Zaawansowana pracownia komputerowa

Rafał J. Wysocki

Instytut Fizyki Teoretycznej, Wydział Fizyki UW

22 maja 2011

Kontakt

- http://www.fuw.edu.pl/~rwys/zpk
- rwys@fuw.edu.pl
- tel. 22 55 32 263
- ul. Hoża 69, pok. 142

Materiał na ćwiczenia

1. Tworzenie stron WWW w HTML, z użyciem CSS
2. Przygotowywanie dokumentów z wykorzystaniem LaTeX-a
3. Obliczenia symboliczne z wykorzystaniem programu *Maple*^{TM}

Plan wykładu

1. WWW i HTML
   - Historia powstania i struktura WWW
   - Dokumenty hipertekstowe, wersje HTML
2. Sieci komputerowe
   - Zasada działania sieci komputerowej
   - Rodzaje sieci komputerowych
   - Sieci lokalne
   - TCP/IP i Internet
   - Usługi sieciowe
   - Historia sieci komputerowych
3. Systemy operacyjne
   - Jądro systemu
   - Warstwa *middleware*
   - Aplikacje

Hipertekst i hipermedia

Vannevar Bush, 1945

Artykuł *As We May Think* w *The Atlantic Monthly* opisujący fikcyjne urządzenie o nazwie *Memex*.

Hipertekst i hipermedia

Vannevar Bush, 1945

Artykuł *As We May Think* w *The Atlantic Monthly* opisujący fikcyjne urządzenie o nazwie *Memex*.

Theodor Nelson, 1960–1965

- Projekt Xanadu (1960) mający na celu stworzenie sieci komputerowej z prostym (intuicyjnym) interfejsem użytkownika.
- Wprowadzenie pojęć hipertekst (*ang. hypertext*) i hipermedia (*ang. hypermedia*) w latach 1963–1965.

Hipertekst i hipermedia

Vannevar Bush, 1945

Artykuł *As We May Think* w *The Atlantic Monthly* opisujący fikcyjne urządzenie o nazwie *Memex*.

Theodor Nelson, 1960–1965

- Projekt Xanadu (1960) mający na celu stworzenie sieci komputerowej z prostym (intuicyjnym) interfejsem użytkownika.
- Wprowadzenie pojęć hipertekst (*ang. hypertext*) i hipermedia (*ang. hypermedia*) w latach 1963–1965.

Hipertekst

Tekst wyświetlany na komputerze lub innym urządzeniu elektronicznym z referencjami, zwanymi hiperlinkami (*ang. hyperlink*), do innego tekstu, do którego osoba czytająca może uzyskać natychmiastowy dostęp.

NLS

Douglas Engelbart, lata 1960

System komputerowy NLS (*ang. oN-Line System*) zaprojektowany w *Augmentation Research Center* (ARC) w *Stanford Research Institute* (SRI).

NLS

Douglas Engelbart, lata 1960

System komputerowy NLS (*ang. oN-Line System*) zaprojektowany w *Augmentation Research Center* (ARC) w *Stanford Research Institute* (SRI).

Demonstracja NLS, 9 grudnia 1968, *The Mother of All Demos*

Zaprezentowano przełomowe możliwości systemu, z których część dopiero zaczyna być dostępna we współczesnych systemach komputerowych, a część jest uznawana za standard (np. stosowanie myszy o 3 przyciskach) oraz bardzo zaawansowane techniki video.

NLS

Douglas Engelbart, lata 1960

System komputerowy NLS (*ang. oN-Line System*) zaprojektowany w *Augmentation Research Center* (ARC) w *Stanford Research Institute* (SRI).

Demonstracja NLS, 9 grudnia 1968, *The Mother of All Demos*

Zaprezentowano przełomowe możliwości systemu, z których część dopiero zaczyna być dostępna we współczesnych systemach komputerowych, a część jest uznawana za standard (np. stosowanie myszy o 3 przyciskach) oraz bardzo zaawansowane techniki video.

W NLS możliwe było przetwarzanie informacji w postaci hipermedialnej (zgodnie z ideą T. Nelsona), w tym także modyfikowanie ich z pomocą programu *Journal*, działającego w analogii do współczesnego *Wiki*.

Enquire

Sir Timothy Berners-Lee, 1980

Zaczyna pracować w CERN i widzi potrzebę wprowadzenia systemu pozwalającego na łatwe odnajdywanie i udostępnianie informacji przez badaczy.
Po przeczytaniu prac T. Nelsona, z pomocą Roberta Cailliau tworzy prototypowy system o nazwie *Enquire* (przypomina on *Journal* z NLS).

Enquire

Sir Timothy Berners-Lee, 1980

Zaczyna pracować w CERN i widzi potrzebę wprowadzenia systemu pozwalającego na łatwe odnajdywanie i udostępnianie informacji przez badaczy.
Po przeczytaniu prac T. Nelsona, z pomocą Roberta Cailliau tworzy prototypowy system o nazwie *Enquire* (przypomina on *Journal* z NLS).

T. Berners-Lee wraca do CERN, 1984

Pracuje nad węzłem internetowym CERN (największy węzeł internetowy w Europie w 1989 r.). Widzi możliwość połączenia idei hipertekstu z funkcjonalnością Internetu.

World Wide Web

T. Berners-Lee, 1990

Tworzy prototypy przeglądarki o nazwie *WorldWideWeb* oraz serwera o nazwie *httpd* (*ang. HyperText Transfer Protocol daemon*). Projekt został zgłoszony 12 grudnia 1990 r., prace rozpoczęły się następnego dnia, oprogramowanie zostało stworzone w okresie Świąt Bożego Narodzenia 1990–1991.

World Wide Web

T. Berners-Lee, 1990

Tworzy prototypy przeglądarki o nazwie *WorldWideWeb* oraz serwera o nazwie *httpd* (*ang. HyperText Transfer Protocol daemon*). Projekt został zgłoszony 12 grudnia 1990 r., prace rozpoczęły się następnego dnia, oprogramowanie zostało stworzone w okresie Świąt Bożego Narodzenia 1990–1991.

T. Berners-Lee, grudzień 1991

Pierwsza specyfikacja HTML (*ang. HyperText Markup Language*).

World Wide Web

T. Berners-Lee, 1990

Tworzy prototypy przeglądarki o nazwie *WorldWideWeb* oraz serwera o nazwie *httpd* (*ang. HyperText Transfer Protocol daemon*). Projekt został zgłoszony 12 grudnia 1990 r., prace rozpoczęły się następnego dnia, oprogramowanie zostało stworzone w okresie Świąt Bożego Narodzenia 1990–1991.

T. Berners-Lee, grudzień 1991

Pierwsza specyfikacja HTML (*ang. HyperText Markup Language*).

CERN, 30 kwietnia 1993

Ogłoszono, że kod oprogramowania będzie dostępny bezpłatnie, podobnie jak specyfikacje HTML i HTTP, których przyszłe implementacje również zostały zwolnione z wszelkich opłat.

Mosaic i *W3C*

Marc Andreessen, 1993

W *National Center for Supercomputing Applications* na Uniwersytecie Illinois w Urbana-Champaign (NCSA-UIUC) powstaje graficzna przeglądarka WWW o nazwie *Mosaic*. Rozpoczyna się łączenie grafiki z tekstem na stronach WWW.

Mosaic i *W3C*

Marc Andreessen, 1993

W *National Center for Supercomputing Applications* na Uniwersytecie Illinois w Urbana-Champaign (NCSA-UIUC) powstaje graficzna przeglądarka WWW o nazwie *Mosaic*. Rozpoczyna się łączenie grafiki z tekstem na stronach WWW.

T. Berners-Lee, październik 1994

Założone zostaje *World Wide Web Consortium* (W3C). Powstaje ono w *Massachusetts Institute of Technology Laboratory for Computer Science* (MIT/LCS), przy wsparciu Komisji Europejskiej oraz *Defense Advanced Research Projects Agency* (DARPA), jako organizacja międzynarodowa zajmująca się standaryzacją WWW.

Czym jest WWW

System wzajemnie powiązanych dokumentów hipertekstowych (hipermedialnych) dostępnych za pośrednictwem Internetu.

Czym jest WWW

System wzajemnie powiązanych dokumentów hipertekstowych (hipermedialnych) dostępnych za pośrednictwem Internetu.

Używając przeglądarki WWW (*ang. web browser*) można czytać tak zwane strony WWW (*ang. web pages*), zawierające tekst, obrazy, filmy i inne multimedia oraz nawigować między nimi z pomocą hiperlinków.

Czym jest WWW

System wzajemnie powiązanych dokumentów hipertekstowych (hipermedialnych) dostępnych za pośrednictwem Internetu.

Używając przeglądarki WWW (*ang. web browser*) można czytać tak zwane strony WWW (*ang. web pages*), zawierające tekst, obrazy, filmy i inne multimedia oraz nawigować między nimi z pomocą hiperlinków.

Uniform Resource Indentifier (URI), Uniform Resource Locator (URL)

„Odnośnik" pozwalający na zlokalizowanie danego zasobu w WWW (w ogólności w Internecie), zawierający informacje o:

1. sposobie jego udostępnienia (np. `http://`),
2. nazwie udostępniającego go węzła sieci (np. `www.fuw.edu.pl`),
3. lokalizacji zasobu w obrębie tego węzła (np. `/~rwys/zpk`).

WWW jako usługa sieciowa

Hiperlinki tworzy się włączając do dokumentów (w odpowiedni sposób) URI zasobów reprezentujących powiązane dokumenty. Zasoby te są udostępniane przez węzły sieci zwane serwerami WWW (*ang. web server*), a przy ich przesyłaniu między swerwerami i przeglądarkami WWW wykorzystywany jest protokół HTTP.

WWW jako usługa sieciowa

Hiperlinki tworzy się włączając do dokumentów (w odpowiedni sposób) URI zasobów reprezentujących powiązane dokumenty. Zasoby te są udostępniane przez węzły sieci zwane serwerami WWW (*ang. web server*), a przy ich przesyłaniu między swerwerami i przeglądarkami WWW wykorzystywany jest protokół HTTP.

WWW można traktować jako przestrzeń informacyjną (*ang. information space*), w której interesujące nas rzeczy (zasoby) są oznaczane z pomocą globalnych identyfikatorów (URI).

WWW jako usługa sieciowa

Hiperlinki tworzy się włączając do dokumentów (w odpowiedni sposób) URI zasobów reprezentujących powiązane dokumenty. Zasoby te są udostępniane przez węzły sieci zwane serwerami WWW (*ang. web server*), a przy ich przesyłaniu między swerwerami i przeglądarkami WWW wykorzystywany jest protokół HTTP.

WWW można traktować jako przestrzeń informacyjną (*ang. information space*), w której interesujące nas rzeczy (zasoby) są oznaczane z pomocą globalnych identyfikatorów (URI).

WWW nie jest Internetem, tylko usługą (*ang. service*) działającą *w oparciu o* Internet.

Skąd wywodzi się HTML

Markup

W poligrafii : Adnotacje na rękopisie naniesione przez redaktora.

W informatyce : Adnotacje określające *sposób wyświetlania* lub *drukowania* tekstu.

Skąd wywodzi się HTML

Markup

W poligrafii : Adnotacje na rękopisie naniesione przez redaktora.

W informatyce : Adnotacje określające *sposób wyświetlania* lub *drukowania* tekstu.

Markup language

System umieszczania adnotacji w tekście, określający sposób wyświetlania bądź drukowania go, składniowo odróżnialny od tego tekstu. Zwykle polega na łączeniu *treści*, zapisanej jako zwykły tekst (*ang. plain text*), z określonymi znacznikami strukturalnymi (*ang. structural marker*).

Skąd wywodzi się HTML

Markup

W poligrafii : Adnotacje na rękopisie naniesione przez redaktora.

W informatyce : Adnotacje określające *sposób wyświetlania* lub *drukowania* tekstu.

Markup language

System umieszczania adnotacji w tekście, określający sposób wyświetlania bądź drukowania go, składniowo odróżnialny od tego tekstu. Zwykle polega na łączeniu *treści*, zapisanej jako zwykły tekst (*ang. plain text*), z określonymi znacznikami strukturalnymi (*ang. structural marker*).

HTML (*ang. HyperText Markup Language*)

System zapisu dokumentów z wykorzystaniem znaczników (*ang. tag*) stworzony na potrzeby WWW.

SGML

SGML (*ang. Standard Generalized Markup Language*) ISO 8879:1986

Standard międzynarodowy określający sposób definiowania uogólnionych języków adnotacyjnych (*ang. markup languages*). Wywodzi się z języka GML opracowanego w latach 1960 w IBM.

SGML

SGML (*ang. Standard Generalized Markup Language*) ISO 8879:1986

Standard międzynarodowy określający sposób definiowania uogólnionych języków adnotacyjnych (*ang. markup languages*). Wywodzi się z języka GML opracowanego w latach 1960 w IBM.

SGML powstał przede wszystkim w celu umożliwienia wymiany dokumentów elektronicznych (*ang. machine-readable documents*) w przemyśle, instytucjach rządowych oraz prawniczych w USA.

SGML

SGML (*ang. Standard Generalized Markup Language*) ISO 8879:1986

Standard międzynarodowy określający sposób definiowania uogólnionych języków adnotacyjnych (*ang. markup languages*). Wywodzi się z języka GML opracowanego w latach 1960 w IBM.

SGML powstał przede wszystkim w celu umożliwienia wymiany dokumentów elektronicznych (*ang. machine-readable documents*) w przemyśle, instytucjach rządowych oraz prawniczych w USA.

W 1998 roku na podstawie SGML został opracowany XML (*ang. eXtensible Markup Language*), który jest *podzbiorem* (*ang. subset*) lub *profilem* (*ang. profile*) SGML.

SGML/XML i HTML

W HTML wykorzystywane są elementy pochodzące z SGML/XML, choć jest on znacznie prostszy (w zasadzie można *zdefiniować* HTML stosując reguły z SGML/XML).

SGML/XML i HTML

W HTML wykorzystywane są elementy pochodzące z SGML/XML, choć jest on znacznie prostszy (w zasadzie można *zdefiniować* HTML stosując reguły z SGML/XML).

Znaczniki (*ang. tag*) HTML

Służą do umieszczania adnotacji w tekście. Mogą mieć część otwierającą i zamykająca lub tylko otwierającą (znaczniki jednoczęściowe).

SGML/XML i HTML

W HTML wykorzystywane są elementy pochodzące z SGML/XML, choć jest on znacznie prostszy (w zasadzie można *zdefiniować* HTML stosując reguły z SGML/XML).

Znaczniki (*ang. tag*) HTML

Służą do umieszczania adnotacji w tekście. Mogą mieć część otwierającą i zamykająca lub tylko otwierającą (znaczniki jednoczęściowe).

Część otwierająca znacznika ma postać `<nazwa>`, a część zamykająca ma postać `</nazwa>`, przy czym nazwy znaczników HTML są ustalone (nie można wprowadzać nowych znaczników).

SGML/XML i HTML

W HTML wykorzystywane są elementy pochodzące z SGML/XML, choć jest on znacznie prostszy (w zasadzie można *zdefiniować* HTML stosując reguły z SGML/XML).

Znaczniki (*ang. tag*) HTML

Służą do umieszczania adnotacji w tekście. Mogą mieć część otwierającą i zamykająca lub tylko otwierającą (znaczniki jednoczęściowe).

Część otwierająca znacznika ma postać `<nazwa>`, a część zamykająca ma postać `</nazwa>`, przy czym nazwy znaczników HTML są ustalone (nie można wprowadzać nowych znaczników).

Znaczniki jednoczęściowa mają tylko część otwierającą, lecz można je zapisywać w postaci `<nazwa/>`.

Atrybuty znaczników

```
<nazwa atrybut='wartość' ...> </nazwa>
```

Atrubuty znaczników mają następujące funkcje:

1. Doprecyzowują interpretację niektórych znaczników (np. <a>).
2. Określają własności „zawartości" znacznika (np. <td>, <p>).
3. Określają własności reprezentowanego obiektu (np. <img>).

Atrybuty znaczników

`<nazwa atrybut='wartość' ...> </nazwa>`

Atrubuty znaczników mają następujące funkcje:

1. Doprecyzowują interpretację niektórych znaczników (np. <a>).
2. Określają własności „zawartości" znacznika (np. <td>, <p>).
3. Określają własności reprezentowanego obiektu (np. <img>).

Znacznik <a> (*ang. anchor*) z atrubutem `href` (*ang. hyperreference*) oznacza hiperlink, w którym wartość `href` określa docelowy URI.

Atrybuty znaczników

`<nazwa atrybut='wartość' ...> </nazwa>`

Atrubuty znaczników mają następujące funkcje:

1. Doprecyzowują interpretację niektórych znaczników (np. <a>).
2. Określają własności „zawartości" znacznika (np. <td>, <p>).
3. Określają własności reprezentowanego obiektu (np. <img>).

Znacznik <a> (*ang. anchor*) z atrubutem `href` (*ang. hyperreference*) oznacza hiperlink, w którym wartość `href` określa docelowy URI.

W HTML zbiór możliwych atrybutów dla każdego znacznika jest ustalony (nie można wprowadzać nowych atrybutów). Zbiór możliwych wartości dla każdego atrybutu również jest określony.

Struktura dokumentu w HTML

Znaczniki HTML określają strukturę dokumentu oraz sposób wyświetlania (drukowania) tekstu, a także zależności (połączenia) między różnymi dokumentami oraz różnymi częściami jednego dokumentu.

Struktura dokumentu w HTML

Znaczniki HTML określają strukturę dokumentu oraz sposób wyświetlania (drukowania) tekstu, a także zależności (połączenia) między różnymi dokumentami oraz różnymi częściami jednego dokumentu.

`<html> </html>`

Początek i koniec dokumentu.

Struktura dokumentu w HTML

Znaczniki HTML określają strukturę dokumentu oraz sposób wyświetlania (drukowania) tekstu, a także zależności (połączenia) między różnymi dokumentami oraz różnymi częściami jednego dokumentu.

`<html> </html>`

Początek i koniec dokumentu.

`<head> </head>`

Początek i koniec nagłówka (*ang. header*) dokumentu.

Struktura dokumentu w HTML

Znaczniki HTML określają strukturę dokumentu oraz sposób wyświetlania (drukowania) tekstu, a także zależności (połączenia) między różnymi dokumentami oraz różnymi częściami jednego dokumentu.

`<html> </html>`

Początek i koniec dokumentu.

`<head> </head>`

Początek i koniec nagłówka (*ang. header*) dokumentu.

`<body> </body>`

Początek i koniec treści (*ang. body*) dokumentu.

Nagłówek dokumentu HTML

`<title> </title>`

Tytuł dokumentu (ciąg znaków wyświetlany na „pasku” przeglądarki).

Nagłówek dokumentu HTML

`<title> </title>`

Tytuł dokumentu (ciąg znaków wyświetlany na „pasku” przeglądarki).

`<meta />`

Definicje donoszące się do całej zawartości dokumentu, np.:

```
<meta http-equiv='Content-type' content='text/html; charset=utf-8' />
<meta http-equiv='Content-language' content='pl' />
```

Nagłówek dokumentu HTML

<title> </title>

Tytuł dokumentu (ciąg znaków wyświetlany na „pasku” przeglądarki).

<meta />

Definicje donoszące się do całej zawartości dokumentu, np.:

```
<meta http-equiv='Content-type' content='text/html; charset=utf-8' />
<meta http-equiv='Content-language' content='pl' />
```

<link />

Arkusze stylów itp., np.

```
<link href='css/style.css' rel='stylesheet' type='text/css' />
```

Arkusze stylów

CSS (*ang. Cascading Style Sheet*)

Dokument określający sposób wyświetlania (drukowania) różnych elementów dokumentu opisanych znacznikami HTML (lub innego języka adnotacyjnego).

Arkusze stylów

CSS (*ang. Cascading Style Sheet*)

Dokument określający sposób wyświetlania (drukowania) różnych elementów dokumentu opisanych znacznikami HTML (lub innego języka adnotacyjnego).

Format arkusza stylu

`selektor {własność:wartość; [własność:wartość; ...]}` np.

- `h2 {color:red;}` („selektor" jest nazwą znacznika HTML)
- `.important {color:red;}` („selektor" jest klasą elementu)

Arkusze stylów

CSS (*ang. Cascading Style Sheet*)

Dokument określający sposób wyświetlania (drukowania) różnych elementów dokumentu opisanych znacznikami HTML (lub innego języka adnotacyjnego).

Format arkusza stylu

`selektor {własność:wartość; [własność:wartość; ...]}` np.

- `h2 {color:red;}` („selektor" jest nazwą znacznika HTML)
- `.important {color:red;}` („selektor" jest klasą elementu)

Klasy (*ang. class*) elementów

Klasę można określić m. in. dla znaczników `<h[1-6]>`, `<p>`, `<td>`, `<th>`, np. `<h2 class='important'>Ważny nagłówek</h2>`

Zastosowanie arkuszy stylów

Arkusze stylów służą do rozdzielenia zawartości merytorycznej (*ang. contents*) dokumentu od jego prezentacji (*ang. presentation*).

Zastosowanie arkuszy stylów

Arkusze stylów służą do rozdzielenia zawartości merytorycznej (*ang. contents*) dokumentu od jego prezentacji (*ang. presentation*).

Zmieniając arkusz stylu można zmodyfikować wygląd wszystkich elementów określonego rodzaju (np. kolor linków) w całym dokumencie bez konieczności ingerowania w jego treść.

Zastosowanie arkuszy stylów

Arkusze stylów służą do rozdzielenia zawartości merytorycznej (*ang. contents*) dokumentu od jego prezentacji (*ang. presentation*).

Zmieniając arkusz stylu można zmodyfikować wygląd wszystkich elementów określonego rodzaju (np. kolor linków) w całym dokumencie bez konieczności ingerowania w jego treść.

Przenośność arkuszy stylów

Nie wszystkie przeglądarki WWW interpretują CSS w jednakowy sposób (zwłaszcza dotyczy to starszych przeglądarek). Współczesne przeglądarki w większości implementując CSS 2.1.

Wersje HTML

HTML 3.2, styczeń 1997

Pierwsza wersja HTML opracowana całkowicie przez W3C.

Wersje HTML

HTML 3.2, styczeń 1997

Pierwsza wersja HTML opracowana całkowicie przez W3C.

HTML 4.0, 1997–1998

Wprowadzono wiele typów elementów i atrybutów rozpoznawanych przez określone przeglądarki, ograniczono zastosowanie znaczników wizualnych pochodzących z przeglądarek *Netscape* na korzyść arkuszy stylów. HTML 4 jest aplikacją SGML zgodną ze standardem ISO 8879.

Wersje HTML

HTML 3.2, styczeń 1997

Pierwsza wersja HTML opracowana całkowicie przez W3C.

HTML 4.0, 1997–1998

Wprowadzono wiele typów elementów i atrybutów rozpoznawanych przez określone przeglądarki, ograniczono zastosowanie znaczników wizualnych pochodzących z przeglądarek *Netscape* na korzyść arkuszy stylów. HTML 4 jest aplikacją SGML zgodną ze standardem ISO 8879.

HTML 4.01, 1999–2001

Poprawki w stosunku do wersji 4.0, standard ISO/IEC 15445:2000.

Wersje HTML

HTML 3.2, styczeń 1997

Pierwsza wersja HTML opracowana całkowicie przez W3C.

HTML 4.0, 1997–1998

Wprowadzono wiele typów elementów i atrybutów rozpoznawanych przez określone przeglądarki, ograniczono zastosowanie znaczników wizualnych pochodzących z przeglądarek *Netscape* na korzyść arkuszy stylów. HTML 4 jest aplikacją SGML zgodną ze standardem ISO 8879.

HTML 4.01, 1999–2001

Poprawki w stosunku do wersji 4.0, standard ISO/IEC 15445:2000.

HTML 5, ciągle rozwijana

Nowe elementy związane z obsługą multimediów.

Na czym polega problem

Chcemy przesyłać informacje w postaci (reprezentacji) binarnej na odległość.

Na czym polega problem

Chcemy przesyłać informacje w postaci (reprezentacji) binarnej na odległość.

Nadawca informacji dysponuje ciągiem bitów, który ma być dostarczony do odbiorcy w niezmienionej postaci (o ile jest to możliwe) oraz odpowiednio krótkim czasie (im szybciej, tym lepiej).

Na czym polega problem

Chcemy przesyłać informacje w postaci (reprezentacji) binarnej na odległość.

Nadawca informacji dysponuje ciągiem bitów, który ma być dostarczony do odbiorcy w niezmienionej postaci (o ile jest to możliwe) oraz odpowiednio krótkim czasie (im szybciej, tym lepiej).

Dane te powinny zostać zakodowane (*ang. encoded*) w taki sposób, aby mogły pokonać przestrzeń między nadawcą i odbiorcą oraz aby mogły być odtworzone przez odbiorcę.

Na czym polega problem

Chcemy przesyłać informacje w postaci (reprezentacji) binarnej na odległość.

Nadawca informacji dysponuje ciągiem bitów, który ma być dostarczony do odbiorcy w niezmienionej postaci (o ile jest to możliwe) oraz odpowiednio krótkim czasie (im szybciej, tym lepiej).

Dane te powinny zostać zakodowane (*ang. encoded*) w taki sposób, aby mogły pokonać przestrzeń między nadawcą i odbiorcą oraz aby mogły być odtworzone przez odbiorcę.

Zwykle w tym celu używane są zaburzenia rozchodzące się w postaci fal (*ang. wave*), zwane sygnałami (*ang. signal*).

Sygnały w sieciach komputerowych

W sieciach komputerowych wykorzystywane są (prawie wyłącznie) sygnały elektromagnetyczne (EM), ze względu na stosunkowo łatwe posługiwanie się nimi i dużą odporność na zakłócenia.

Sygnały w sieciach komputerowych

W sieciach komputerowych wykorzystywane są (prawie wyłącznie) sygnały elektromagnetyczne (EM), ze względu na stosunkowo łatwe posługiwanie się nimi i dużą odporność na zakłócenia.

Nadawca

Wytwarza sygnały EM w ściśle określony sposób, aby po dotarciu do miejsca przeznaczenia można było odczytać z nich zakodowane bity danych.

Sygnały w sieciach komputerowych

W sieciach komputerowych wykorzystywane są (prawie wyłącznie) sygnały elektromagnetyczne (EM), ze względu na stosunkowo łatwe posługiwanie się nimi i dużą odporność na zakłócenia.

Nadawca

Wytwarza sygnały EM w ściśle określony sposób, aby po dotarciu do miejsca przeznaczenia można było odczytać z nich zakodowane bity danych.

Odbiorca

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

Protokoły komunikacyjne

Nadawca danych musi umówić się z odbiorcą co do tego jak mają być interpretowane różne sygnały (np. jaki sygnał będzie oznaczać bit 0, a jaki – bit 1).

Protokoły komunikacyjne

Nadawca danych musi umówić się z odbiorcą co do tego jak mają być interpretowane różne sygnały (np. jaki sygnał będzie oznaczać bit 0, a jaki – bit 1).

Taka nieformalna umowa może wystarczyć dla jednego odbiorcy i jednego nadawcy.

Protokoły komunikacyjne

Nadawca danych musi umówić się z odbiorcą co do tego jak mają być interpretowane różne sygnały (np. jaki sygnał będzie oznaczać bit 0, a jaki – bit 1).

Taka nieformalna umowa może wystarczyć dla jednego odbiorcy i jednego nadawcy.

Przy większej liczbie potencjalnych odbiorców i nadawców trzeba formlalnie określić zasady współdziałania.

Protokoły komunikacyjne

Nadawca danych musi umówić się z odbiorcą co do tego jak mają być interpretowane różne sygnały (np. jaki sygnał będzie oznaczać bit 0, a jaki – bit 1).

Taka nieformalna umowa może wystarczyć dla jednego odbiorcy i jednego nadawcy.

Przy większej liczbie potencjalnych odbiorców i nadawców trzeba formlalnie określić zasady współdziałania.

Protokół komunikacyjny (*ang. communications protocol*)

Formalna specyfikacja zasad współdziałania między użytkownikami systemu komunikacji (np. sieci komputerowej).

Sieć komputerowa i węzły sieci

Sieć komputerowa (*ang. computer network*)

Zbiór komputerów, które mogą przesyłać dane między sobą zgodnie z określonym protokołem komunikacyjnym (lub większą liczbą takich protokołów).

Sieć komputerowa i węzły sieci

Sieć komputerowa (*ang. computer network*)

Zbiór komputerów, które mogą przesyłać dane między sobą zgodnie z określonym protokołem komunikacyjnym (lub większą liczbą takich protokołów).

Węzeł sieci (*ang. network node*)

Komputer (lub ogólnie urządzenie), który może być źródłem (*ang. source*) lub miejscem przeznaczenia (*ang. destination*) danych w sieci.

Sieć komputerowa i węzły sieci

Sieć komputerowa (*ang. computer network*)

Zbiór komputerów, które mogą przesyłać dane między sobą zgodnie z określonym protokołem komunikacyjnym (lub większą liczbą takich protokołów).

Węzeł sieci (*ang. network node*)

Komputer (lub ogólnie urządzenie), który może być źródłem (*ang. source*) lub miejscem przeznaczenia (*ang. destination*) danych w sieci.

Adres sieciowy (*ang. network address*)

Identyfikator określający jednoznacznie węzeł sieci (dla sieci komputerowej jest to ciąg bitów, który może być interpretowany jako liczba całkowita w reprezentacji bezznakowej).

Łącza i przełączanie obwodów

Łącze (*ang. link*)

Infrastruktura (np. okablowanie) i urządzenia pozwalające na przesyłanie sygnałów między dwiema różnymi częściami sieci.

Łącza i przełączanie obwodów

Łącze (*ang. link*)

Infrastruktura (np. okablowanie) i urządzenia pozwalające na przesyłanie sygnałów między dwiema różnymi częściami sieci.

BR (*ang. bit rate*)

Liczba bitów, które mogą być przesłane przez dane łącze w jednostce czasu.

Łącza i przełączanie obwodów

Łącze (*ang. link*)

Infrastruktura (np. okablowanie) i urządzenia pozwalające na przesyłanie sygnałów między dwiema różnymi częściami sieci.

BR (*ang. bit rate*)

Liczba bitów, które mogą być przesłane przez dane łącze w jednostce czasu.

Przełączanie obwodów (*ang. circuit switching*)

Zestawianie łączy o ustalonych parametrach (np. BR) między parami węzłów sieci do wyłącznego użytku przez pewien czas.

Łącza i przełączanie obwodów

Łącze (*ang. link*)

Infrastruktura (np. okablowanie) i urządzenia pozwalające na przesyłanie sygnałów między dwiema różnymi częściami sieci.

BR (*ang. bit rate*)

Liczba bitów, które mogą być przesłane przez dane łącze w jednostce czasu.

Przełączanie obwodów (*ang. circuit switching*)

Zestawianie łączy o ustalonych parametrach (np. BR) między parami węzłów sieci do wyłącznego użytku przez pewien czas.

Nieskuteczne przy dużej liczbie węzłów sieci.

Pakiety

Przełączanie pakietów (*ang. packet switching*)

Polega na tym, że łącza są zestawiane na stałe (lub na długi czas), a dane są przesyłane w blokach o odpowiednim rozmiarze (długości), zwanych pakietami (*ang. packet*), z których każdy, oprócz danych użytecznych, zawiera informacje o ich źródle i miejscu przeznaczenia (np. ich adresy sieciowe) oraz inne informacje umożliwiające bądź ułatwiające dostarczenie danych użytecznych do miejsca przeznaczenia.

Pakiety

Przełączanie pakietów (*ang. packet switching*)

Polega na tym, że łącza są zestawiane na stałe (lub na długi czas), a dane są przesyłane w blokach o odpowiednim rozmiarze (długości), zwanych pakietami (*ang. packet*), z których każdy, oprócz danych użytecznych, zawiera informacje o ich źródle i miejscu przeznaczenia (np. ich adresy sieciowe) oraz inne informacje umożliwiające bądź ułatwiające dostarczenie danych użytecznych do miejsca przeznaczenia.

Leonard Kleinrock, 1961–1962

Pakiety

Przełączanie pakietów (*ang. packet switching*)

Polega na tym, że łącza są zestawiane na stałe (lub na długi czas), a dane są przesyłane w blokach o odpowiednim rozmiarze (długości), zwanych pakietami (*ang. packet*), z których każdy, oprócz danych użytecznych, zawiera informacje o ich źródle i miejscu przeznaczenia (np. ich adresy sieciowe) oraz inne informacje umożliwiające bądź ułatwiające dostarczenie danych użytecznych do miejsca przeznaczenia.

Leonard Kleinrock, 1961–1962

Nagłówek (*ang. header*) i stopka (*ang. footer*) pakietu

Części pakietu zawierające informacje nie będące danymi użytecznymi i umieszczane w pakiecie odpowiednio przed oraz za blokiem danych (*ang. data payload*).

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 – przykład

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

Pakiety i transmisja głosu – przykład

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
	...		

Sieci z przełączaniem pakietów

Sieć z przełączaniem pakietów (*ang. packet-switched network*)

Sieć, w której wykorzystywana jest metoda komunikacji polegająca na przełączaniu pakietów.

Sieci z przełączaniem pakietów

Sieć z przełączaniem pakietów (*ang. packet-switched network*)

Sieć, w której wykorzystywana jest metoda komunikacji polegająca na przełączaniu pakietów.

W sieci z przełączaniem pakietów jedno łącze może obsługiwać pakiety pochodzące z wielu różnych źródeł i przesyłane do wielu różnych miejsc przeznaczenia.

Sieci z przełączaniem pakietów

Sieć z przełączaniem pakietów (*ang. packet-switched network*)

Sieć, w której wykorzystywana jest metoda komunikacji polegająca na przełączaniu pakietów.

W sieci z przełączaniem pakietów jedno łącze może obsługiwać pakiety pochodzące z wielu różnych źródeł i przesyłane do wielu różnych miejsc przeznaczenia.

Dzięki temu możliwe jest lepsze wykorzystanie przepustowości (*ang. throughput*) dostępnych łączy, która bywa także nazywana szerokością pasma (*ang. bandwidth*).

Sieci z przełączaniem pakietów

Sieć z przełączaniem pakietów (*ang. packet-switched network*)

Sieć, w której wykorzystywana jest metoda komunikacji polegająca na przełączaniu pakietów.

W sieci z przełączaniem pakietów jedno łącze może obsługiwać pakiety pochodzące z wielu różnych źródeł i przesyłane do wielu różnych miejsc przeznaczenia.

Dzięki temu możliwe jest lepsze wykorzystanie przepustowości (*ang. throughput*) dostępnych łączy, która bywa także nazywana szerokością pasma (*ang. bandwidth*).

Wszystkie współczesne sieci komputerowe oraz większość sieci telefonicznych to sieci z przełączaniem pakietów.

Sieci lokalne i rozległe

Sieć lokalna, LAN (*ang. local area network*)

Sieć komputerowa, w której każdy węzeł może przesyłać dane do każdego innego węzła bez pośrednictwa innych węzłów sieci (tzn. wszystkie węzły LAN są połączone *w pewnym sensie* bezpośrednio ze sobą nawzajem).

Sieci lokalne i rozległe

Sieć lokalna, LAN (*ang. local area network*)

Sieć komputerowa, w której każdy węzeł może przesyłać dane do każdego innego węzła bez pośrednictwa innych węzłów sieci (tzn. wszystkie węzły LAN są połączone *w pewnym sensie* bezpośrednio ze sobą nawzajem).

Sieć rozległa, WAN (*ang. wide area network*)

Sieć komputerowa, w której istnieje co najmniej jedna para węzłów taka, że przesyłanie danych z jednego z nich do drugiego wymaga pośrednictwa trzeciego węzła (lub większej liczby węzłów).

Sieci lokalne i rozgległe

Sieć lokalna, LAN (*ang. local area network*)

Sieć komputerowa, w której każdy węzeł może przesyłać dane do każdego innego węzła bez pośrednictwa innych węzłów sieci (tzn. wszystkie węzły LAN są połączone *w pewnym sensie* bezpośrednio ze sobą nawzajem).

Sieć rozległa, WAN (*ang. wide area network*)

Sieć komputerowa, w której istnieje co najmniej jedna para węzłów taka, że przesyłanie danych z jednego z nich do drugiego wymaga pośrednictwa trzeciego węzła (lub większej liczby węzłów).

Ruter (*ang. router*)

Węzeł sieci pośredniczący w przesyłaniu danych w sieci rozległej.

Sieć Ethernet

Ethernet

Pierwszy rozpowszechniony standard sieci lokalnej, zakładał istnienie medium transmisyjnego wspólnego dla wszystkich węzłów sieci, zwanych stacjami (*ang. station*).

Sieć Ethernet

Ethernet

Pierwszy rozpowszechniony standard sieci lokalnej, zakładał istnienie medium transmisyjnego wspólnego dla wszystkich węzłów sieci, zwanych stacjami (*ang. station*).

Robert Metcalfe, David Boggs, 1973–1976

Palo Alto Research Center (PARC), firma Xerox.

Sieć Ethernet

Ethernet

Pierwszy rozpowszechniony standard sieci lokalnej, zakładał istnienie medium transmisyjnego wspólnego dla wszystkich węzłów sieci, zwanych stacjami (*ang. station*).

Robert Metcalfe, David Boggs, 1973–1976

Palo Alto Research Center (PARC), firma Xerox.

Ramki (*ang. frame*)

W sieci Ethernet dane są przesyłane w pakietach zwanych ramkami.

Sieć Ethernet

Ethernet

Pierwszy rozpowszechniony standard sieci lokalnej, zakładał istnienie medium transmisyjnego wspólnego dla wszystkich węzłów sieci, zwanych stacjami (*ang. station*).

Robert Metcalfe, David Boggs, 1973–1976

Palo Alto Research Center (PARC), firma Xerox.

Ramki (*ang. frame*)

W sieci Ethernet dane są przesyłane w pakietach zwanych ramkami.

W „klasycznej” sieci Ethernet każda ramka wysłana przez dowolną stację może być odebrana przez każdą z pozostałych stacji.

Zasada działania sieci Ethernet

Kolizja sygnałów (*ang. signal collision*)

Destruktywna interferencja fal EM przenoszących sygnały.

Zasada działania sieci Ethernet

Kolizja sygnałów (*ang. signal collision*)

Destruktywna interferencja fal EM przenoszących sygnały.

CSMA/CD (*ang. Carrier Sense, Multiple Access with Collision Detection*)

Technika pozwalająca na wykorzystywanie wspólnego medium transmisyjnego i wykrywanie kolizji sygnałów.

Zasada działania sieci Ethernet

Kolizja sygnałów (*ang. signal collision*)

Destruktywna interferencja fal EM przenoszących sygnały.

CSMA/CD (*ang. Carrier Sense, Multiple Access with Collision Detection*)

Technika pozwalająca na wykorzystywanie wspólnego medium transmisyjnego i wykrywanie kolizji sygnałów.

Wysyłanie ramki

1. Czekamy, aż medium transmisyjne będzie wolne (nikt nie nadaje).
2. Jeżeli medium transmisyjne jest wolne, 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.

Specyfikacja sieci Ethernet (IEEE 802.3)

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

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

Specyfikacja sieci Ethernet (IEEE 802.3)

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

- MAC (*ang. Media Access Control*) – dostęp do medium transmisyjnego.
- 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 (IEEE 802.3)

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

- MAC (*ang. Media Access Control*) – dostęp do medium transmisyjnego.
- 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 medium transmisyjnego 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. Ładunek danych – 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 zastosowania koncentratorów (*ang. hub*) lub przełączników (*ang. switch*).

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 zastosowania koncentratorów (*ang. hub*) lub przełączników (*ang. switch*).

Sieci światłowodowe wymagają stosowania przełączników.

Koncentratory i przełączniki

Dla skrętki i światłowodów stacje nie są bezpośrednio łączone kablami.

Koncentratory i przełączniki

Dla skrętki i światłowodów stacje nie są bezpośrednio łączone kablami.

Koncentrator

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

Koncentratory i przełączniki

Dla skrętki i światłowodów stacje nie są bezpośrednio łączone kablami.

Koncentrator

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

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 w domenie rozgłaszania.

Ethernet w sieciach lokalnych

Obecnie praktycznie wszystkie sieci lokalne poza sieciami bezprzewodowymi są sieciami typu Ethernet (przynajmniej są zgodne ze standardem w warstwie MAC).

Ethernet w sieciach lokalnych

Obecnie praktycznie wszystkie sieci lokalne poza sieciami bezprzewodowymi są sieciami typu Ethernet (przynajmniej są zgodne ze standardem w warstwie MAC).

Standard Ethernet wyparł z rynku wszystkie konkurencyjne standardy głównie ze względu na stosunkowo niską cenę urządzeń i kabli.

Ethernet w sieciach lokalnych

Obecnie praktycznie wszystkie sieci lokalne poza sieciami bezprzewodowymi są sieciami typu Ethernet (przynajmniej są zgodne ze standardem w warstwie MAC).

Standard Ethernet wyparł z rynku wszystkie konkurencyjne standardy głównie ze względu na stosunkowo niską cenę urządzeń i kabli.

Wadą sieci Ethernet jest mała przewidywalność związana z techniką CSMA/CD, którą jednak można poprawić poprzez stosowanie przełączników na dużą skalę (w szczególności przełączników zarządzalnych z możliwością kontrolowania każdego portu z osobna).

Ethernet w sieciach lokalnych

Obecnie praktycznie wszystkie sieci lokalne poza sieciami bezprzewodowymi są sieciami typu Ethernet (przynajmniej są zgodne ze standardem w warstwie MAC).

Standard Ethernet wyparł z rynku wszystkie konkurencyjne standardy głównie ze względu na stosunkowo niską cenę urządzeń i kabli.

Wadą sieci Ethernet jest mała przewidywalność związana z techniką CSMA/CD, którą jednak można poprawić poprzez stosowanie przełączników na dużą skalę (w szczególności przełączników zarządzalnych z możliwością kontrolowania każdego portu z osobna).

Obecnie w większości sieci Ethernet nie stosuje się koncentratorów, tylko przełączniki.

Standard IEEE 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 IEEE 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 IEEE 802.11, 1997

Pierwsza specyfikacja bezprzewodowej sieci LAN:

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

PHY dla sieci bezprzewodowych

IEEE 802.11a, 1999 – pasmo 5 GHz, maks. BR 54 Mb/s

IEEE 802.11b, 1999 – pasmo 2,4 GHz, maks. BR 11 Mb/s, 14 kanałów o szerokości 22 MHz każdy

IEEE 802.11g, 2003 – pasmo 2,4 GHz, maks. BR 54 Mb/s, 14 kanałów o szerokości 22 MHz każdy

IEEE 802.11n, 2009 – pasmo 2,4 GHz albo 5 GHz, maks. BR 600 Mb/s, kanały 40 MHz, wiele anten

PHY dla sieci bezprzewodowych

IEEE 802.11a, 1999 – pasmo 5 GHz, maks. BR 54 Mb/s

IEEE 802.11b, 1999 – pasmo 2,4 GHz, maks. BR 11 Mb/s, 14 kanałów o szerokości 22 MHz każdy

IEEE 802.11g, 2003 – pasmo 2,4 GHz, maks. BR 54 Mb/s, 14 kanałów o szerokości 22 MHz każdy

IEEE 802.11n, 2009 – pasmo 2,4 GHz albo 5 GHz, maks. BR 600 Mb/s, kanały 40 MHz, wiele anten

Wi-Fi

Znak firmowy i symbol marketingowy odpowiadający rodzinie standardów IEEE 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*), SS (*ang. 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*), SS (*ang. service set*)

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

ESS (*ang. Extended Service Set*)

Struktura złożona z wielu komórek (SS), odpowiadająca domenie rozgłaszania w sieci Ethernet. Połączenia między AP mogą być bezprzewodowe lub zbudowane z segmentów sieci Ethernet.

Ramki RTS i CTS

Problem „widzialności”

Jeżeli między węzłami sieci bezprzewodowej znajduje się bariera fizyczna tłumiąca fale EM, to nie można przesyłać ramek bezpośrednio między nimi. Dlatego pojęcie „wolnego medium transmisyjnego” nie jest dobrze zdefiniowane.

Ramki RTS i CTS

Problem „widzialności”

Jeżeli między węzłami sieci bezprzewodowej znajduje się bariera fizyczna tłumiąca fale EM, to nie można przesyłać ramek bezpośrednio między nimi. Dlatego pojęcie „wolnego medium transmisyjnego” nie jest dobrze zdefiniowane.

RTS (*ang. Request To Send*)

Ramka kontrolna wysyłana przez stację, która zamierza nadawać. Powinna ona dotrzeć do punktu dostępowego (AP).

Ramki RTS i CTS

Problem „widzialności”

Jeżeli między węzłami sieci bezprzewodowej znajduje się bariera fizyczna tłumiąca fale EM, to nie można przesyłać ramek bezpośrednio między nimi. Dlatego pojęcie „wolnego medium transmisyjnego” nie jest dobrze zdefiniowane.

RTS (*ang. Request To Send*)

Ramka kontrolna wysyłana przez stację, która zamierza nadawać. Powinna ona dotrzeć do punktu dostępowego (AP).

CTS (*ang. Clear To Send*)

Ramka kontrolna wysyłana przez AP na potwierdzenie, że określona stacja może zacząć nadawanie ramki z danymi.

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 PSK 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 PSK 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 PSK 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 (czyli liczby bitów przesyłanych w jednostce czasu).

Kodowanie typu Manchester (*ang. Manchester encoding*)

Wariant PSK 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 (czyli liczby bitów przesyłanych w jednostce czasu).

Dopuszczalne dla BR co najwyżej rzędu $10^7 b/s$ 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 (*ang. 5-level Pulse Amplitude Modulation*)

Zastosowania technik wąskopasmowych

Sieć Ethernet

10Base-T – kodowanie typu Manchester

100Base-TX – MLT-3 z 4B/5B

1000Base-T – PAM-5 (*ang. 5-level Pulse Amplitude Modulation*)

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.
- (Można pokazać, że) Sygnały od różnych nadawców nie interferują destruktywnie.

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 znacznie 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*).

Rodzina protokołów TCP/IP

IP (*ang. Internet Protocol*)

Protokół seciowy opracowany na potrzeby sieci rozległych w latach 1970-1980. Określa identyfikację węzłów sieci oraz sposób przesyłania pakietów między dowolną parą węzłów sieci.

Rodzina protokołów TCP/IP

IP (*ang. Internet Protocol*)

Protokół seciowy opracowany na potrzeby sieci rozległych w latach 1970-1980. Określa identyfikację węzłów sieci oraz sposób przesyłania pakietów między dowolną parą węzłów sieci.

UDP (*ang. User Datagram Protocol*)

Określa zasady bezpołączeniowego przesyłania danych między procesami (programami) działającymi na węzłach sieci wykorzystującej IP.

TCP (*ang. Transmission Control Protocol*)

Określa zasady przesyłania strumieni (*ang. stream*) danych między procesami (programami) działającymi na węzłach sieci wykorzystującej IP. TCP jest protokołem połączeniowym (*ang. connection-oriented*).

Adresy IP

Adresy IP (*ang. IP address*)

Słowa 32-bitowe identyfikujące węzły sieci działającej zgodnie z protokołem IP.

Adresy IP

Adresy IP (*ang. IP address*)

Słowa 32-bitowe identyfikujące węzły sieci działającej zgodnie z protokołem IP.

Nie każde takie słowo może być adresem IP węzła sieci. W szczególności adresem IP węzła sieci nie może być słowo, w którym trzy najbardziej znaczące bity są jedynkami lub którego najbardziej znaczący bajt ma wartość 0.

Adresy IP

Adresy IP (*ang. IP address*)

Słowa 32-bitowe identyfikujące węzły sieci działającej zgodnie z protokołem IP.

Nie każde takie słowo może być adresem IP węzła sieci. W szczególności adresem IP węzła sieci 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 osobno, 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 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).

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.

Procesy i hosty

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

Procesy i hosty

IP reguluje dostarczanie danych między węzłami sieci, 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*).

Procesy i hosty

IP reguluje dostarczanie danych między węzłami sieci, 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 węźle za pośrednictwem usługi SSH (*ang. Secure SHell*).

Procesy i hosty

IP reguluje dostarczanie danych między węzłami sieci, 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 węźle za pośrednictwem usługi SSH (*ang. Secure SHell*).

Host

Węzeł sieci IP udostępniający usługi lub korzystający z usług udostępnianych przez inne węzły.

Przesyłanie danych z procesu do procesu

Aby umożliwić komunikację między procesami, które mogą reprezentować różnych użytkowników korzystających z różnych węzłów sieci, 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 węzłów sieci, 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 węzłów sieci, 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. Zasoby te mają unikatowe 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. Zasoby te mają unikatowe 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. Zasoby te mają unikatowe 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 węzła sieci.

Gniazda (*ang. socket*)

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

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 węzła sieci.

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 węzłów sieci.

Gniazda (*ang. socket*)

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

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 węzłów sieci.

W celu dokonania zapisu do gniazda UDP proces musi podać adres IP węzła sieci i numer portu UDP odpowiadający procesowi (na tym węźle), 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 węzeł, którego adres IP pokrywa się z adresem IP miejsca przeznaczenia datagramu.
2. Na tym węźle istnieje 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 jest mowa.

Odbieranie danych wysłanych z użyciem UDP

Dane wysłane z użyciem UDP mogą być odebrane, gdy:

1. W sieci jest węzeł, którego adres IP pokrywa się z adresem IP miejsca przeznaczenia datagramu.
2. Na tym węźle istnieje 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 jest 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

Operacja, 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

Operacja, 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 IP o określonym nagłówku i ładunku danych z hosta-nadawcy do hosta-odbiorcy.

Nawiązywanie połączenia TCP

Operacja, 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 IP o określonym nagłówku i ładunku danych 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.

Porty TCP – numery portów TCP źródła i miejsca przeznaczenia segmentu.

Inicjowanie komunikacji z użyciem TCP

W celu rozpoczęcia komunikacji z użyciem TCP jądro systemu operacyjnego hosta-nadawcy wysyła do hosta-odbiorcy segment TCP bez danych użytecznych 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 użytecznych 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 użytecznych, 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 użytecznych, 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.

Dobrze znane usługi

Proces rozpoczynający wymianę danych musi znać z wyprzedzeniem numer portu procesu, z którym ma wymieniać dane.

Dobrze znane usługi

Proces rozpoczynający wymianę danych musi znać z wyprzedzeniem numer portu procesu, z którym ma wymieniać dane.

W związku z tym powstała konwencja polegająca na przydzielaniu określonych numerów portów (do nasłuchiwania) dla poszczególnych rodzajów usług sieciowych.

Dobrze znane usługi

Proces rozpoczynający wymianę danych musi znać z wyprzedzeniem numer portu procesu, z którym ma wymieniać dane.

W związku z tym powstała konwencja polegająca na przydzielaniu określonych numerów portów (do nasłuchiwania) dla poszczególnych rodzajów usług sieciowych.

Na przykład serwery WWW zwykle używają numerów portów 80 TCP i 443 TCP (do przesyłania danych z szyfrowaniem), serwery poczty elektronicznej używają portów 25 i 587 TCP itd.

Dobrze znane usługi

Proces rozpoczynający wymianę danych musi znać z wyprzedzeniem numer portu procesu, z którym ma wymieniać dane.

W związku z tym powstała konwencja polegająca na przydzielaniu określonych numerów portów (do nasłuchiwania) dla poszczególnych rodzajów usług sieciowych.

Na przykład serwery WWW zwykle używają numerów portów 80 TCP i 443 TCP (do przesyłania danych z szyfrowaniem), serwery poczty elektronicznej używają portów 25 i 587 TCP itd.

Dobrze znana usługa (*ang. well known service*)

Usługa, która ma przydzielony standardowy numer portu (TCP do nasłuchiwania lub UDP).

Przydzielanie portów na żądanie

Proces rozpoczynający komunikację zwykle nie potrzebuje mieć przydzielonego określonego numeru portu TCP lub UDP (druga strona i tak odczyta ten numer z nagłówka pierwszego przesłanego segmentu TCP lub datagramu UDP).

Przydzielanie portów na żądanie

Proces rozpoczynający komunikację zwykle nie potrzebuje mieć przydzielonego określonego numeru portu TCP lub UDP (druga strona i tak odczyta ten numer z nagłówka pierwszego przesłanego segmentu TCP lub datagramu UDP).

Dlatego procesy rozpoczynające komunikację bardzo często pozwalają na to, aby jądro systemu operacyjnego przydzielało im numery portów do tego celu (zmniejsza to prawdopodobieństwo konfliktów).

Przydzielanie portów na żądanie

Proces rozpoczynający komunikację zwykle nie potrzebuje mieć przydzielonego określonego numeru portu TCP lub UDP (druga strona i tak odczyta ten numer z nagłówka pierwszego przesłanego segmentu TCP lub datagramu UDP).

Dlatego procesy rozpoczynające komunikację bardzo często pozwalają na to, aby jądro systemu operacyjnego przydzielało im numery portów do tego celu (zmniejsza to prawdopodobieństwo konfliktów).

Porty te są zwykle wybierane losowo spośród wszystkich portów dostępnych w danej chwili czasu.

Porty i adres IP

Port TCP (do nasłuchiwania) lub UDP może być skojarzony z konkretnym adresem IP (np. `127.0.0.1`).

Porty i adres IP

Port TCP (do nasłuchiwania) lub UDP może być skojarzony z konkretnym adresem IP (np. `127.0.0.1`).

Wtedy wymiana danych z procesem, któremu został przydzielony ten port, może być rozpoczęta tylko poprzez wysłanie pakietu na adres IP, z którym jest on skojarzony.

Porty i adres IP

Port TCP (do nasłuchiwania) lub UDP może być skojarzony z konkretnym adresem IP (np. `127.0.0.1`).

Wtedy wymiana danych z procesem, któremu został przydzielony ten port, może być rozpoczęta tylko poprzez wysłanie pakietu na adres IP, z którym jest on skojarzony.

W przypadku braku takich ograniczeń porty TCP i UDP są kojarzone z adresem `0.0.0.0`, co oznacza, że do nawiązania wymiany danych z używającymi ich procesami wystarczy wysłanie pakietu na dowolny adres przypisany węzłowi sieci, na którym zostały uruchomione te procesy.

Protokoły usług sieciowych

Procesy wymieniające dane użyteczne z wykorzystaniem TCP lub UDP (dalej będzie mowa o TCP) muszą poprawnie interpretować te dane.

Protokoły usług sieciowych

Procesy wymieniające dane użyteczne z wykorzystaniem TCP lub UDP (dalej będzie mowa o TCP) muszą poprawnie interpretować te dane.

Zatem proces wysyłający dane musi je formatować tak, aby proces odbierający te dane mógł je wykorzystać do uzyskania określonych wyników.

Protokoły usług sieciowych

Procesy wymieniające dane użyteczne z wykorzystaniem TCP lub UDP (dalej będzie mowa o TCP) muszą poprawnie interpretować te dane.

Zatem proces wysyłający dane musi je formatować tak, aby proces odbierający te dane mógł je wykorzystać do uzyskania określonych wyników.

Na przykład serwer WWW musi „wiedzieć" jaka część przesłanych do niego danych reprezentuje URL zasobu, do którego należy odwołać się.

Protokoły usług sieciowych

Procesy wymieniające dane użyteczne z wykorzystaniem TCP lub UDP (dalej będzie mowa o TCP) muszą poprawnie interpretować te dane.

Zatem proces wysyłający dane musi je formatować tak, aby proces odbierający te dane mógł je wykorzystać do uzyskania określonych wyników.

Na przykład serwer WWW musi „wiedzieć" jaka część przesłanych do niego danych reprezentuje URL zasobu, do którego należy odwołać się.

Jeżeli programy uczestniczące w przesyłaniu danych mają różnych autorów, to najczęściej nie kontaktują się oni bezpośrednio ze sobą nawzajem i potrzeby jest protokół (tzn. standard) określający jak należy interpretować przesyłane dane.

Protokoły usług sieciowych (c. d.)

Usługi sieciowe zwykle są definiowane wraz z protokołami określającymi interpretację przesyłanych danych (np. w jaki sposób ma być przygotowane zgłoszenie od klienta i jaki ma być format odpowiedzi serwera).

Protokoły usług sieciowych (c. d.)

Usługi sieciowe zwykle są definiowane wraz z protokołami określającymi interpretację przesyłanych danych (np. w jaki sposób ma być przygotowane zgłoszenie od klienta i jaki ma być format odpowiedzi serwera).

Wiele z tych protokołów zakłada, że dane będą przesyłane z wykorzystaniem TCP, ale (na ogół) nie odwołują się one do zawartości nagłówków TCP oraz IP. Dotyczą one tego, w jaki sposób mają być formatowane i interpretowane dane, które z punktu widzenia TCP/IP stanowią dane użyteczne.

Protokoły usług sieciowych (c. d.)

Usługi sieciowe zwykle są definiowane wraz z protokołami określającymi interpretację przesyłanych danych (np. w jaki sposób ma być przygotowane zgłoszenie od klienta i jaki ma być format odpowiedzi serwera).

Wiele z tych protokołów zakłada, że dane będą przesyłane z wykorzystaniem TCP, ale (na ogół) nie odwołują się one do zawartości nagłówków TCP oraz IP. Dotyczą one tego, w jaki sposób mają być formatowane i interpretowane dane, które z punktu widzenia TCP/IP stanowią dane użyteczne.

Tego rodzaju protokoły zostały stworzone dla praktycznie wszystkich popularnych usług sieciowych (np. WWW, poczta elektroniczna, FTP, SSH, komunikatory) i definicje większości z nich są publikowane.

Usługi sieciowe i warstwa aplikacji

Ponieważ protokoły usług sieciowych odnoszą się do danych, które z punktu widzenia TCP/IP są danymi użytecznymi, można powiedzieć, że stanowią one „wyższy poziom ogólności" w stosunku do TCP (są „bardziej abstrakcyjne").

Usługi sieciowe i warstwa aplikacji

Ponieważ protokoły usług sieciowych odnoszą się do danych, które z punktu widzenia TCP/IP są danymi użytecznymi, można powiedzieć, że stanowią one „wyższy poziom ogólności" w stosunku do TCP (są „bardziej abstrakcyjne").

Ponieważ określają one zasady komunikacji między aplikacjami, odpowiadając im „poziom ogólności" jest nazywany warstwą aplikacji (*ang. application layer*).

Usługi sieciowe i warstwa aplikacji

Ponieważ protokoły usług sieciowych odnoszą się do danych, które z punktu widzenia TCP/IP są danymi użytecznymi, można powiedzieć, że stanowią one „wyższy poziom ogólności" w stosunku do TCP (są „bardziej abstrakcyjne").

Ponieważ określają one zasady komunikacji między aplikacjami, odpowiadając im „poziom ogólności" jest nazywany warstwą aplikacji (*ang. application layer*).

Na tym poziomie nie jest ważne to, w jaki sposób dane są dostarczane do procesu, z którym wymieniamy dane, ani to, gdzie ten proces fizycznie znajduje się. Ważna jest tylko treść (*ang. contents*) danych przesyłanych w jednym i w drugim kierunku.

Podział standardów komunikacyjnych na warstwy

Wszystkie standardy związane z przesyłaniem danych w sieciach TCP/IP można pogrupować w analogii do protokołów z warstwy aplikacji.

Podział standardów komunikacyjnych na warstwy

Wszystkie standardy związane z przesyłaniem danych w sieciach TCP/IP można pogrupować w analogii do protokołów z warstwy aplikacji.

Wtedy okazuje się, że można wyróżnić cztery grupy standardów wykorzystywanych w sieciach TCP/IP, zwanych warstwami (*ang. layer*):

1. Warstwa fizyczna i łącza danych (*ang. physical and data link layer*), np. Ethernet, ADSL itp.
2. Warstwa sieciowa (*ang. network layer*) – IP, ICMP.
3. Warstwa transportowa (*ang. transport layer*) – TCP, UDP.
4. Warstwa aplikacji.

Podział standardów komunikacyjnych na warstwy

Wszystkie standardy związane z przesyłaniem danych w sieciach TCP/IP można pogrupować w analogii do protokołów z warstwy aplikacji.

Wtedy okazuje się, że można wyróżnić cztery grupy standardów wykorzystywanych w sieciach TCP/IP, zwanych warstwami (*ang. layer*):

1. Warstwa fizyczna i łącza danych (*ang. physical and data link layer*), np. Ethernet, ADSL itp.
2. Warstwa sieciowa (*ang. network layer*) – IP, ICMP.
3. Warstwa transportowa (*ang. transport layer*) – TCP, UDP.
4. Warstwa aplikacji.

W związku z podziałem na warstwy rodzina protokołów TCP/IP bywa nazywana stosem protokołów (*ang. protocol stack*) TCP/IP.

Inne stosy protokołów

TCP/IP nie jest jedynym stosem protokołów o takim charakterze, ale obecnie jest najczęściej wykorzystywany w praktyce.

Inne stosy protokołów

TCP/IP nie jest jedynym stosem protokołów o takim charakterze, ale obecnie jest najczęściej wykorzystywany w praktyce.

Drugim często wykorzystywanym stosem protokołów jest stos TCP/IPv6, w której „tradycyjny" protokół IPv4 (oraz protokoły zależne od niego) jest zastąpiony protokołem IPv6 (większa przestrzeń adresowa, „wbudowane" szyfrowanie itp.).

Inne stosy protokołów

TCP/IP nie jest jedynym stosem protokołów o takim charakterze, ale obecnie jest najczęściej wykorzystywany w praktyce.

Drugim często wykorzystywanym stosem protokołów jest stos TCP/IPv6, w której „tradycyjny" protokół IPv4 (oraz protokoły zależne od niego) jest zastąpiony protokołem IPv6 (większa przestrzeń adresowa, „wbudowane" szyfrowanie itp.).

W przeszłości były wykorzystywane inne stosy protokołów, jak IPX/SPX firmy *Novell* lub DECNet firmy *Digital Equipment Corporation*, ale z czasem zostały one zastąpione przez TCP/IP.

Model OSI

W związku z istnieniem różnych stosów protokołów sieciowych został stworzony model teoretyczny opisujący komunikację sieciową z podziałem na maksymalną sensowną liczbę warstw funkcjonalnych.

Model OSI

W związku z istnieniem różnych stosów protokołów sieciowych został stworzony model teoretyczny opisujący komunikację sieciową z podziałem na maksymalną sensowną liczbę warstw funkcjonalnych.

Został on nazwany modelem OSI (*ang. Open Systems Interconnect*) i miał stanowić model odniesienia dla teoretycznego opisu komunikacji sieciowej.

Model OSI

W związku z istnieniem różnych stosów protokołów sieciowych został stworzony model teoretyczny opisujący komunikację sieciową z podziałem na maksymalną sensowną liczbę warstw funkcjonalnych.

Został on nazwany modelem OSI (*ang. Open Systems Interconnect*) i miał stanowić model odniesienia dla teoretycznego opisu komunikacji sieciowej.

Model OSI dzieli komunikację sieciową na 7 warstw funkcjonalnych, które nie stanowią bezpośrednich odpowiedników dla warstw sieci TCP/IP.

Warstwy sieci według modelu OSI

1. Fizyczna – fizyczne i elektryczne specyfikacje urządzeń.
2. Łącza danych – transfer danych między urządzeniami w sieci.
3. Sieciowa – transfer sekwencji danych o zmiennej długości między źródłem w jednej sieci i miejscem przeznaczenia w drugiej.
4. Transportowa – „przezroczysty" transfer danych między użytkownikami sieci.
5. Sesji (*ang. session*) – zarządzanie połączeniami między procesami w sieci.
6. Prezentacji (*ang. presentation*) – tłumaczenie danych między formatami sieci i aplikacji (np. szyfrowanie).
7. Aplikacji.

Nazwy węzłów sieci i adresy IP

Dla ludzi znacznie wygodniejsze jest posługiwanie się nazwami węzłów sieci, niż ich adresami IP (zwłaszcza w przypadku IPv6).

Nazwy węzłów sieci i adresy IP

Dla ludzi znacznie wygodniejsze jest posługiwanie się nazwami węzłów sieci, niż ich adresami IP (zwłaszcza w przypadku IPv6).

Powoduje to jednak konieczność tłumaczenia nazw, którymi posługują się ludzie, na adresy IP potrzebne do przesyłania danych.

Nazwy węzłów sieci i adresy IP

Dla ludzi znacznie wygodniejsze jest posługiwanie się nazwami węzłów sieci, niż ich adresami IP (zwłaszcza w przypadku IPv6).

Powoduje to jednak konieczność tłumaczenia nazw, którymi posługują się ludzie, na adresy IP potrzebne do przesyłania danych.

Do realizacji tego zadania potrzebna jest baza danych, w której przechowywane są informacje dotyczące przyporządkowania między nazwami i adresami IP, jednak nie może to być baza danych zarządzana centralnie dla całej sieci (takie rozwiązanie byłoby bardzo nieefektywne).

Nazwy węzłów sieci i adresy IP

Dla ludzi znacznie wygodniejsze jest posługiwanie się nazwami węzłów sieci, niż ich adresami IP (zwłaszcza w przypadku IPv6).

Powoduje to jednak konieczność tłumaczenia nazw, którymi posługują się ludzie, na adresy IP potrzebne do przesyłania danych.

Do realizacji tego zadania potrzebna jest baza danych, w której przechowywane są informacje dotyczące przyporządkowania między nazwami i adresami IP, jednak nie może to być baza danych zarządzana centralnie dla całej sieci (takie rozwiązanie byłoby bardzo nieefektywne).

Okazuje się, ze implementacja takiej bazy danych jest łatwiejsza, gdy zbiór wszystkich możliwych nazw węzłów sieci ma pewną strukturę.

Domeny nazewnicze

Podzielmy zbiór wszystkich dopuszczalnych nazw węzłów sieci, zwany domeną główną (*ang. root domain*), na takie podzbiory, że wszystkie nazwy w danym podzbiorze mają identyczne zakończenie, zaczynające się od znaku `.`, np. `.com`, `.org`, `.net` itd. Każdy taki podzbiór nazwiemy domeną najwyższego poziomu lub TLD (*ang. Top-Level Domain*).

Domeny nazewnicze

Podzielmy zbiór wszystkich dopuszczalnych nazw węzłów sieci, zwany domeną główną (*ang. root domain*), na takie podzbiory, że wszystkie nazwy w danym podzbiorze mają identyczne zakończenie, zaczynające się od znaku `.`, np. `.com`, `.org`, `.net` itd. Każdy taki podzbiór nazwiemy domeną najwyższego poziomu lub TLD (*ang. Top-Level Domain*).

Z kolei każdą TLD dzielimy na takie podzbiory, że wszystkie nazwy w danym podzbiorze mają identyczne zakończenie, zaczynające się od znaku `.` i kończące się ciągiem znaków przypisanym danej TLD, np. `.com.pl`, `.edu.pl`, `.art.pl` itd. Podzbiory te nazywa się po prostu domenami (*ang. domain*).

Domeny nazewnicze

Podzielmy zbiór wszystkich dopuszczalnych nazw węzłów sieci, zwany domeną główną (*ang. root domain*), na takie podzbiory, że wszystkie nazwy w danym podzbiorze mają identyczne zakończenie, zaczynające się od znaku `.`, np. `.com`, `.org`, `.net` itd. Każdy taki podzbiór nazwiemy domeną najwyższego poziomu lub TLD (*ang. Top-Level Domain*).

Z kolei każdą TLD dzielimy na takie podzbiory, że wszystkie nazwy w danym podzbiorze mają identyczne zakończenie, zaczynające się od znaku `.` i kończące się ciągiem znaków przypisanym danej TLD, np. `.com.pl`, `.edu.pl`, `.art.pl` itd. Podzbiory te nazywa się po prostu domenami (*ang. domain*).

Domeny zawarte w określonej TLD nazywa się jej poddomenami (*ang. subdomain*).

Serwery nazw

W poddomenach TLD także można utworzyć poddomeny (w analogiczny sposób) itd.

Serwery nazw

W poddomenach TLD także można utworzyć poddomeny (w analogiczny sposób) itd.

Nadając nazwę węzłowi sieci określa się domenę, do której ma ona należeć (wyznacza to zakończenie nazwy węzła sieci) oraz część lokalną nazwy, która musi być unikatowa w obrębie danej domeny (np. nazwa `www.fuw.edu.pl` należy do domeny `fuw.edu.pl`, a jej częścią lokalną jest `www`).

Serwery nazw

W poddomenach TLD także można utworzyć poddomeny (w analogiczny sposób) itd.

Nadając nazwę węzłowi sieci określa się domenę, do której ma ona należeć (wyznacza to zakończenie nazwy węzła sieci) oraz część lokalną nazwy, która musi być unikatowa w obrębie danej domeny (np. nazwa `www.fuw.edu.pl` należy do domeny `fuw.edu.pl`, a jej częścią lokalną jest `www`).

Dla każdej domeny trzeba skonfigurować co najmniej jeden komputer przechowujący informacje dotyczące przyporządkowania nazw węzłów sieci w tej domenie do adresów IP. Jest on nazywany serwerem nazw (*ang. name server*) dla danej domeny.

Drugorzędne serwery nazw

Liczba serwerów nazw dla jednej domeny może być dowolna, ale jeden z nich musi być serwerem podstawowym (*ang. primary*), przechowującym „oryginał” informacji o przyporządkowaniu nazw do adresów IP w tej domenie.

Drugorzędne serwery nazw

Liczba serwerów nazw dla jednej domeny może być dowolna, ale jeden z nich musi być serwerem podstawowym (*ang. primary*), przechowującym „oryginał" informacji o przyporządkowaniu nazw do adresów IP w tej domenie.

Pozostałe serwery nazw dla danej domeny są drugorzędne (*ang. secondary*) i przechowują kopie danych z podstawowego serwera nazw.

Drugorzędne serwery nazw

Liczba serwerów nazw dla jednej domeny może być dowolna, ale jeden z nich musi być serwerem podstawowym (*ang. primary*), przechowującym „oryginał” informacji o przyporządkowaniu nazw do adresów IP w tej domenie.

Pozostałe serwery nazw dla danej domeny są drugorzędne (*ang. secondary*) i przechowują kopie danych z podstawowego serwera nazw.

Drugorzędne serwery nazw okresowo pobierają dane z serwera podstawowego. Może to również następować na żądanie serwera podstawowego (np. po wprowadzeniu zmian).

Lokalne serwery nazw

Dla każdego węzła sieci TCP/IP powinien być określony lokalny serwer nazw, do którego ten węzeł będzie w pierwszej kolejności wysyłał pytania o adres IP odpowiadający danej nazwie węzła sieci.

Lokalne serwery nazw

Dla każdego węzła sieci TCP/IP powinien być określony lokalny serwer nazw, do którego ten węzeł będzie w pierwszej kolejności wysyłał pytania o adres IP odpowiadający danej nazwie węzła sieci.

Lokalny swerwer nazw przechowuje pewną liczbę ostatnio udzielonych odpowiedzi, co pozwala znacząco zredukować obciążenie sieci związane z tzw. rozwiązywaniem nazw (*ang. name resolving*).

Lokalne serwery nazw

Dla każdego węzła sieci TCP/IP powinien być określony lokalny serwer nazw, do którego ten węzeł będzie w pierwszej kolejności wysyłał pytania o adres IP odpowiadający danej nazwie węzła sieci.

Lokalny swerwer nazw przechowuje pewną liczbę ostatnio udzielonych odpowiedzi, co pozwala znacząco zredukować obciążenie sieci związane z tzw. rozwiązywaniem nazw (*ang. name resolving*).

Jeżeli odpowiedź na pytanie zadane przez węzeł sieci, dla którego dany serwer jest lokalnym serwerem nazw, jest już przez niego przechowywana, jest ona natychmiast udzielana węzłowi sieci zadającemu pytanie.

Lokalne serwery nazw

Dla każdego węzła sieci TCP/IP powinien być określony lokalny serwer nazw, do którego ten węzeł będzie w pierwszej kolejności wysyłał pytania o adres IP odpowiadający danej nazwie węzła sieci.

Lokalny swerwer nazw przechowuje pewną liczbę ostatnio udzielonych odpowiedzi, co pozwala znacząco zredukować obciążenie sieci związane z tzw. rozwiązywaniem nazw (*ang. name resolving*).

Jeżeli odpowiedź na pytanie zadane przez węzeł sieci, dla którego dany serwer jest lokalnym serwerem nazw, jest już przez niego przechowywana, jest ona natychmiast udzielana węzłowi sieci zadającemu pytanie.

W przeciwnym wypadku lokalny serwer nazw przeprowadza rekurencyjne (*ang. recursive*) poszukiwanie adresu IP odpowiadającego danej nazwie.

Poszukiwanie adresu IP odpowiadającego danej nazwie

1. Pytanie jest zadawane jednemu z serwerów nazw dla domeny głównej (ich adresy IP są znane wszystkim lokalnym serwerom nazw).
2. Serwer nazw dla domeny głównej określa TLD, do której należy nazwa i przekazuje w odpowiedzi listę adresów IP serwerów nazw dla tej TLD.
3. Pytanie jest zadawane jednemu z serwerów nazw w danej TLD.
4. Serwer nazw dla TLD określa poddomenę, do której należy nazwa i przekazuje w odpowiedzi listę adresów IP serwerów nazw dla tej poddomeny.
5. ...
6. Pytanie jest zadawane jednemu z serwerów nazw w najmniejszej domenie, do której należy nazwa.
7. Serwer nazw dla najmniejszej domeny, do której należy nazwa, przechowuje informację o odpowiadającym tej nazwie adresie IP i przesyła go w odpowiedzi.

Domenowy system nazw

DNS (*ang. Domain Name Service*)

Usługa realizowana przez wszystkie współpracujące ze sobą serwery nazw w sieci TCP/IP, pozwalająca na (stosunkowo szybkie) znalezienie adresu IP skojarzonego z podaną nazwą węzła sieci.

Domenowy system nazw

DNS (*ang. Domain Name Service*)

Usługa realizowana przez wszystkie współpracujące ze sobą serwery nazw w sieci TCP/IP, pozwalająca na (stosunkowo szybkie) znalezienie adresu IP skojarzonego z podaną nazwą węzła sieci.

Do sprawnego funkcjonowania DNS potrzebny jest (odpowiedni) podział domeny głównej na TLD i zbiór serwerów nazw dla domeny głównej, które są dostępne z całej sieci. System złożony z serwerów nazw wspólnie realizujących usługę DNS również bywa nazywany DNS (*ang. Domain Name System*).

Domenowy system nazw

DNS (*ang. Domain Name Service*)

Usługa realizowana przez wszystkie współpracujące ze sobą serwery nazw w sieci TCP/IP, pozwalająca na (stosunkowo szybkie) znalezienie adresu IP skojarzonego z podaną nazwą węzła sieci.

Do sprawnego funkcjonowania DNS potrzebny jest (odpowiedni) podział domeny głównej na TLD i zbiór serwerów nazw dla domeny głównej, które są dostępne z całej sieci. System złożony z serwerów nazw wspólnie realizujących usługę DNS również bywa nazywany DNS (*ang. Domain Name System*).

Informacje przechowywane przez wszystkie serwery nazw w DNS stanowią rozproszoną bazę danych (*ang. distributed database*), którą można przeszukiwać za pośrednictwem serwerów nazw.