać `255.255.255.0`.

Bezpośrednio połączone węzły sieci

Każdy węzeł sieci IP „zna” maski podsieci, do których należy (węzeł może mieć więcej adresów IP, niż jeden).

Bezpośrednio połączone węzły sieci

Każdy węzeł sieci IP „zna” maski podsieci, do których należy (węzeł może mieć więcej adresów IP, niż jeden).

Węzły sieci IP należące do tej samej podsieci (tzn. mające adresy IP z jednakowym prefiksem, skojarzone z tą samą maską podsieci) uważa się za bezpośrednio połączone.

Bezpośrednio połączone węzły sieci

Każdy węzeł sieci IP „zna” maski podsieci, do których należy (węzeł może mieć więcej adresów IP, niż jeden).

Węzły sieci IP należące do tej samej podsieci (tzn. mające adresy IP z jednakowym prefiksem, skojarzone z tą samą maską podsieci) uważa się za bezpośrednio połączone.

Iloczyn bitowy (*ang. bitwise AND*)

Operacja na słowach o jednakowej liczbie bitów, n, dająca w wyniku słowo n-bitowe, w którym cyfra na pozycji bitowej i jest jedynką tylko wtedy, gdy w każdym z argumentów cyfra na pozycji bitowej i jest jedynką.

Bezpośrednio połączone węzły sieci c. d.

Węzeł sieci IP może użyć iloczynu bitowego w celu stwierdzenia, czy miejsce przeznaczenia pakietu jest bezpośrednio połączone z nim.

Bezpośrednio połączone węzły sieci c. d.

Węzeł sieci IP może użyć iloczynu bitowego w celu stwierdzenia, czy miejsce przeznaczenia pakietu jest bezpośrednio połączone z nim.

W tym celu dla każdego ze swoich adresów IP:

1. Oblicza iloczyn bitowy tego adresu z opdowiadającą mu maską podsieci i otrzymuje prefiks podsieci.
2. Oblicza iloczyn bitowy adresu miejsca przeznaczenia pakietu z maską podsieci odpowiadającą temu adresowi i porównuje wynik z prefiksem podsieci otrzymanym w poprzednim kroku.
3. Jeśli są one jednakowe, miejsce przeznaczenia pakietu jest bezpośrednio połączone z danym węzłem sieci.

Zasada przydziału adresów IP

W sieci lokalnej (LAN)

Wszystkie stacje powinny być traktowane jako bezpośrednio połączone ze sobą nawzajem, czyli powinny mieć adresy z tej samej podsieci IP.

Zasada przydziału adresów IP

W sieci lokalnej (LAN)

Wszystkie stacje powinny być traktowane jako bezpośrednio połączone ze sobą nawzajem, czyli powinny mieć adresy z tej samej podsieci IP.

Dla sieci Ethernet każdej domenie rozgłaszania powinna odpowiadać oddzielna podsieć IP, dla której długość prefiksu, k, spełnia nierówność:

$$2^{32-k} - 2 \geqslant n$$

gdzie n jest liczbą stacji w danej domenie rozgłaszania.

Zasada przydziału adresów IP

W sieci lokalnej (LAN)

Wszystkie stacje powinny być traktowane jako bezpośrednio połączone ze sobą nawzajem, czyli powinny mieć adresy z tej samej podsieci IP.

Dla sieci Ethernet każdej domenie rozgłaszania powinna odpowiadać oddzielna podsieć IP, dla której długość prefiksu, k, spełnia nierówność:

$$2^{32-k} - 2 \geqslant n$$

gdzie n jest liczbą stacji w danej domenie rozgłaszania.

Adresy, w których wszystkie bity poza prefiksem mają jednakową wartość (tzn. wszystkie są jedynkami albo zerami), są zarezerwowane.

Zasada przesyłania pakietów IP (w sieci lokalnej)

Jeżeli miejsce przeznaczenia pakietu jest bezpośrednio połączone z danym węzłem, powinien on wykorzystać protokół niższego poziomu (np. protokół sieci Ethernet) do przesłania pakietu.

Zasada przesyłania pakietów IP (w sieci lokalnej)

Jeżeli miejsce przeznaczenia pakietu jest bezpośrednio połączone z danym węzłem, powinien on wykorzystać protokół niższego poziomu (np. protokół sieci Ethernet) do przesłania pakietu.

W przeciwnym wypadku powinien on wyznaczyć taki węzeł bezpośrednio połączony z nim, który dysponuje informacjami o tym, gdzie znajduje się miejsce przeznaczenia pakietu. Następnie pakiet powinien być przekazany temu węzłowi z wykorzystaniem protokołu niższego poziomu.

Zasada przesyłania pakietów IP (w sieci lokalnej)

Jeżeli miejsce przeznaczenia pakietu jest bezpośrednio połączone z danym węzłem, powinien on wykorzystać protokół niższego poziomu (np. protokół sieci Ethernet) do przesłania pakietu.

W przeciwnym wypadku powinien on wyznaczyć taki węzeł bezpośrednio połączony z nim, który dysponuje informacjami o tym, gdzie znajduje się miejsce przeznaczenia pakietu. Następnie pakiet powinien być przekazany temu węzłowi z wykorzystaniem protokołu niższego poziomu.

Ruter (*ang. router*)

Węzeł sieci rozległej (np. sieci IP), który należy do wielu różnych podsieci i pośredniczy w przesyłaniu pakietów między węzłami znajdującymi się w tych podsieciach.

Model sieci IP

Łącza punkt-punkt między ruterami mogą być traktowane jako podsieci IP o dwóch węzłach (zawierające po 4 adresy).

Model sieci IP

Łącza punkt-punkt między ruterami mogą być traktowane jako podsieci IP o dwóch węzłach (zawierające po 4 adresy).

Przy takim założeniu można powiedzieć, że:

1. Sieć IP w ogólności składa się z wielu podsieci.
2. W każdej z tych podsieci węzły są bezpośrednio połączone ze sobą i wykorzystują protokoły niższego poziomu do przesyłania pakietów między sobą.
3. Przesyłanie pakietów między węzłami położonymi w różnych podsieciach wymaga pośrednictwa ruterów.

Model sieci IP

Łącza punkt-punkt między ruterami mogą być traktowane jako podsieci IP o dwóch węzłach (zawierające po 4 adresy).

Przy takim założeniu można powiedzieć, że:

1. Sieć IP w ogólności składa się z wielu podsieci.
2. W każdej z tych podsieci węzły są bezpośrednio połączone ze sobą i wykorzystują protokoły niższego poziomu do przesyłania pakietów między sobą.
3. Przesyłanie pakietów między węzłami położonymi w różnych podsieciach wymaga pośrednictwa ruterów.

W takiej sieci pakiety przesyłane są „skokami” (*ang. hop*), nadawca-ruter, ruter-ruter lub ruter-odbiorca.

Tabele tras

Węzły sieci IP wyznaczają adres następnego skoku (*ang. next hop address*) dla pakietów korzystając ze specjalnych tabel.

Tabele tras

Węzły sieci IP wyznaczają adres następnego skoku (*ang. next hop address*) dla pakietów korzystając ze specjalnych tabel.

Tabela tras (*ang. routing table*)

Zawiera informacje o miejscach przeznaczenia pakietów dostępnych z danego węzła. Każdemu znanemu miejscu przeznaczenia pakietów przypisuje się wiersz tabeli, nazywany trasą (*ang. route*), zawierający m. in.:

1. Prefiks miejsca przeznaczenia pakietów uzupełniony zerami do pełnego adresu IP.
2. Maskę podsieci miejsca przeznaczenia pakietów.
3. Adres IP węzła, któremu należy przekazać pakiet, jeżeli pasuje on do tego miejsca przeznaczenia.

Dopasowywanie pakietów do tras

Dla każdej trasy wykonuje się iloczyn bitowy maski podsieci z adresem miejsca przeznaczenia w pakiecie i wynik jest porównywany z prefiksem odpowiadającym tej trasie. Jeżeli są identyczne, trasę uważa się za dopasowaną do pakietu (poszukiwanie dopasowania kończy się).

Dopasowywanie pakietów do tras

Dla każdej trasy wykonuje się iloczyn bitowy maski podsieci z adresem miejsca przeznaczenia w pakiecie i wynik jest porównywany z prefiksem odpowiadającym tej trasie. Jeżeli są identyczne, trasę uważa się za dopasowaną do pakietu (poszukiwanie dopasowania kończy się).

W pierwszej kolejności sprawdzane są trasy odpowiadające miejscom przeznaczenia o najdłuższych prefiksach (tzn. o największej liczbie jedynek w masce podsieci).

Dopasowywanie pakietów do tras

Dla każdej trasy wykonuje się iloczyn bitowy maski podsieci z adresem miejsca przeznaczenia w pakiecie i wynik jest porównywany z prefiksem odpowiadającym tej trasie. Jeżeli są identyczne, trasę uważa się za dopasowaną do pakietu (poszukiwanie dopasowania kończy się).

W pierwszej kolejności sprawdzane są trasy odpowiadające miejscom przeznaczenia o najdłuższych prefiksach (tzn. o największej liczbie jedynek w masce podsieci).

Trasa domyślna (*ang. default route*)

Trasa, dla której prefix i maska podsieci są słowami złożonymi z samych zer (pasuje ona do każdego pakietu).

Budowanie tabel tras

Tabele tras mogą być statyczne lub dynamiczne.

Budowanie tabel tras

Tabele tras mogą być statyczne lub dynamiczne.

Statyczne tabele tras zawierają informacje wprowadzone „ręcznie” przez administartorów ruterów. Dynamiczne tabele tras są tworzone przez same rutery na podstawie informacji uzyskanych od innych ruterów.

Budowanie tabel tras

Tabele tras mogą być statyczne lub dynamiczne.

Statyczne tabele tras zawierają informacje wprowadzone „ręcznie” przez administartorów ruterów. Dynamiczne tabele tras są tworzone przez same rutery na podstawie informacji uzyskanych od innych ruterów.

Dynamiczne tabele tras mogą zmieniać się w reakcji na zmiany konfiguracji sieci wykrywane przez rutery.

Budowanie tabel tras

Tabele tras mogą być statyczne lub dynamiczne.

Statyczne tabele tras zawierają informacje wprowadzone „ręcznie” przez administartorów ruterów. Dynamiczne tabele tras są tworzone przez same rutery na podstawie informacji uzyskanych od innych ruterów.

Dynamiczne tabele tras mogą zmieniać się w reakcji na zmiany konfiguracji sieci wykrywane przez rutery.

Rutery tworzące dynamiczne tabele tras komunikują się między sobą z pomocą specjalnych protokołów wymiany informacji, zwanych protokołami rutingu (*ang. routing protocol*).

Struktura pakietu IP

Pakiet IP składa się z nagłówka (*ang. header*), zawierającego informacje potrzebne do dostarczenia go do miejsca przeznaczenia oraz użytecznego ładunku danych (*ang. data payload*).

Struktura pakietu IP

Pakiet IP składa się z nagłówka (*ang. header*), zawierającego informacje potrzebne do dostarczenia go do miejsca przeznaczenia oraz użytecznego ładunku danych (*ang. data payload*).

Minimalna długość nagłówka pakietu IP wynosi 20 B.

Struktura pakietu IP

Pakiet IP składa się z nagłówka (*ang. header*), zawierającego informacje potrzebne do dostarczenia go do miejsca przeznaczenia oraz użytecznego ładunku danych (*ang. data payload*).

Minimalna długość nagłówka pakietu IP wynosi 20 B.

Nagłówek pakietu IP zawiera m. in.:

- Adres nadawcy (*ang. source address*).
- Adres miejsca przeznaczenia (*ang. destination address*).
- Długość (*ang. length*) pakietu (max. 65535).
- Limit liczby skoków, czyli TTL (*ang. Time To Live*).
- Sumę kontrolną dla nagłówka.

Pola nagłówka IP i przesyłanie pakietów

Pole TTL w nagłówku pakietu jest inicjowane przez nadawcę, a później zminiejszane o 1 przez każdy ruter przesyłający pakiet. Ruter, dla którego pole TTL w pakiecie osiągnie wartość 0, odrzuca pakiet i wysyła komunikat do nadawcy (wykorzystując protokół ICMP).

Pola nagłówka IP i przesyłanie pakietów

Pole TTL w nagłówku pakietu jest inicjowane przez nadawcę, a później zminiejszane o 1 przez każdy ruter przesyłający pakiet. Ruter, dla którego pole TTL w pakiecie osiągnie wartość 0, odrzuca pakiet i wysyła komunikat do nadawcy (wykorzystując protokół ICMP).

Każdy ruter przesyłający pakiet sprawdza i przelicza sumę kontrolną dla nagłówka. W przypadku stwierdzenia niezgodności pakiet jest odrzucany jako uszkodzony i do nadawcy wysyłany jest komunikat (z wykorzystaniem ICMP).

Pola nagłówka IP i przesyłanie pakietów

Pole TTL w nagłówku pakietu jest inicjowane przez nadawcę, a później zmniejszane o 1 przez każdy ruter przesyłający pakiet. Ruter, dla którego pole TTL w pakiecie osiągnie wartość 0, odrzuca pakiet i wysyła komunikat do nadawcy (wykorzystując protokół ICMP).

Każdy ruter przesyłający pakiet sprawdza i przelicza sumę kontrolną dla nagłówka. W przypadku stwierdzenia niezgodności pakiet jest odrzucany jako uszkodzony i do nadawcy wysyłany jest komunikat (z wykorzystaniem ICMP).

Jeśli w tabeli tras rutera nie ma trasy pasującej do adresu miejsca przeznaczenia w pakiecie, jest on odrzucany i do nadawcy wysyłany jest komunikat (z wykorzystaniem ICMP).

Pakiety IP i sieć Ethernet

W sieci Ethernet pakiety IP są przesyłane wewnątrz ramek.

Pakiety IP i sieć Ethernet

W sieci Ethernet pakiety IP są przesyłane wewnątrz ramek.

Z punktu widzenia protokołu sieci Ethernet pakiet IP stanowi (w całości, łącznie z nagłówkiem) użyteczny ładunek danych i jest umieszczany (w całości) w polu ramki przeznaczonym na dane.

Pakiety IP i sieć Ethernet

W sieci Ethernet pakiety IP są przesyłane wewnątrz ramek.

Z punktu widzenia protokołu sieci Ethernet pakiet IP stanowi (w całości, łącznie z nagłówkiem) użyteczny ładunek danych i jest umieszczany (w całości) w polu ramki przeznaczonym na dane.

Adresy IP są odwzorowywane na adresy MAC w sieci Ethernet z wykorzystaniem specjalnego protokołu wymiany informacji o nazwie ARP (*ang. Address Resolution Protocol*).

Pakiety IP i sieć Ethernet

W sieci Ethernet pakiety IP są przesyłane wewnątrz ramek.

Z punktu widzenia protokołu sieci Ethernet pakiet IP stanowi (w całości, łącznie z nagłówkiem) użyteczny ładunek danych i jest umieszczany (w całości) w polu ramki przeznaczonym na dane.

Adresy IP są odwzorowywane na adresy MAC w sieci Ethernet z wykorzystaniem specjalnego protokołu wymiany informacji o nazwie ARP (*ang. Address Resolution Protocol*).

Stacja poszukująca adresu MAC odpowiadającego danemu adresowi IP wysyła ramkę rozgłoszeniową z pytaniem o ten adres. W odpowiedzi powinna otrzymać ramkę od „właściciela" poszukiwanego adresu.

Hosty i procesy

IP reguluje dostarczanie danych z hosta do hosta, ale nie rozwiązuje do końca problemu komunikacji między różnymi procesami.

Hosty i procesy

IP reguluje dostarczanie danych z hosta do hosta, ale nie rozwiązuje do końca problemu komunikacji między różnymi procesami.

Procesy (*ang. process*)

Programy w pamięci komputera, które mogą być wykonywane z wykorzystaniem strategii podziału czasu (*ang. time sharing*).

Hosty i procesy

IP reguluje dostarczanie danych z hosta do hosta, ale nie rozwiązuje do końca problemu komunikacji między różnymi procesami.

Procesy (*ang. process*)

Programy w pamięci komputera, które mogą być wykonywane z wykorzystaniem strategii podziału czasu (*ang. time sharing*).

W systemach wielozadaniowych różne procesy mogą reprezentować różnych użytkowników sieci, np. zalogowanych na danym hoście za pośrednictwem usługi SSH (*ang. Secure SHell*).

Przesyłanie danych z procesu do procesu

Aby umożliwić komunikację między procesami, które mogą reprezentować różnych użytkowników korzystających z różnych hostów, trzeba wprowadzić jakąś identyfikację procesów w systemie na potrzeby przesyłania danych w sieci.

Przesyłanie danych z procesu do procesu

Aby umożliwić komunikację między procesami, które mogą reprezentować różnych użytkowników korzystających z różnych hostów, trzeba wprowadzić jakąś identyfikację procesów w systemie na potrzeby przesyłania danych w sieci.

Identyfikacja procesów musi być niezależna od architektury systemu, czyli potrzebny jest protokół wymiany danych, który określi jej zasady.

Przesyłanie danych z procesu do procesu

Aby umożliwić komunikację między procesami, które mogą reprezentować różnych użytkowników korzystających z różnych hostów, trzeba wprowadzić jakąś identyfikację procesów w systemie na potrzeby przesyłania danych w sieci.

Identyfikacja procesów musi być niezależna od architektury systemu, czyli potrzebny jest protokół wymiany danych, który określi jej zasady.

UDP (*ang. User Datagram Protocol*)

Protokół wymiany danych należący do rodziny protokołów TCP/IP, wprowadzający prostą identyfikację procesów w oparciu o tzw. porty.

Zasady korzystania z portów UDP

Port UDP

Zasób wykorzystywany przez system operacyjny na potrzeby wymiany danych w sieci z wykorzystaniem protokołu UDP (lub TCP). Zasoby te mają (unikatowe w przypadku UDP) numery w zakresie od 0 do 65535 (numer 0 jest zarezerwowany).

Zasady korzystania z portów UDP

Port UDP

Zasób wykorzystywany przez system operacyjny na potrzeby wymiany danych w sieci z wykorzystaniem protokołu UDP (lub TCP). Zasoby te mają (unikatowe w przypadku UDP) numery w zakresie od 0 do 65535 (numer 0 jest zarezerwowany).

Nazwa „port" pochodzi od sposobu, w jaki dawniej podłączano do komputerów urządzenia wejścia/wyjścia (urządzenie było podłączane kablem do złącza, któremu był przypisany numeryczny adres wykorzystywany przez procesor do zapisu i odczytywania danych do i z urządzenia, odpowiednio).

Zasady korzystania z portów UDP

Port UDP

Zasób wykorzystywany przez system operacyjny na potrzeby wymiany danych w sieci z wykorzystaniem protokołu UDP (lub TCP). Zasoby te mają (unikatowe w przypadku UDP) numery w zakresie od 0 do 65535 (numer 0 jest zarezerwowany).

Nazwa „port" pochodzi od sposobu, w jaki dawniej podłączano do komputerów urządzenia wejścia/wyjścia (urządzenie było podłączane kablem do złącza, któremu był przypisany numeryczny adres wykorzystywany przez procesor do zapisu i odczytywania danych do i z urządzenia, odpowiednio).

Dwa różne procesy w tym samym systemie nie mogą używać tego samego portu UDP.

Rezerwowanie portów UDP

Każdy proces, który będzie wykorzystywał port UDP do wysyłania lub odbierania danych (może wykorzystywać wiele portów na raz), musi zadeklarować ten fakt i uzyskać dostęp do portu na wyłączność.

Rezerwowanie portów UDP

Każdy proces, który będzie wykorzystywał port UDP do wysyłania lub odbierania danych (może wykorzystywać wiele portów na raz), musi zadeklarować ten fakt i uzyskać dostęp do portu na wyłączność.

Otwieranie portu UDP

Operacja polegająca na przydzieleniu procesowi portu UDP o określonym numerze do wykorzystania (proces może żądać przydzielenia portu o konkretnym numerze lub pozwolić, aby jądro systemu operacyjnego wybrało dla niego numer portu).

Rezerwowanie portów UDP

Każdy proces, który będzie wykorzystywał port UDP do wysyłania lub odbierania danych (może wykorzystywać wiele portów na raz), musi zadeklarować ten fakt i uzyskać dostęp do portu na wyłączność.

Otwieranie portu UDP

Operacja polegająca na przydzieleniu procesowi portu UDP o określonym numerze do wykorzystania (proces może żądać przydzielenia portu o konkretnym numerze lub pozwolić, aby jądro systemu operacyjnego wybrało dla niego numer portu).

Zamykanie portu UDP

Operacja polegająca na zwolnieniu przez proces otwartego portu UDP.

Gniazda (*ang. socket*)

Struktury danych reprezentujące porty UDP (lub TCP) skojarzone z określonym adresem IP hosta.

Gniazda (*ang. socket*)

Struktury danych reprezentujące porty UDP (lub TCP) skojarzone z określonym adresem IP hosta.

Procesy mogą korzystać z gniazd w taki sposób, w jaki korzystają z plików.

Gniazda (*ang. socket*)

Struktury danych reprezentujące porty UDP (lub TCP) skojarzone z określonym adresem IP hosta.

Procesy mogą korzystać z gniazd w taki sposób, w jaki korzystają z plików.

Dane zapisane do gniazda przez proces są wysyłane przez sieć, natomiast dane odczytywane z gniazda pochodzą (na ogół) od innych hostów w sieci.

Gniazda (*ang. socket*)

Struktury danych reprezentujące porty UDP (lub TCP) skojarzone z określonym adresem IP hosta.

Procesy mogą korzystać z gniazd w taki sposób, w jaki korzystają z plików.

Dane zapisane do gniazda przez proces są wysyłane przez sieć, natomiast dane odczytywane z gniazda pochodzą (na ogół) od innych hostów w sieci.

W celu dokonania zapisu do gniazda UDP proces musi podać adres IP hosta i numer portu UDP odpowiadający procesowi (na tym hoście), dla którego przeznaczone są dane.

Przesyłanie danych z użyciem UDP

Dane zapisane przez proces do gniazda są dzielone na porcje, które zostaną umieszczone w różnych pakietach IP.

Przesyłanie danych z użyciem UDP

Dane zapisane przez proces do gniazda są dzielone na porcje, które zostaną umieszczone w różnych pakietach IP.

Nagłówek UDP (*ang. UDP header*)

Struktura danych dołączana do każdej porcji danych użytecznych, które mają być przesłane z wykorzystaniem UDP. Zawiera:

1. Numer portu UDP procesu zapisującego dane (*ang. source*).
2. Numer portu UDP procesu procesu, dla którego przeznaczone są dane (*ang. destination*).
3. Liczbę bajtów danych (muszą mieścić się w pakiecie IP).
4. Sumę kontrolną dla danych (słowo 16-bitowe).

Przesyłanie danych z użyciem UDP c. d.

Datagram UDP (*ang. UDP datagram*)

Pakiet IP z nagłówkiem UDP i danymi użytecznymi w polu danych.

Przesyłanie danych z użyciem UDP c. d.

Datagram UDP (*ang. UDP datagram*)

Pakiet IP z nagłówkiem UDP i danymi użytecznymi w polu danych.

Wszystkie datagramy zawierające dane są przesyłane przez sieć jako niezależne pakiety IP.

Przesyłanie danych z użyciem UDP c. d.

Datagram UDP (*ang. UDP datagram*)

Pakiet IP z nagłówkiem UDP i danymi użytecznymi w polu danych.

Wszystkie datagramy zawierające dane są przesyłane przez sieć jako niezależne pakiety IP.

Nie ma gwarancji, że zostaną one dostarczone do miejsca przeznaczenia. Ponadto mogą być przesyłane różnymi ścieżkami (*ang. path*) w sieci i mogą dotrzeć do miejsca przeznaczenia w kolejności różnej od kolejności wysyłania.

Przesyłanie danych z użyciem UDP c. d.

Datagram UDP (*ang. UDP datagram*)

Pakiet IP z nagłówkiem UDP i danymi użytecznymi w polu danych.

Wszystkie datagramy zawierające dane są przesyłane przez sieć jako niezależne pakiety IP.

Nie ma gwarancji, że zostaną one dostarczone do miejsca przeznaczenia. Ponadto mogą być przesyłane różnymi ścieżkami (*ang. path*) w sieci i mogą dotrzeć do miejsca przeznaczenia w kolejności różnej od kolejności wysyłania.

W związku z tym mówi się, że UDP jest protokołem niepewnym (*ang. unreliable protocol*).

Odbieranie danych wysłanych z użyciem UDP

Dane wysłane z użyciem UDP mogą być odebrane, gdy:

1. W sieci jest host, którego adres IP pokrywa się z adresem IP miejsca przeznaczenia datagramu.
2. Na tym hoście jest proces, który ma otwarty port UDP o numerze odpowiadającym numerowi portu UDP przeznaczenia w datagramie.
3. Proces ten podejmie próbę odczytania danych z gniazda skojarzonego z portem UDP, o którym mowa.

Odbieranie danych wysłanych z użyciem UDP

Dane wysłane z użyciem UDP mogą być odebrane, gdy:

1. W sieci jest host, którego adres IP pokrywa się z adresem IP miejsca przeznaczenia datagramu.
2. Na tym hoście jest proces, który ma otwarty port UDP o numerze odpowiadającym numerowi portu UDP przeznaczenia w datagramie.
3. Proces ten podejmie próbę odczytania danych z gniazda skojarzonego z portem UDP, o którym mowa.

Dane odczytane z datagramu (o ile w ogóle dotrze on do miejsca przeznaczenia) są wówczas przekazywane procesowi, który podjął próbę odczytania ich, jako dane wejściowe (podobnie, jak dane odczytywane z pliku).

Wady niepewności UDP

Proces odczytujący dane z gniazda nie ma informacji o tym, czy odczytuje wszystkie wysłane dane i czy są one odczytywane we właściwej kolejności.

Wady niepewności UDP

Proces odczytujący dane z gniazda nie ma informacji o tym, czy odczytuje wszystkie wysłane dane i czy są one odczytywane we właściwej kolejności.

Dlatego nie zaleca się wykorzystywania UDP do zastosowań, w których rozmiary ciągu danych do wysłania przekraczają rozmiary pola danych w pojedynczym pakiecie IP (minus długość nagłówka UDP, czyli 8 B).

Wady niepewności UDP

Proces odczytujący dane z gniazda nie ma informacji o tym, czy odczytuje wszystkie wysłane dane i czy są one odczytywane we właściwej kolejności.

Dlatego nie zaleca się wykorzystywania UDP do zastosowań, w których rozmiary ciągu danych do wysłania przekraczają rozmiary pola danych w pojedynczym pakiecie IP (minus długość nagłówka UDP, czyli 8 B).

Praktycznie oznacza to, że np. w sieci Ethernet rozmiary ciągu danych użytecznych przesyłanych z użyciem UDP nie powinny przekraczać rozmiarów pola danych w ramce (minus długość nagłówków IP i UDP, łącznie minimum 28 B).

Praktyczne zastosowania UDP

UDP stosuje się w sytuacjach, w których można sobie pozwolić na gubienie danych w drodze od nadawcy do odbiorcy.

Praktyczne zastosowania UDP

UDP stosuje się w sytuacjach, w których można sobie pozwolić na gubienie danych w drodze od nadawcy do odbiorcy.

Jest tak na przykład wtedy, gdy informacje są wysyłane okresowo i stracenie jednego uaktualnienia nie ma wielkiego znaczenia (tak jest np. w przypadku usług synchronizacji zegarów, jak NTP).

Praktyczne zastosowania UDP

UDP stosuje się w sytuacjach, w których można sobie pozwolić na gubienie danych w drodze od nadawcy do odbiorcy.

Jest tak na przykład wtedy, gdy informacje są wysyłane okresowo i stracenie jednego uaktualnienia nie ma wielkiego znaczenia (tak jest np. w przypadku usług synchronizacji zegarów, jak NTP).

Ponadto można go używać wtedy, gdy dane reprezentują pytania i odpowiedzi na nie zakodowane w postaci krótkich komunikatów. Wtedy każdy komunikat mieści się w jednym pakiecie i w razie „zgubienia" odpowiedzi można zadać ponownie to samo pytanie (tak jest np. w systemie DNS).

Konieczność zapewnienia spójności danych

Spójność (*ang. integrity*)

Własność danych polegająca na tym, że reprezentują one ciągle te same informacje, niezależnie od tego, co się z nimi dzieje (tzn. danych nie ubywa i poszczególne bity zachowują swoje wartości).

Konieczność zapewnienia spójności danych

Spójność (*ang. integrity*)

Własność danych polegająca na tym, że reprezentują one ciągle te same informacje, niezależnie od tego, co się z nimi dzieje (tzn. danych nie ubywa i poszczególne bity zachowują swoje wartości).

W większości zastosowań proces odczytujący dane powinien mieć gwarancję, że ich spójność nie została naruszona w wyniku przesyłania przez sieć (przynajmniej prawdopodobieństwo takiego zdarzenia powinno być rozsądnie małe).

Konieczność zapewnienia spójności danych

Spójność (*ang. integrity*)

Własność danych polegająca na tym, że reprezentują one ciągle te same informacje, niezależnie od tego, co się z nimi dzieje (tzn. danych nie ubywa i poszczególne bity zachowują swoje wartości).

W większości zastosowań proces odczytujący dane powinien mieć gwarancję, że ich spójność nie została naruszona w wyniku przesyłania przez sieć (przynajmniej prawdopodobieństwo takiego zdarzenia powinno być rozsądnie małe).

Zapewnienie spójności danych powinno należeć do zadań realizowanych na poziomie systemu operacyjnego.

Mechanizmy zapewniania spójności danych

Potwierdzanie (*ang. acknowledgement*) odbioru danych

Pozwala odbiorcy na poinformowanie nadawcy, że dane dotarły na miejsce przeznaczenia „w całości” (brak potwierdzenia oznacza problem).

Mechanizmy zapewniania spójności danych

Potwierdzanie (*ang. acknowledgement*) odbioru danych

Pozwala odbiorcy na poinformowanie nadawcy, że dane dotarły na miejsce przeznaczenia „w całości” (brak potwierdzenia oznacza problem).

Numerowanie przesyłanych bajtów danych

Zapewnia, że kolejność bajtów u odbiorcy będzie taka, jak u nadawcy. Pozwala na wykrycie sytuacji, w których wysłane dane nie dotarły do odbiorcy.

Mechanizmy zapewniania spójności danych

Potwierdzanie (*ang. acknowledgement*) odbioru danych

Pozwala odbiorcy na poinformowanie nadawcy, że dane dotarły na miejsce przeznaczenia „w całości" (brak potwierdzenia oznacza problem).

Numerowanie przesyłanych bajtów danych

Zapewnia, że kolejność bajtów u odbiorcy będzie taka, jak u nadawcy. Pozwala na wykrycie sytuacji, w których wysłane dane nie dotarły do odbiorcy.

Retransmisja (*ang. retransmission*)

Ponowne wysyłanie danych, o których wiadomo, że nadawca je wysłał, ale nie dotarły do odbiorcy.

TCP (*ang. Transmission Control Protocol*)

Zastosowanie wymienionych mechanizmów kontroli spójności danych wymaga posługiwania się specjalnym protokołem.

TCP (*ang. Transmission Control Protocol*)

Zastosowanie wymienionych mechanizmów kontroli spójności danych wymaga posługiwania się specjalnym protokołem.

W sieciach TCP/IP rolę tę spełnia protokół TCP.

TCP (*ang. Transmission Control Protocol*)

Zastosowanie wymienionych mechanizmów kontroli spójności danych wymaga posługiwania się specjalnym protokołem.

W sieciach TCP/IP rolę tę spełnia protokół TCP.

Jest on skonstruowany tak, że ciąg danych przesyłany z wykorzystaniem go może być traktowany jako strumień (*ang. stream*) danych (czyli tak, jak zawartość pliku) zarówno przez nadawcę, jak i przez odbiorcę.

TCP (*ang. Transmission Control Protocol*)

Zastosowanie wymienionych mechanizmów kontroli spójności danych wymaga posługiwania się specjalnym protokołem.

W sieciach TCP/IP rolę tę spełnia protokół TCP.

Jest on skonstruowany tak, że ciąg danych przesyłany z wykorzystaniem go może być traktowany jako strumień (*ang. stream*) danych (czyli tak, jak zawartość pliku) zarówno przez nadawcę, jak i przez odbiorcę.

Poza kontrolą spójności danych TCP definiuje mechanizm pozwalający odbiorcy na sterowanie szybkością wysyłania danych przez nadawcę.

Porty TCP

Przesyłanie danych z użyciem TCP wymaga przeprowadzenia określonych czynności wstępnych przez system operacujny hosta-nadawcy i hosta-odbiorcy danych.

Porty TCP

Przesyłanie danych z użyciem TCP wymaga przeprowadzenia określonych czynności wstępnych przez system operacujny hosta-nadawcy i hosta-odbiorcy danych.

TCP, podobnie jak UDP, wykorzystuje zasoby zwane portami, numerowane liczbami w zakresie od 0 do 65535 (numer 0 jest zarezerwowany).

Porty TCP

Przesyłanie danych z użyciem TCP wymaga przeprowadzenia określonych czynności wstępnych przez system operacujny hosta-nadawcy i hosta-odbiorcy danych.

TCP, podobnie jak UDP, wykorzystuje zasoby zwane portami, numerowane liczbami w zakresie od 0 do 65535 (numer 0 jest zarezerwowany).

Proces, który będzie odbierał dane z użyciem TCP musi zadeklarować ten fakt poprzez zarezerwowanie portu o określonym numerze (zwykle proces żąda konkretnego numeru portu).

Port w stanie nasłuchu

Otwieranie portu TCP

Operacja, podczas której proces otrzymuje do dyspozycji port TCP. Proces musi zadeklarować, czy będzie oczekiwał na zgłoszenie od nadawcy danych, czy sam będzie wysyłał dane.

Port w stanie nasłuchu

Otwieranie portu TCP

Operacja, podczas której proces otrzymuje do dyspozycji port TCP. Proces musi zadeklarować, czy będzie oczekiwał na zgłoszenie od nadawcy danych, czy sam będzie wysyłał dane.

Port TCP w stanie nasłuchu (*ang. listening*)

Port TCP otwarty w oczekiwaniu na zgłoszenie od (potencjalnego) nadawcy danych.

Port w stanie nasłuchu

Otwieranie portu TCP

Operacja, podczas której proces otrzymuje do dyspozycji port TCP. Proces musi zadeklarować, czy będzie oczekiwał na zgłoszenie od nadawcy danych, czy sam będzie wysyłał dane.

Port TCP w stanie nasłuchu (*ang. listening*)

Port TCP otwarty w oczekiwaniu na zgłoszenie od (potencjalnego) nadawcy danych.

Dwa różne procesy nie mogą jednocześnie otworzyć portu TCP o tym samym numerze i utrzymywać go w stanie nasłuchu.

Port w stanie nawiązywania połączenia

Proces, który będzie próbował wysyłać dane z użyciem TCP, musi zadeklarować ten fakt poprzez otworzenie portu TCP (numer portu może być wybrany przez proces lub losowy) w celu wysłania zgłoszenia do (potencjalnego) odbiorcy danych.

Port w stanie nawiązywania połączenia

Proces, który będzie próbował wysyłać dane z użyciem TCP, musi zadeklarować ten fakt poprzez otworzenie portu TCP (numer portu może być wybrany przez proces lub losowy) w celu wysłania zgłoszenia do (potencjalnego) odbiorcy danych.

Port TCP w stanie nawiązywania połączenia (*ang. connecting*)

Port TCP otwarty w celu wysłania zgłoszenia do (potencjalnego) odbiorcy danych.

Port w stanie nawiązywania połączenia

Proces, który będzie próbował wysyłać dane z użyciem TCP, musi zadeklarować ten fakt poprzez otworzenie portu TCP (numer portu może być wybrany przez proces lub losowy) w celu wysłania zgłoszenia do (potencjalnego) odbiorcy danych.

Port TCP w stanie nawiązywania połączenia (*ang. connecting*)

Port TCP otwarty w celu wysłania zgłoszenia do (potencjalnego) odbiorcy danych.

Dwa różne procesy nie mogą jednocześnie otworzyć portu TCP o tym samym numerze i utrzymywać go w stanie nawiązywania połączenia.

Nawiązywanie połączenia TCP

Jest operacją, podczas której proces dysponujący portem TCP w stanie nawiązywania połączenia wysyła zgłoszenie do procesu dysponującgo portem TCP w stanie nasłuchu i otrzymuje odpowiedź na swoje zgłoszenie.

Nawiązywanie połączenia TCP

Jest operacją, podczas której proces dysponujący portem TCP w stanie nawiązywania połączenia wysyła zgłoszenie do procesu dysponującgo portem TCP w stanie nasłuchu i otrzymuje odpowiedź na swoje zgłoszenie.

Nawiązywanie połączenia TCP jest inicjowane poprzez wysłanie pakietu o określonej strukturze z hosta-nadawcy do hosta-odbiorcy.

Nawiązywanie połączenia TCP

Jest operacją, podczas której proces dysponujący portem TCP w stanie nawiązywania połączenia wysyła zgłoszenie do procesu dysponującgo portem TCP w stanie nasłuchu i otrzymuje odpowiedź na swoje zgłoszenie.

Nawiązywanie połączenia TCP jest inicjowane poprzez wysłanie pakietu o określonej strukturze z hosta-nadawcy do hosta-odbiorcy.

Nagłówek TCP (*ang. TCP header*)

Struktura danych zawierająca informacje kontrolne wykorzystywane podczas przesyłania danych z użyciem TCP, zapisywana (przed danymi) w każdym przesyłanym pakiecie IP (ma ona co najmniej 20 B długości).

Zawartość nagłówka TCP

Segment TCP (*ang. TCP segment*)

Pakiet IP zawierający (w polu danych) nagłówek TCP i ewentualnie dane.

Zawartość nagłówka TCP

Segment TCP (*ang. TCP segment*)

Pakiet IP zawierający (w polu danych) nagłówek TCP i ewentualnie dane.

Nagłówek TCP zawiera m. in. następujące pola:

Flagi (*ang. flags*) – określają znaczenie innych pól.

SN (*ang. Sequence Number*) – numer ostatniego bajtu danych wysłanego przez nadawcę.

AN (*ang. Acknowledgement Number*) – numer następnego bajtu danych oczekiwanego przez odbiorcę.

Okno (*ang. window*) – liczba bajtów danych, jaką nadawca może wysłać do odbiorcy bez oczekiwania na potwierdzenie odbioru.

Zawartość nagłówka TCP c. d.

Poza wymienionymi polami, nagłówek TCP zawiera m. in. numery portów TCP odpowiadające procesowi-nadawcy i procesowi-odbiorcy.

Zawartość nagłówka TCP c. d.

Poza wymienionymi polami, nagłówek TCP zawiera m. in. numery portów TCP odpowiadające procesowi-nadawcy i procesowi-odbiorcy.

W celu rozpoczęcia nawiązywania połączenia TCP jądro systemu operacyjnego hosta-nadawcy wysyła do hosta-odbiorcy segment TCP bez danych zawierający w nagłówku TCP:

1. Flagę SYN ustawioną na 1.
2. Zainicjowane pole SN (powinna to być losowa wartość).
3. Zainicjowane pole Okno.
4. Numer portu TCP procesu wysyłającego zgłoszenie.
5. Numer portu TCP procesu-adresata zgłoszenia.

Przebieg nawiązywania połączenia TCP

Po otrzymaniu zgłoszenia proces-adresat może je przyjąć lub odrzucić. Jeśli je odrzuci, port TCP procesu wysyłającego zgłoszenie jest zamykany i proces ten jest informowany o błędzie.

Przebieg nawiązywania połączenia TCP

Po otrzymaniu zgłoszenia proces-adresat może je przyjąć lub odrzucić. Jeśli je odrzuci, port TCP procesu wysyłającego zgłoszenie jest zamykany i proces ten jest informowany o błędzie.

W przypadku przyjęcia zgłoszenia jądro systemu operacyjnego hosta-odbiorcy wysyła do hosta-nadawcy segment TCP bez danych zawierający w nagłówku TCP:

1. Flagi SYN i ACK ustawione na 1.
2. Zainicjowane pole SN (powinna to być losowa wartość).
3. Zainicjowane pole Okno.
4. Numer portu TCP procesu akceptującego zgłoszenie.
5. Numer portu TCP procesu, który wysłał zgłoszenie.

Nadawca i odbiorca

Po otrzymaniu segmentu TCP akceptującego zgłoszenie jądro systemu operacyjnego hosta-nadawcy wysyła do hosta odbiorcy segment TCP z pierwszą porcją danych, zawierający w nagłówku:

1. Flagę ACK ustawioną na 1.
2. Pole SN o wartości o 1 większej, niż w poprzednim segmencie wysłanym przez ten host.
3. Numer portu TCP procesu, który wysłał zgłoszenie.
4. Numer portu TCP procesu, który zaakceptował zgłoszenie.

Nadawca i odbiorca

Po otrzymaniu segmentu TCP akceptującego zgłoszenie jądro systemu operacyjnego hosta-nadawcy wysyła do hosta odbiorcy segment TCP z pierwszą porcją danych, zawierający w nagłówku:

1. Flagę ACK ustawioną na 1.
2. Pole SN o wartości o 1 większej, niż w poprzednim segmencie wysłanym przez ten host.
3. Numer portu TCP procesu, który wysłał zgłoszenie.
4. Numer portu TCP procesu, który zaakceptował zgłoszenie.

Od tego momentu proces, który wysłał zgłoszenie, nazywany jest nadawcą (*ang. sender*), a proces, który je przyjął, nazywany jest odbiorcą (*ang. receiver*) i dane mogą być przesyłane.

Sesja TCP

Sesja TCP (*ang. TCP session*)

Ciąg operacji, podczas którego przeprowadzane jest nawiązywanie połączenia TCP, przesyłanie danych oraz zamykanie połączenia.

Sesja TCP

Sesja TCP (*ang. TCP session*)

Ciąg operacji, podczas którego przeprowadzane jest nawiązywanie połączenia TCP, przesyłanie danych oraz zamykanie połączenia.

Połączenie TCP (*ang. TCP connection*)

Logiczna zależność między procesem-nadawcą i procesem-odbiorcą pozwalająca temu pierwszemu na zapisywanie danych do gniazda i temu drugiemu na odczytywanie tych danych z gniazda (po przesłaniu ich przez sieć) w takim samym porządku.

Sesja TCP

Sesja TCP (*ang. TCP session*)

Ciąg operacji, podczas którego przeprowadzane jest nawiązywanie połączenia TCP, przesyłanie danych oraz zamykanie połączenia.

Połączenie TCP (*ang. TCP connection*)

Logiczna zależność między procesem-nadawcą i procesem-odbiorcą pozwalająca temu pierwszemu na zapisywanie danych do gniazda i temu drugiemu na odczytywanie tych danych z gniazda (po przesłaniu ich przez sieć) w takim samym porządku.

Trzystopniowy uścisk dłoni (*ang. three-stage handshake*)

Operacja nawiązywania połączenia TCP polegająca na wymianie trzech inicjujących segmentów TCP między systemem (przyszłego) nadawcy i systemem (przyszłego) odbiorcy danych.

Zestawione połączenie TCP

Po przeprowadzeniu trzystopniowego uścisku dłoni połączenie TCP między nadawcą i odbiorcą uważa się za zestawione (*ang. established*). W związku z tym używane przez nich porty TCP zmieniają stan (są odtąd w stanie „zestawionego połączenia") i ich numery mogą być ponownie użyte do nasłuchiwania lub nawiązywania nowego połączenia (przez inne procesy).

Zestawione połączenie TCP

Po przeprowadzeniu trzystopniowego uścisku dłoni połączenie TCP między nadawcą i odbiorcą uważa się za zestawione (*ang. established*). W związku z tym używane przez nich porty TCP zmieniają stan (są odtąd w stanie „zestawionego połączenia") i ich numery mogą być ponownie użyte do nasłuchiwania lub nawiązywania nowego połączenia (przez inne procesy).

Po zestawieniu połączenia nadawca zapisuje dane do gniazda, a jądro systemu operacyjnego jego hosta wysyła te dane w kolejnych segmentach TCP. W każdym z tych segmentów pole SN ma wartość równą wartości pola SN z poprzedniego segmentu powiększonej o liczbę bajtów danych wysłanych w poprzednim segmencie.

Połączenie TCP – odbieranie danych

Po zestawieniu połączenia jądro systemu operacyjnego hosta odbiorcy otrzymuje kolejne segmenty TCP z danymi kontrolując numerację otrzymanych bajtów danych.

Połączenie TCP – odbieranie danych

Po zestawieniu połączenia jądro systemu operacyjnego hosta odbiorcy otrzymuje kolejne segmenty TCP z danymi kontrolując numerację otrzymanych bajtów danych.

W tym celu zapisuje w pamięci numer następnego bajtu danych do odebrania (SN z ostatnio odebranego segmentu powiększony o liczbę bajtów danych w tym segmencie).

Połączenie TCP – odbieranie danych

Po zestawieniu połączenia jądro systemu operacyjnego hosta odbiorcy otrzymuje kolejne segmenty TCP z danymi kontrolując numerację otrzymanych bajtów danych.

W tym celu zapisuje w pamięci numer następnego bajtu danych do odebrania (SN z ostatnio odebranego segmentu powiększony o liczbę bajtów danych w tym segmencie).

Jeżeli w kolejnym odebranym segmencie pole SN nagłówka ma wartość równą numerowi następnego bajtu do odebrania, numer ten jest zwiększany o liczbę bajtów danych wysłanych w tym segmencie.

Połączenie TCP – potwierdzenia

Jądro systemu operacyjnego odbiorcy okresowo potwierdza odebranie danych wysyłając do hosta-nadawcy segmenty TCP bez danych, zawierające w nagłówku TCP:

1. Flagę ACK ustawioną na 1.
2. Numer następnego bajtu do odebrania w polu AN.
3. Liczbę bajtów danych, jaką nadawca może wysłać bez oczekiwania na kolejne potwierdzenie, w polu Okno.

Połączenie TCP – potwierdzenia

Jądro systemu operacyjnego odbiorcy okresowo potwierdza odebranie danych wysyłając do hosta-nadawcy segmenty TCP bez danych, zawierające w nagłówku TCP:

1. Flagę ACK ustawioną na 1.
2. Numer następnego bajtu do odebrania w polu AN.
3. Liczbę bajtów danych, jaką nadawca może wysłać bez oczekiwania na kolejne potwierdzenie, w polu Okno.

Jeżeli nadawca nie otrzyma potwierdzenia obioru jednego z segmentów TCP wysłanych w danej sesji, to musi retransmitować (przesłać ponownie) ten segment oraz wszystkie segmenty wysłane później.

Retransmisje

Przy braku potwierdzenia odbioru wysłanych segmentów TCP z danymi w określonym czasie jądro systemu operacyjnego hosta nadawcy powtarza retransmisje tych segmentów.

Retransmisje

Przy braku potwierdzenia odbioru wysłanych segmentów TCP z danymi w określonym czasie jądro systemu operacyjnego hosta nadawcy powtarza retransmisje tych segmentów.

Przy każdej kolejnej retransmisji tego samego segmentu czas oczekiwania na potwierdzenie odbioru jest dwukrotnie dłuższy, niż w poprzedniej próbie.

Retransmisje

Przy braku potwierdzenia odbioru wysłanych segmentów TCP z danymi w określonym czasie jądro systemu operacyjnego hosta nadawcy powtarza retransmisje tych segmentów.

Przy każdej kolejnej retransmisji tego samego segmentu czas oczekiwania na potwierdzenie odbioru jest dwukrotnie dłuższy, niż w poprzedniej próbie.

Przy przekroczeniu pewnej ustalonej krytycznej wartości czasu oczekiwania na potwierdzenie odbioru segmentu TCP jądro systemu operacyjnego nadawcy uznaje, że nie ma kontaktu z odbiorcą i połączenie TCP jest jednostronnie zamykane (zamykany jest port TCP procesu-nadawcy i proces ten otrzymuje informację o błędzie).

Sterowanie przepływem danych (*ang. data flow control*)

Pole Okno w nagłówkach TCP segmentów potwierdzających odbiór danych informuje nadawcę ile bajtów danych może być wysłanych bez oczekiwania na potwierdzenie odbioru i przed rozpoczęciem retransmitowania nie potwierdzonych segmentów TCP z danymi.

Sterowanie przepływem danych (*ang. data flow control*)

Pole Okno w nagłówkach TCP segmentów potwierdzających odbiór danych informuje nadawcę ile bajtów danych może być wysłanych bez oczekiwania na potwierdzenie odbioru i przed rozpoczęciem retransmitowania nie potwierdzonych segmentów TCP z danymi.

Liczba ta obejmuje bajty danych wysłane w segmentach, których odbiór nie został jeszcze potwierdzony.

Sterowanie przepływem danych (*ang. data flow control*)

Pole Okno w nagłówkach TCP segmentów potwierdzających odbiór danych informuje nadawcę ile bajtów danych może być wysłanych bez oczekiwania na potwierdzenie odbioru i przed rozpoczęciem retransmitowania nie potwierdzonych segmentów TCP z danymi.

Liczba ta obejmuje bajty danych wysłane w segmentach, których odbiór nie został jeszcze potwierdzony.

Z pomocą tego pola jądro systemu operacyjnego hosta odbiorcy może wpływać na szybkość wysyłania segmentów TCP przez host nadawcy (i liczbę bajtów danych wysyłanych w każdym segmencie).

Sterowanie przepływem danych (*ang. data flow control*)

Pole Okno w nagłówkach TCP segmentów potwierdzających odbiór danych informuje nadawcę ile bajtów danych może być wysłanych bez oczekiwania na potwierdzenie odbioru i przed rozpoczęciem retransmitowania nie potwierdzonych segmentów TCP z danymi.

Liczba ta obejmuje bajty danych wysłane w segmentach, których odbiór nie został jeszcze potwierdzony.

Z pomocą tego pola jądro systemu operacyjnego hosta odbiorcy może wpływać na szybkość wysyłania segmentów TCP przez host nadawcy (i liczbę bajtów danych wysyłanych w każdym segmencie).

Mechanizm ten w naturalny sposób dostosowuje szybkość wysyłania danych do przepustowości łączy.

Zakończenie sesji TCP

Po wysłaniu wszystkich danych przez nadawcę i odczytaniu ich przez odbiorcę mogą oni zamienić się rolami (wtedy przydaje się wartość SN wysłana przez odbiorcę i wartość Okna wysłana przez nadawcę podczas nawiązywania połączenia TCP).

Zakończenie sesji TCP

Po wysłaniu wszystkich danych przez nadawcę i odczytaniu ich przez odbiorcę mogą oni zamienić się rolami (wtedy przydaje się wartość SN wysłana przez odbiorcę i wartość Okna wysłana przez nadawcę podczas nawiązywania połączenia TCP).

W przeciwnym wypadku połączenie TCP jest zamykane poprzez przeprowadzenie procedury analogicznej do trzystopniowego uścisku dłoni.

Zakończenie sesji TCP

Po wysłaniu wszystkich danych przez nadawcę i odczytaniu ich przez odbiorcę mogą oni zamienić się rolami (wtedy przydaje się wartość SN wysłana przez odbiorcę i wartość Okna wysłana przez nadawcę podczas nawiązywania połączenia TCP).

W przeciwnym wypadku połączenie TCP jest zamykane poprzez przeprowadzenie procedury analogicznej do trzystopniowego uścisku dłoni.

W rezultacie porty TCP nadawcy i odbiorcy są zamykane i sesja TCP kończy się.