

BAB II

TINJAUAN PUSTAKA

II.1 Jaringan Komputer

II.1.1 Sejarah Jaringan Komputer

Berkembangnya kebutuhan pengolahan data dan informasi, didalam perusahaan dibutuhkan beberapa komputer yang digunakan orang banyak yang bekerja dalam sebuah tim. Untuk saling bertukar data dan informasi, maka komputer-komputer yang digunakan akan terhubung satu dengan yang lainnya. Kumpulan komputer yang terhubung disebut jaringan komputer.

Keuntungan yang didapat dengan menggunakan jaringan komputer dapat mengakses data komputer lain yang digunakan, data yang digunakan dapat disimpan atau dicopy kebeberapa komputer sehingga bila salah satu komputer rusak, maka salinan dikomputer yang lain masih bisa digunakan.

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung. Informasi dan data bergerak melalui kabel-kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data, mencetak pada printer yang sama dan bersama sama menggunakan *hardware/software* yang terhubung dengan jaringan. Tiap komputer, *printer* atau *periferal* yang terhubung dengan jaringan disebut *node*. Sebuah jaringan komputer dapat memiliki dua, puluhan, ribuan atau bahkan jutaan node.

Sebuah jaringan biasanya terdiri dari 2 atau lebih komputer yang saling berhubungan diantara satu dengan yang lain, dan saling berbagi sumber daya misalnya CDROM, *Printer*, pertukaran *file*, atau memungkinkan untuk saling berkomunikasi secara elektronik. (Agung, leo. 2010 : 2-3).

II.1.2 Definisi Jaringan Komputer

Jaringan komputer adalah sebuah kumpulan komputer, *printer* dan peralatan lainnya yang terhubung. Informasi dan data bergerak melalui kabel-kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data, mencetak pada printer yang sama dan bersama sama menggunakan *hardware* dan *software* yang terhubung dengan jaringan (Kadir, Abdul.2009;11)

Menurut definisi (Sofana, 2011, hal. 4-5), yang dimaksud dengan jaringan komputer (komputer *networks*) adalah suatu himpunan *interkoneksi* sejumlah komputer *autonomous*. Kata “*autonomous*” mengandung pengertian bahwa komputer tersebut memiliki kendali atas dirinya sendiri. Memiliki manajemen sendiri (ada admin sendiri), memiliki topologi jaringan, *hardware* dan *software* sendiri. Sedangkan, dua buah komputer dikatakan “interkoneksi” apabila keduanya berbagi *resources* yang dimiliki, seperti saling bertukar data/informasi, berbagi *printer*, berbagi media penyimpanan dan sebagainya.

Jadi, jaringan komputer dapat dikatakan sebagai kumpulan beberapa komputer yang terhubung satu sama lain dan dapat saling berbagi *resources*.

Dalam jaringan ada tiga komponen utama yang harus dipahami. Komponen utama dalam jaringan komputer) yaitu :

1. *Host* atau *node*, yaitu sistem komputer yang berfungsi sebagai sumber atau penerima dari data yang dikirimkan. *Node* ini dapat berupa:
 - a. *Server* : suatu sistem komputer yang menyediakan jenis layanan tertentu untuk client dalam suatu jaringan komputer. *Server* dilengkapi dengan sistem operasi khusus untuk mengontrol akses dan sumber daya yang ada di dalamnya biasanya sistem operasi khusus tersebut disebut sistem operasi jaringan atau *network operating system*. (Agung, leo. 2010 : 4).
 - b. *Client* : komputer yang memiliki sebuah software aplikasi yang memungkinkan pengguna untuk mengakses servis atau layanan dari komputer server. Sebutan untuk mengatur komputer dalam jaringan (node/simpul) yang mengambil data/informasi dari komputer lain.
 - c. *Shared periperal* : peralatan-peralatan yang terhubung dan digunakan dalam jaringan (misalnya, *printer, scanner, harddisk, modem*, dan lain-lain). (Agung, leo. 2010 : 4-5).
2. *Link*, adalah media komunikasi yang menghubungkan antara node yang satu dengan *node* lainnya. Media ini dapat berupa saluran transmisi kabel dan tanpa kabel. (Sofana, Iwan. 2010 : 56)
3. *Software* (Perangkat Lunak), yaitu program yang mengatur dan mengelola jaringan secara keseluruhan. Termasuk di dalamnya sistem operasi jaringan yang berfungsi sebagai pengatur komunikasi data dan periferal dalam jaringan.



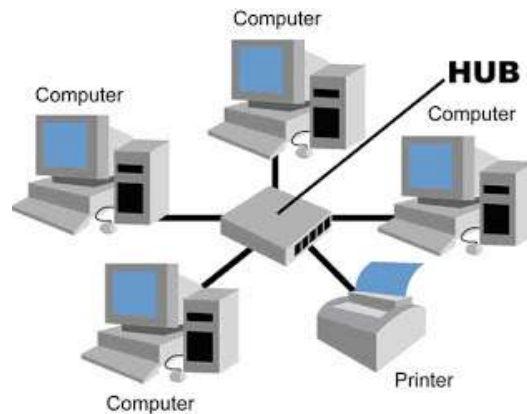
Gambar II.1 Komponen Utama Dalam Jaringan Komputer
(Sumber :Madcoms, 2009: 122).

II.1.3 Tipe-tipe Jaringan Komputer

Ada beberapa tipe jaringan komputer yang umumnya digunakan. Berikut ini beberapa klasifikasi tipe jaringan komputer yang ada,

1. Berdasarkan letak geografis terbagi atas,
 - a. *Local Area Network (LAN)*, merupakan jaringan yang menghubungkan sejumlah komputer yang ada dalam suatu lokasi dengan area yang terbatas seperti ruang atau gedung. LAN dapat menggunakan media komunikasi seperti kabel dan *wireless*.
 - b. *Wide Area Network (WAN)*, merupakan jaringan antara LAN satu dengan yang lain yang dipisahkan oleh lokasi yang cukup jauh. Hubungan WAN antara kantor pusat dengan kantor cabang yang ada didaerah-daerahnya.
 - c. *Metropolitan area network (MAN)* merupakan jaringan yang lebih besar dari jaringan LAN tetapi lebih kecil dari jaringan WAN sama-sama menghubungkan beberapa LAN yang membedakan hanya lingkup area yang berbeda. (Agung, leo. 2010 : 2-3).

2. Berdasarkan arsitektur jaringan terbagi atas,
- a. Jaringan *peer to peer* adalah jaringan komputer yang terdiri dari beberapa komputer (biasanya tidak lebih dari 10 komputer dengan 1-2 *printer*). Dalam sistem jaringan ini yang diutamakan adalah penggunaan program, data dan printer secara bersama-sama. Misalnya Pemakai komputer bernama Dona dapat memakai program yang dipasang di komputer Dino, dan mereka berdua dapat mencetak ke *printer* yang sama pada saat yang bersamaan.



Gambar II.2. jaringan *peer to peer*
(Sumber : Agung, leo. 2010 : 4)

- b. Jaringan berbasis *server* (*server-based network/server-client network*).
 - c. Jaringan *hibrid*.
3. Berdasarkan teknologi transmisi terbagi atas,
- a. Jaringan *switch*, merupakan perangkat yang bekerja hampir sama dengan hub. *Switch* mengenal MAC address yang digunakan untuk memilah data mana yang harus ditransmisikan. *Switch* menampung daftar MAC address yang menghubungkan dengan *port-port* yang akan digunakan untuk

menentukan kemana harus mengirim paket, sehingga akan mengurangi traffic pada jaringan.

- b. Jaringan *broadcast*, merupakan jaringan yang penyampaian informasi dari pengirim ke penerima dilakukan secara *broadcast* (disiarkan ke segala arah) baik melalui saluran kabel maupun saluran tanpa kabel.

(Agung, leo. 2010 : 9).

Beberapa komponen dasar yang biasanya membentuk suatu LAN adalah sebagai berikut,

1. *Workstation*

Workstation adalah komputer yang memanfaatkan jaringan untuk menghubungkan komputer tersebut dengan komputer lain atau komputer tersebut dengan *server*. Pemanfaatan jaringan tersebut dapat berupa *sharing data*, *sharing printer* dan sebagainya. Apabila terjadi kerusakan pada komputer *workstation* berarti komputer yang digunakan tidak dapat masuk dalam jaringan sehingga tidak dapat berkomunikasi dengan komputer *server* maupun komputer lain dalam jaringan tersebut. (Iwan Sofana. 2010 ; 60)

Workstation dapat berfungsi sebagai,

- a. *Server*

Sesuai dengan namanya, ini adalah perangkat keras yang berfungsi untuk melayani jaringan dan klien yang terhubung pada jaringan tersebut.

Server dapat berupa sistem komputer yang khusus dibuat untuk keperluan tertentu, seperti untuk penggunaan *printer* secara bersama (*print server*), untuk hubungan eksternal LAN ke jaringan lain (*communication server*),

dan *file server* yakni *disk* yang digunakan secara bersama oleh beberapa klien. *Server* ini tidak dapat digunakan sebagai klien, karena baik secara *hardware* maupun *software*, hanya berfungsi untuk mengelola jaringan.

Ada pula *server* yang berupa *workstation* dengan *disk drive* yang cukup besar kapasitasnya, sehingga *server* tersebut dapat juga digunakan sebagai *workstation* oleh *user*.

b. *Client* (klien)

Sebuah *workstation* umumnya berfungsi sebagai klien dari suatu *server*, karena memang *workstation* akan menggunakan fasilitas yang diberikan oleh suatu *server*. Jadi, *server* melayani, sedangkan klien dilayani (Iwan Sofana. 2010 ; 61)

2. *Link*

Link atau hubungan dalam Jaringan Lokal dikenal sebagai media transmisi berupa kabel maupun tanpa kabel, yang secara fisik menghubungkan *server* dan klien. Ada pula peralatan tambahan seperti *repeater*, *bridge*, *gateway* yang terlibat sebagai perpanjangan jarak capai hubungan jaringan.

3. *Transceiver*

Transceiver (*transmitter-receiver*) merupakan perangkat keras yang menghubungkan *workstation* atau sistem komputer dengan media transmisi .

4. Kartu Jaringan (*Network Interface Card / NIC*)

Kartu jaringan ini adalah kartu yang dipasang pada PC yang mengendalikan pertukaran data antar *workstation* yang ada dalam jaringan

lokal. Setiap *workstation* harus dilengkapi dengan NIC yang secara fisik terhubung langsung dengan *bus internal* dari PC.

5. Perangkat Lunak Jaringan

Perangkat lunak jaringan mencakup,

- a. Sistem operasi LAN.
- b. Perangkat lunak aplikasi.
- c. Perangkat lunak pemrograman.
- d. Program utiliti.

Perangkat lunak ini sangat penting dan mutlak untuk memungkinkan komunikasi antara sistem komputer yang satu dengan sistem komputer lainnya. Tanpa perangkat lunak ini, jaringan tidak akan berfungsi. Sistem komputer dengan LAN dapat menjalankan semua perangkat lunak aplikasi yang dapat berjalan pada *stand-alone* PC (Iwan Sofana. 2010 ; 64-66)

II.1.4 Media Transmisi

1. Media *Wires* (dengan kabel) yang menggunakan kabel tembaga berupa kabel koaksial dan kabel *twisted pair* atau kabel serat optik
 - a. Jaringan Berkabel (*Wire Network*)

Wire Network adalah jaringan komputer yang menggunakan kabel sebagai media pengantar.

Kabel yang digunakan pada jaringan komputer biasanya terbuat dari bahan tembaga. Ada juga jenis kabel lain yang dibuat menggunakan bahan *fiber optic* atau serat optik. Biasanya kabel berbahan tembaga banyak digunakan pada *Local Area Network* (LAN). Sedangkan untuk *Metropolitan Area Network* (MAN) atau

Wide Area Network (WAN) banyak menggunakan kabel tembaga dan *fiber optic* (FO).

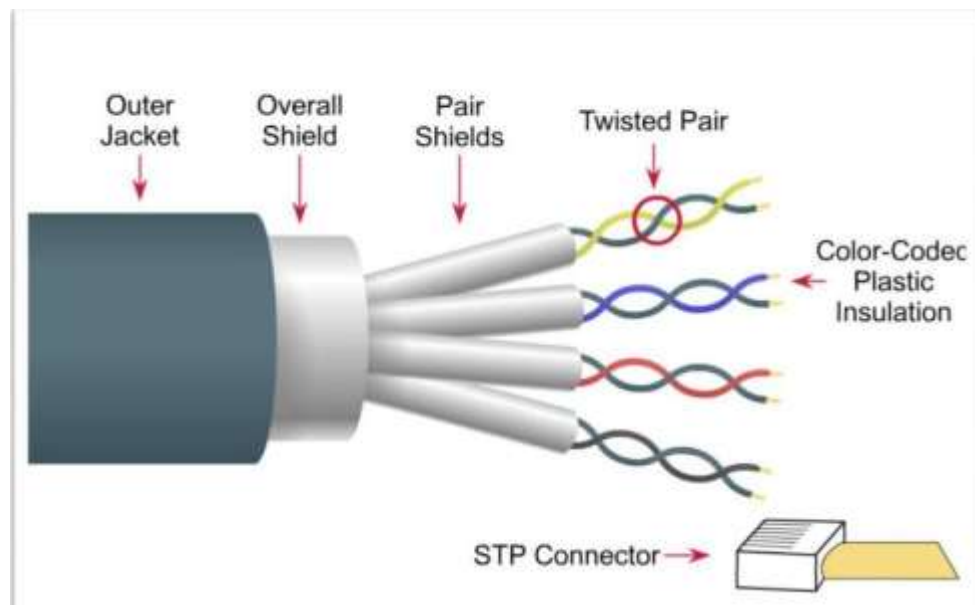
Pada *wirenetwork*, ada beberapa pilihan kabel yang dapat digunakan, yaitu kabel *coaxial*, kabel *twistedpair* (TP), dan kabel *fiberoptic*. (Sofana, 2011 : 31)

b. Twisted pair

Kabel *twistedpair* dibagi menjadi dua yaitu :

- *Unshielded twisted pair* (UTP)
- *Shielded twisted pair* (STP)

Kabel *Unshielded twisted pair* (UTP) lebih banyak digunakan untuk *network* topologi *Star* yang di install di dalam ruangan/gedung. Sedangkan *Shielded twisted pair* (STP) banyak digunakan untuk menghubungkan beberapa buah *network* topologi *star*. Misalkan menghubungkan 1 gedung dengan gedung yang lain (outdoor). *Shielded twisted pair* (STP) digunakan pada *network Token Ring*. Kabel *Shielded twisted pair* (STP) lebih tahan terhadap gangguan interferensi elektromagnetik. (Sofana, 2011 : 33)



Gambar II.3 Kabel *Twisted-Pair*
(sumber : Sofana, 2011 : 32)

c. Kabel *coaxial*

Terdiri atas konduktor, *shield*, isolator dalam, dan isolator luar. Kabel *coaxial* memiliki sebuah konduktor tembaga di pusatnya. Konduktor digunakan untuk jalur transmisi data. Lapisan plastik digunakan sebagai isolasi antara konduktor pusat dan *shield* di sekelilingnya. *Shield* berupa jalinan logam/metal memblokir berbagai interferensi elektromagnetik yang berasal dari luar.

Kabel *coaxial* dibagi menjadi dua jenis yaitu *thickcoaxialcable* dan *thincoaxialcable*. *Thick coaxialcable* mempunyai diameter besar (sekitar 9.5 mm) dan biasanya berwarna kuning sedangkan *thincoaxialcable* mempunyai diameter lebih kecil (sekitar 5 mm) dan biasanya berwarna hitam.

Kabel *coaxial* saat ini mulai jarang digunakan. Salah satu penyebabnya adalah proses instalasi kabel yang cukup sulit, kecepatan transfer data yang kurang

memadai untuk kebutuhan saat ini, dan harganya cukup mahal dibandingkan kabel jenis *twisted pair*. Kecepatan transmisi data pada kabel *coaxial* lebih rendah dari kabel *Unshielded Twisted Pair* (UTP). Meskipun kabel *coaxial* sulit diinstall, namun cukup resisten terhadap sinyal interferensi, dan bisa lebih panjang dibandingkan kabel *twistedpair*. (Sofana, 2011 : 31)



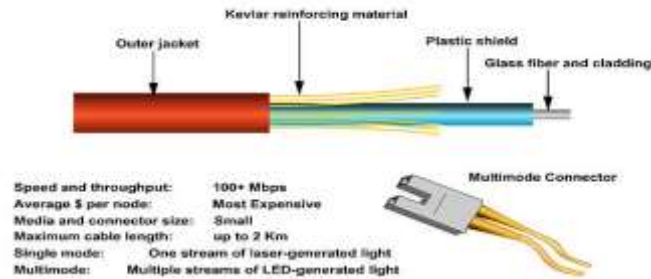
Gambar II.4 *Coaxialcable*

Sumber : (<http://blackgemi.com/wp-content/uploads/2009/03/adhoc1.gif>, 26 Desember 2014)

d. *Fiber optic*

Fiber optic (FO) merupakan jenis kabel yang terbuat dari sejenis bahan kaca atau plastik. Diameter sebuah *Fiber optic* (FO) sangat kecil, sekitar 120 mikrometer. *Fiber optic* (FO) diharapkan dapat mengatasi beberapa kelemahan kabel tembaga (*coaxial*, *twisted pair*). Kabel *Fiber optic* (FO) terdiri atas inti pusat berupa kaca tipis yang digunakan untuk membawa informasi dalam bentuk cahaya. *Fiber optic* (FO) dapat mentransmisikan cahaya (dan bukan sinyal listrik). Sumber cahaya biasanya berasal dari LASER atau LED. *Fiber optic* (FO) lebih tahan terhadap interferensi listrik, sehingga cocok untuk digunakan pada lingkungan yang memiliki interferensi listrik sangat besar.(Iwan sofana, 2011 : 38)

Fiber Optic Cable



Gambar II.5 *Fiber Optic*

Sumber : (<http://cooxmoon.files.wordpress.com/2010/06/untitled.jpg>, 25 Desember 2014)

2. Media *Wireless* (tanpa kabel) yang menggunakan sinyal frekuensi tinggi yaitu,
 - a. *Wireless Network* adalah jaringan komputer yang menggunakan gelombang radio sebagai media transmisi data, sehingga tidak memerlukan kabel. Jaringan Nirkabel biasanya di gunakan pada tempat – tempat umum seperti *cafe, restaurant, mall*. Jaringan Nirkabel memiliki beberapa keunggulan dan kekurangan. (Sofana,2011 :53).
 - b. Sinyal *microwave*, dengan frekuensi gelombang 1 – 500 GHz yang ditransmisikan dari dan ke *transceiver-transceiver*, juga secara *line-of-sight*. Sinyal gelombang mikro ini dikirimkan secara beranting beberapa kali oleh *repeater microwave* sebelum tiba di tujuan.
 - c. Sinyal *infra-red*, dengan frekuensi gelombang 500 GHz – 1 THz yang dapat membawa data dengan kecepatan 16 Mbps untuk transmisi satu arah, dan dengan kecepatan 1 Mbps untuk transmisi dua arah dengan jarak antara pengirim dan penerima hanya sekitar 30 meter.

Jenis media transmisi yang banyak digunakan untuk suatu *Local Area Network* adalah jenis *wires* atau transmisi dengan menggunakan kabel berupa kabel *twisted-pair*, kabel koaksial, ataupun kabel serat *optic* (Sofana,2011 :55).

II.1.5 Arsitektur Jaringan Komputer

Arsitektur jaringan komputer merupakan tata cara penggunaan perangkat keras dan perangkat lunak dalam jaringan agar satu komputer dengan komputer lainnya dapat melakukan komunikasi dan pertukaran data.

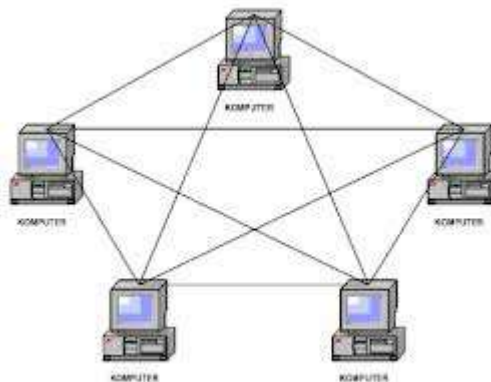
Ada tiga bentuk arsitektur yang umum digunakan dalam jaringan komputer yaitu,

a. Jaringan *peer to peer*

Pada jaringan *peer to peer* (Gambar II.6), semua komputer memiliki posisi setara / sejajar, dalam hierarki yang sama. Setiap komputer dapat menjadi klien terhadap komputer *peer* lainnya, setiap komputer dapat pula berbagi sumber daya dengan komputer yang berada dalam jaringan *peer-to-peer* ini. Sumber daya diletakkan secara desentralisasi pada setiap anggota jaringan, dan tidak memerlukan administrator jaringan.

Aliran informasi bisa mengalir di antara dua komputer secara langsung, di mana pun. Namun, jaringan ini tidak sepenuhnya bebas tanpa kontrol, masih bisa digunakan *password* untuk memproteksi *file* dan *folder*, dapat juga diatur agar orang-orang tertentu tidak bisa menggunakan periferal tertentu.

Karena kemudahan pemasangan, pemeliharaan, serta biaya, jaringan ini lebih populer untuk jaringan dengan jumlah komputer yang sedikit (sekitar 2 sampai 20 komputer).



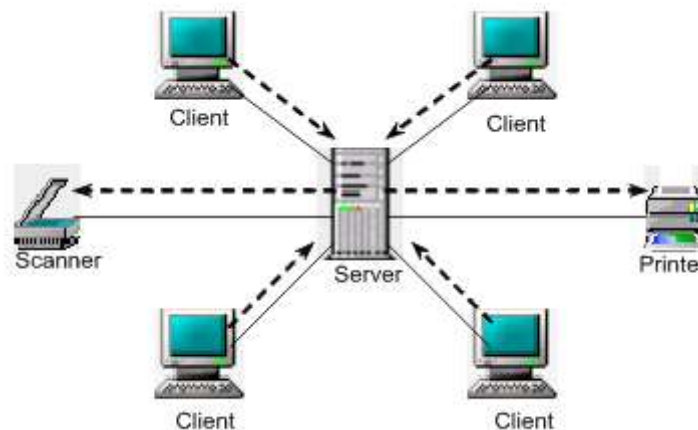
Gambar II.6 Jaringan *Peer-to-Peer*

(Sumber ; <http://agung-empire.mywapblog.com/belajar-mengenal-topologi-jaringan-kompu.shtml>, 27 Desember 2014)

b. Jaringan *client / server*

Pada jaringan *client/server* (Gambar II.6), perangkat lunak yang mengontrol keseluruhan kerja jaringan berada pada *server*. Jaringan ini dapat menghubungkan ratusan komputer dengan tingkat keamanan yang tidak dimungkinkan dalam jaringan *peer-to-peer*.

Jaringan ini bisa diatur sehingga setiap klien harus *log on* ke *server* sebelum bisa memanfaatkan sumber daya yang terhubung ke *server*. *Server* lalu mengotentikasi klien dan memverifikasi bahwa komputer yang digunakan klien tersebut memiliki izin untuk *log on* ke jaringan, dengan memeriksa *username* dan *password* klien tersebut terhadap *database* pada *server*.



Gambar II.7 Jaringan *Client / Server*

(Sumber ; <http://agung-empire.mywapblog.com/belajar-mengenal-topologi-jaringan-kompu.shtml>, 27 Desember 2014)

c. Jaringan *hybrid*

Jaringan ini merupakan gabungan dari sifat pada jaringan *peer to peer* dan *client/server*. *Workgroup* yang terdiri dari beberapa komputer yang saling terhubung dapat mengelola sumber daya tanpa membutuhkan otorisasi dari *administrator* jaringan atau *server*. Pada jenis jaringan ini, terdapat pula sifat dari jaringan *client / server* sedemikian sehingga tingkat keamanan dapat lebih terjaga dan adanya *server* yang mempunyai suatu fungsi layanan tertentu, seperti sebagai *file server*, *print server*, *database server*, *mail server*, dan lainnya.

Sifat jaringan *peer to peer* digunakan untuk hubungan antara setiap komputer yang terhubung dalam jaringan komputer yang ada, sehingga komunikasi data terjadi antar komputer dengan hierarki yang sama karena setiap komputer dapat berfungsi sebagai *server* maupun klien. Sedangkan, sifat jaringan

client/server digunakan untuk memfasilitasi setiap komputer dengan hubungan *internet*. Jadi, komputer *server* dihubungkan ke *Internet Service Provider (ISP)*, sehingga komputer klien yang terhubung dalam jaringan juga dapat melakukan akses *internet*.

II.1.6 Topologi Jaringan Komputer

Topologi jaringan adalah tata cara komputer dan sumber daya lainnya dihubungkan dalam jaringan. Ada dua jenis topologi jaringan yaitu,

- a. Topologi fisik, yaitu tata cara komputer dan peralatan lainnya dihubungkan secara fisik melalui kabel / media transmisi sehingga bisa saling berkomunikasi, dan
- b. Topologi logik, yaitu tata cara komputer dan peralatan dalam jaringan berkomunikasi dan tata cara data ditransmisikan melalui jaringan.

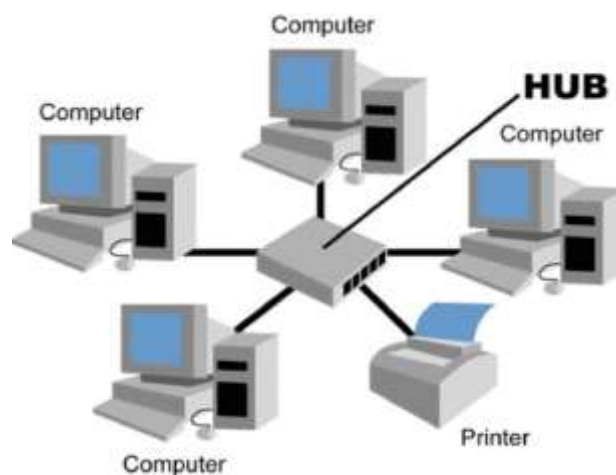
Yang akan dibahas dalam pembahasan ini adalah topologi fisik, yang menyangkut bagaimana sistem atau komputer dan periperalnya dihubungkan satu dengan yang lainnya.

Topologi fisik secara umum ada empat jenis yaitu,

1. Topologi *Star*

Dalam topologi *Star*, sebuah terminal pusat bertindak sebagai pengatur dan pengendali semua komunikasi data yang terjadi. Terminal-terminal lain terhubung padanya dan pengiriman data dari satu terminal lainnya melalui terminal pusat. Terminal pusat akan menyediakan jalur komunikasi khusus pada dua terminal yang akan berkomunikasi.(sofana ,2011:145)

Beberapa keuntungan dari topologi *Star* ini adalah kemudahan untuk menambah peralatan ke jaringan dan jika terjadi kerusakan pada salah satu kabel hanya akan mempengaruhi komputer yang dihubungkan kabel tersebut. Namun, kerugian akan terjadi pada jenis topologi ini apabila *hub* terganggu / rusak, maka secara keseluruhan jaringan akan terganggu. (Lewis,2011:122)



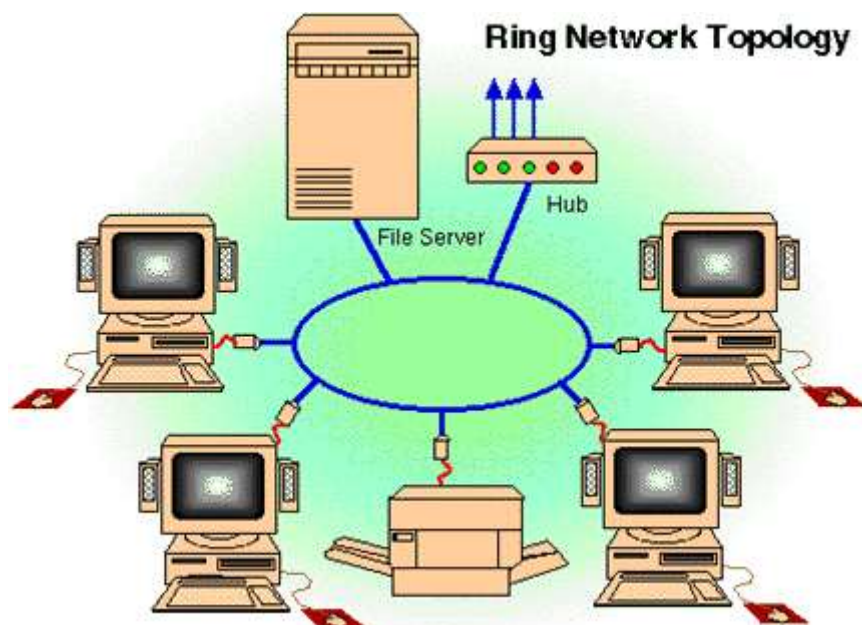
Gambar II.8 Topologi Star

(Sumber <http://www.aldo-expert.com/blog-artikel/5-jenis-topologi-dalam-jaringan-komputer-local.html>)

2. Topologi Ring

Pada topologi *Ring* (Gambar 2.8), setiap simpul dalam jaringan dihubungkan seperti halnya cincin, sehingga jika terjadi transmisi data, maka data akan mengelilingi *ring* sampai tiba di komputer / alamat tujuan.

Keuntungan dari topologi jaringan ini antara lain data dapat dikirimkan dengan kecepatan tinggi tanpa terjadi tumbukan data (*data collision*). Namun, kerugiannya adalah jika kabel mengalami kerusakan, maka jaringan akan terganggu. Jenis topologi ini tidak begitu umum digunakan dalam jaringan komputer.



Gambar II.9 Topologi Ring

(<http://www.aldo-expert.com/blog-artikel/5-jenis-topologi-dalam-jaringan-komputer-local.html> : 18 januari 2015)

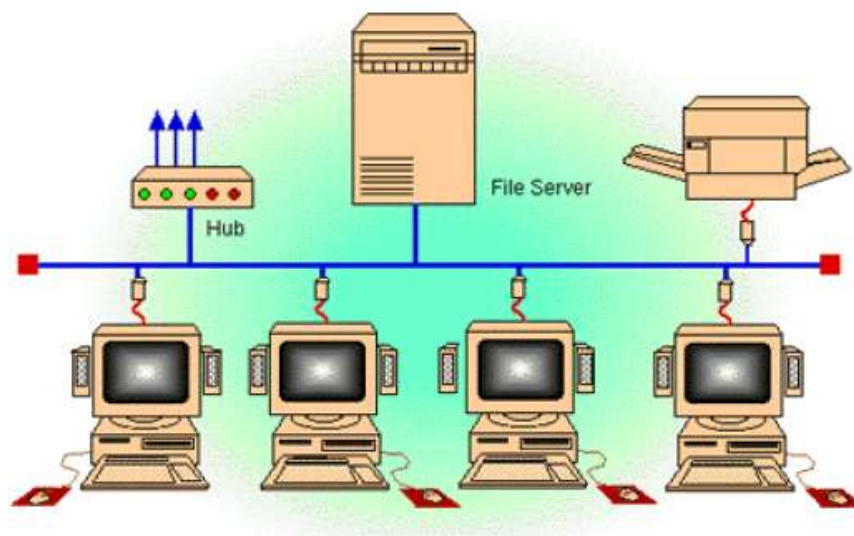
3. Topologi Bus

Pada topologi *Bus* (Gambar II.9), setiap simpul dalam jaringan dihubungkan dengan suatu kabel utama yang disebut bus jaringan atau sering juga disebut sebagai *backbone*, yang pada setiap ujungnya dipasang *terminator* yang berfungsi untuk menyerap sinyal-sinyal yang melewati ujung tersebut. Disebut *bus* karena orang-orang dalam sebuah *bus* dapat naik atau berhenti pada perhentian manapun sepanjang *route*.

Dalam topologi *bus*, sinyal dikirimkan secara *broadcast* ke semua simpul dalam jaringan, tapi hanya simpul / komputer tujuan yang dapat menanggapi sinyal tersebut.

Keuntungan dari topologi ini adalah kemudahan untuk menambah atau menghapus komputer atau peralatan dari jaringan serta kemudahan pemasangan,

sehingga paling banyak digunakan. Topologi *bus* baik untuk peralatan / sistem yang secara fisik terletak dekat satu dengan yang lainnya. Kerugiannya, apabila kabel utama rusak, maka jaringan akan *down* secara keseluruhan. Selain itu, waktu akses juga lebih lambat dibandingkan dengan topologi lainnya.

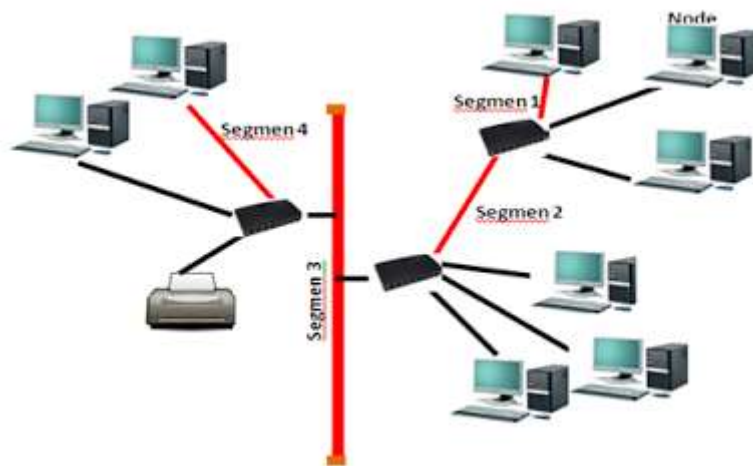


Gambar II.10 Topologi *Bus*

(<http://www.erlangga.co.id/blog/viewtopic.php>: 18 Januari 2015)

4. Topologi *Hierarchical Tree*

Dari namanya *network* ini berbentuk seperti pohon yang bercabang, yang terdiri dari *control node* dihubungkan dengan *node* yang lain secara berjenjang. *Central node* biasanya berupa *large* komputer atau komputer *mainframe* sebagai komputer *host* yang merupakan jenjang tertinggi (*top hierachical*) yang bertugas mengkoordinasi dan mengendalikan *node* jenjang di bawahnya yang dapat berupa mini komputer atau mikro komputer. Penjelasan topologi *hierachical tree* ditunjukkan pada gambar 2.11 berikut,



Gambar II.11 *Topologi Hierarchical Tree*

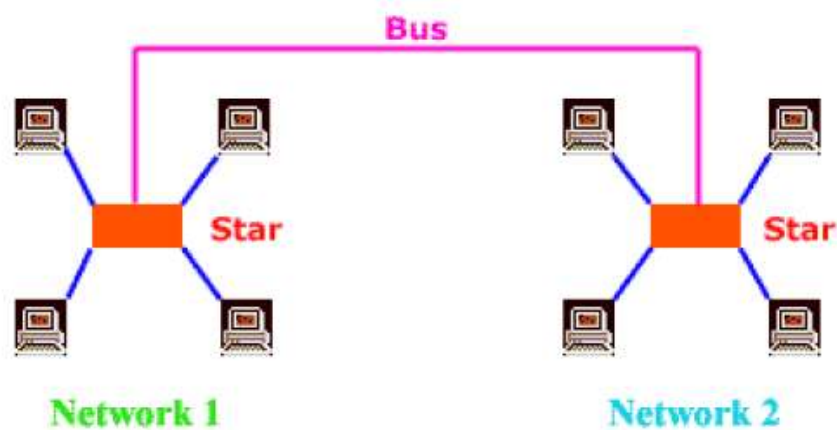
5. (<http://www.erlangga.co.id/blog/viewtopic.php>: 18 Januari 2015)

6. Topologi *Hybrid* atau *Meta*

Jenis topologi *Hybrid* merupakan jenis kombinasi dari jenis-jenis topologi *star*, *ring*, atau *bus*. Jenis topologi inilah yang paling banyak digunakan dalam dunia nyata.

7. Topologi *Mesh*

Topologi *Mesh* menghubungkan setiap komputer secara *point-to-point*. Artinya semua komputer akan saling terhubung satu-satu sehingga tidak dijumpai ada *link* yang terputus. (Sofana, 2011 :14)



Gambar II.12 *Topologi Hybrid* atau *Meta*

(<http://www.erlangga.co.id/blog/viewtopic.php>: 18 Januari 2015)

II.2 Dasar – dasar Jaringan TCP / IP

II.2.1 Arsitektur Internet Protocol (IP)

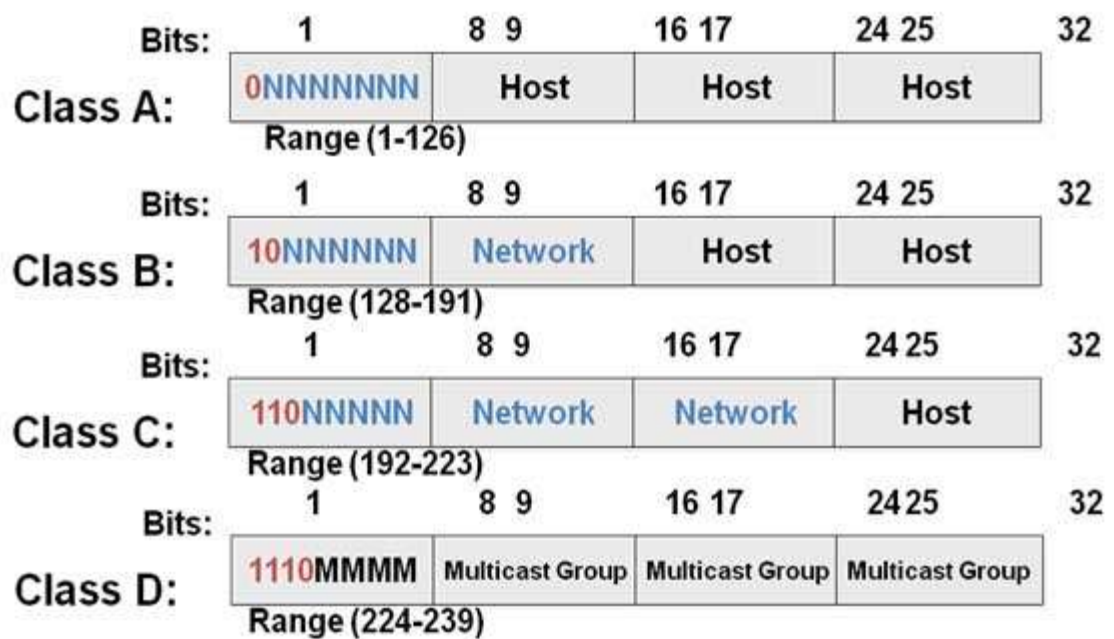
Internet Protocol (IP) merupakan *protokol open system* yang terkenal karena banyak digunakan untuk melakukan hubungan koneksi antar jaringan dan sesuai dengan standar komunikasi LAN maupun WAN. *Internet Protocol*, dua protokol diantaranya yang paling terkenal yaitu *Internet Protocol* (IP) dan *Transmission Control Protocol* (TCP). *Internet Protocol* juga sangat sesuai dengan aplikasi pada *lower layer protocol* seperti TCP dan IP maupun pada *common* aplikasi seperti *electronic mail*, *terminal emulation*, dan *file transfer*. (Odom, wendel, 2009:34).

II.2.2 IP Address

Masing-masing *host* mempunyai *IP address* untuk mengidentifikasi suatu *host* dalam melakukan proses koneksi dalam jaringan *TCP / IP*. Sedangkan

masing-masing *IP address* di dalamnya terdapat *Network ID* dan *Host ID*. (Odom, wendel, 2005:64).

Network ID menunjukkan letak atau tempat sistem berada yang terdapat dalam *IP Router* dalam jaringan fisik yang sama. *Host ID* ini lebih dikenal dengan *workstation*, *server*, *router* dan *host TCP / IP* yang lainnya dalam suatu jaringan.



Gambar II.13 *IP Address* 32 Bit dan Dikelompokkan Dalam 4 Oktet
(sumber Odom, wendel, 2009:35).

Suatu *IP Address* terdiri dari 32 *bit* yang bekerja dalam suatu urutan waktu. Dari 32 *bit* kemudian dipecah menjadi 8 *bit* atau lebih dikenal dengan oktet dimana masing-masing oktet dikonversikan dalam bilangan desimal yang berkisar antara 0 sampai 255 yang diperlihatkan gambar 2.15 di atas. Berikut contoh konversi *IP Address*. . (sofana, iwan; 2011:87).

Biner ke Desimal

$$\begin{array}{r}
 \begin{array}{cccc} 1 & 1 & 1 & 1 \end{array} \quad \begin{array}{cccc} 1 & 1 & 1 & 1 \end{array} \\
 \swarrow \quad \searrow \quad \swarrow \quad \searrow \quad \swarrow \quad \searrow \quad \swarrow \quad \searrow \\
 = 1.2^7 + 1.2^6 + 1.2^5 + 1.2^4 + 1.2^3 + 1.2^2 + 1.2^1 + 1.2^0 \\
 = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 \\
 = 255 \\
 \\
 \begin{array}{cccc} 1 & 1 & 0 & 0 \end{array} \quad \begin{array}{cccc} 0 & 0 & 0 & 1 \end{array} \\
 \swarrow \quad \searrow \quad \swarrow \quad \searrow \quad \swarrow \quad \searrow \quad \swarrow \quad \searrow \\
 = 1.2^7 + 1.2^6 + 0.2^5 + 0.2^4 + 0.2^3 + 0.2^2 + 0.2^1 + 1.2^0 \\
 = 128 + 64 + 0 + 0 + 0 + 0 + 0 + 1 \\
 = 193
 \end{array}$$

Gambar II.14 Contoh Algoritma Untuk Konversi ke IP Address Dalam Format Bilangan Desimal

(<http://hyperpost.blogspot.com/2014/04/konversi-bilangan-biner-octal-desimal.html> 30 Desember 2014)

II. 3 Citra

Bidang pengolahan citra secara digital (*digital image processing*) tercatat dimulai pada awal tahun 1921, pada waktu itu sebuah foto untuk pertama kalinya berhasil ditransmisikan secara digital melalui kabel laut dari kota *New York* ke kota *London* (*Bartlane Cable Picture Trasmision System*). Keuntungan utama yang dirasakan pada waktu itu adalah pengurangan waktu pengiriman foto dari sekitar satu minggu menjadi kurang dari 3 jam. Foto tersebut dikirim dalam bentuk kode digital, selanjutnya diubah kembali oleh suatu printer telegraph pada sisi penerima. Masalah yang muncul pada saat itu berkisar pada teknik transmisi data secara digital serta teknik reproduksi pada sisi penerima untuk mendapatkan satu resolusi gambar yang baik. Walaupun minat dalam bidang ini telah dimulai sejak tahun 1921, tetapi perkembangannya secara pesat baru tercatat pada sekitar tahun 1960. Pada saat itu teknologi komputer telah dianggap memenuhi suatu

kecepatan proses serta kapasitas memori yang dibutuhkan oleh berbagai algoritma pengolahan citra. Sejak itulah berbagai jenis aplikasi mulai dikembangkan, yang secara umum dapat dikelompokkan dalam dua jenis kegiatan yaitu:

1. Memperbaiki kualitas suatu gambar sehingga dapat lebih mudah diinterpretasikan oleh mata manusia.
2. Mengolah informasi yang terdapat pada suatu gambar untuk keperluan pengenalan objek secara otomatis oleh suatu mesin. (Prasetyo, 2011 ; 1)

II.3.1 Definisi Citra

Citra adalah representasi visual yang terdiri dari sekumpulan piksel atau titik berwarna dalam bentuk dua dimensi. Citra merupakan representasi dua dimensi (2-D) dari intensitas cahaya yang dinyatakan dengan fungsi $f(x,y)$, dimana x dan y merupakan koordinat spasial dan nilai fungsi f menunjuk pada titik (x,y) . (Putra, Darma, 2010:19)

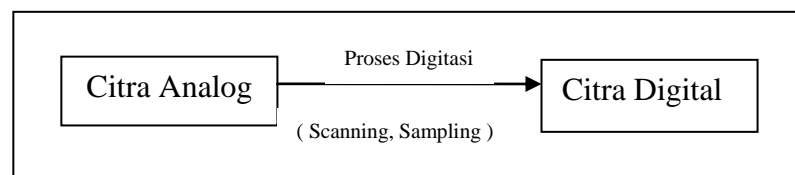
Sebuah citra dapat didefinisikan sebagai fungsi dua dimensi $f(x,y)$, dimana x dan y adalah koordinat spasial, dan amplitudo dari f pada sembarang pasangan koordinat (x,y) disebut *intensity* intensitas atau *gray level* (level keabuan) dari citra pada titik tersebut. Ketika x , y dan nilai intensitas dari f adalah semua batas, *discrete qualities*, kita sebut citra tersebut digital image (citra digital). Citra digital terdiri dari sejumlah elemen tertentu, setiap elemen mempunyai nilai dan lokasi tertentu. Elemen-elemen ini disebut *pictures*

elements, image elements, pels, dan pixels. Pixels adalah istilah yang sudah digunakan secara luas untuk menyatakan elemen citra digital. (Prasetyo, 2011 : 2)

II.3.2 Citra Digital

Citra digital merupakan suatu *array* dua dimensi atau suatu *matriks* yang elemen-elemennya menyatakan tingkat keabuan dari elemen gambar. Jadi informasi terkadang bersifat diskrit.

Citra digital terdiri-dari sinyal-sinyal frekuensi elektromagnetis yang sudah di-*sampling* dan ukuran *pixel* dari citra tersebut sudah dapat ditentukan. *Sampling* merupakan proses pembentukan citra digital dari citra analog. Suatu citra yang dicetak diatas kertas disebut dengan citra analog, jika citra analog tersebut di-*scan* dengan alat scanner maka akan menjadi citra digital. Dengan demikian, *scanner* merupakan alat *sampling*. Proses pembentukan citra digital dari citra analog diperlihatkan pada Gambar 2.14



Gambar II.15 Pembentukan Citra Digital Dari Citra Analog
(Prasetyo, 2011 : 3)

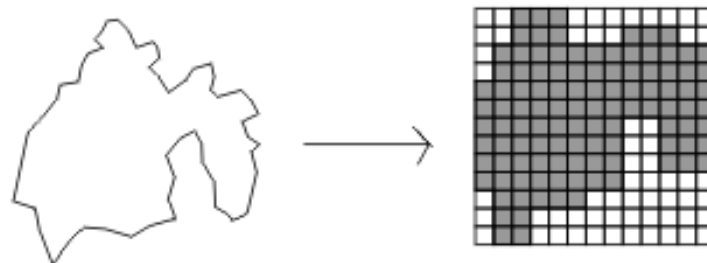
Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada

monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu pita magnetik.

II.3.3 Konversi Citra Analog ke Citra Digital

Citra digital tidak selalu merupakan hasil langsung data rekaman suatu sistem. Kadang-kadang hasil rekaman data bersifat kontinu seperti gambar pada monitor televisi, foto sinar-x dan lain sebagainya. Dengan demikian untuk mendapatkan suatu citra digital diperlukan suatu proses konversi, sehingga citra tersebut selanjutnya dapat diproses dengan komputer.

Untuk mengubah citra bersifat kontinu menjadi citra digital diperlukan proses pembentukan kisi-kisi arah *horizontal* dan *vertikal*, sehingga diperoleh gambar dalam bentuk *array* dua dimensi. Proses tersebut dikenal sebagai proses digitasi atau *sampling*. Pada digitasi atau *sampling* dilakukan pembagian gambar kepada bagian kecil supaya dapat mewakili kandungan gambar. Pembagian dilakukan kepada segiempat kecil (*grid*) yang dipanggil *pixel* (*picture element or pixel*). Setiap *pixel* adalah *sample* gambar asal yang diambil dari domain ruang (*spatial domain*). Gambar berikut menunjukkan proses digitasi :



Citra analog

Citra digital

Gambar II.16 Proses *Digitasi*

Proses yang diperlukan selanjutnya adalah proses kuantisasi. Dalam proses ini tingkat keabuan setiap *pixel* dinyatakan dengan suatu harga integer. Batas-batas harga integer atau besarnya daerah tingkat keabuan yang digunakan untuk menyatakan tingkat keabuan *pixel* akan menentukan resolusi kecerahan dari gambar yang diperoleh. Kalau digunakan 3 bit untuk menyimpan harga integer tersebut, maka akan diperoleh sebanyak 8 tingkat keabuan. Makin besar tingkat keabuan yang digunakan makin baik gambar yang akan diperoleh karena kontinuitas dari tingkat keabuan akan semakin tinggi sehingga mendekati citra aslinya. (Prasetyo, 2011 : 6-7)

II.3.4 Definisi *Pixel*

Pixel didefinisikan sebagai unsur citra (*image*) atau unsur pengindraan, yang menunjuk pada satuan terkecil yang dapat dialamati dalam kegunaan grafik. Pada citra berformat *bitmap*, sekumpulan *pixel* adalah titik-titik yang digunakan untuk membangun suatu citra. (Putra, Darma, 2010:35)

II.3.5 Resolusi

Kualitas sebuah citra ditentukan pula oleh resolusi. Resolusi adalah banyaknya *pixel* yang menghasilkan sebuah citra dalam sebuah layar atau printer. Semakin banyak jumlah *pixel*-nya, maka semakin tinggi resolusinya dan akan dihasilkan citra yang lebih baik dan lebih halus. Resolusi yang ideal merupakan keseimbangan antara kualitas dengan ukuran penyimpanan citra tersebut.

II.3. 6 Format Penyimpanan Citra

Format penyimpanan citra yaitu TIFF (*Tagged Image File Format*), GIF (*Graphics Interchange Format*), JPEG (*Joint Photographic Experts Group*), PNG (*Portable Network Graphics*), PCD (*Photo CD*), BMP (*Bitmap*), PIXAR (*Pixar Image Computers*), WMF (*Windows Metafile*).

II.3. 7 Format File BMP

BMP adalah format *file* gambar standar untuk sistem operasi *Windows*. Format *file* ini dikembangkan oleh *Microsoft* untuk menyimpan gambar (*bitmap*) dan memungkinkan *Windows* untuk menampilkan kembali gambar tersebut. Struktur dari *file* BMP terdiri dari *BitmapFileHeader* yang mengandung informasi mengenai *file*, *BitmapInfoHeader* yang menyimpan informasi mengenai gambar (seperti dimensi, warna dan lain – lain), tabel warna yang didefinisikan sebagai *array* dari struktur *RGBQUAD*, dan sisanya adalah data gambar.

Tabel berikut ini akan memperlihatkan informasi lengkap mengenai struktur *file* BMP untuk gambar yang berukuran 100 x 100 *piksel*, 256 warna, dan tanpa kompresi. Kolom mulai berisi posisi *byte* di dalam *file* dimana elemen data dari struktur yang dijelaskan dimulai. Kolom ukuran berisi jumlah *byte* yang digunakan oleh elemen data tersebut. Kolom nama berisi nama dengan elemen data sesuai dengan dokumentasi *Microsoft API*. Kolom nilai standar berisi nilai standar yang mungkin terisi pada elemen data tersebut. Kolom keterangan berisi penjelasan singkat mengenai elemen data yang dimaksud.

Tabel II.1 Struktur *BitmapFileHeader*

Mulai	Ukuran (Byte)	Nama	Nilai Standar	Keterangan
1	2	<i>BfType</i>	19778	ASCII = 'BM'.
3	4	<i>BfSize</i>	?	Ukuran <i>file</i> dalam <i>byte</i> .
7	2	<i>BfReserved1</i>	0	Tidak digunakan.
9	2	<i>BfReserved2</i>	0	Tidak digunakan.
11	4	<i>BfOffBits</i>	1078	Posisi <i>byte</i> dimana data gambar berada

Tabel II.2 Struktur *BitmapInfoHeader*

Mulai	Ukuran (Byte)	Nama	Nilai Standar	Keterangan
15	4	<i>BiSize</i>	40	Ukuran dari info header dalam <i>byte</i>
19	4	<i>BiWidth</i>	100	Lebar gambar dalam piksel
23	4	<i>BiHeight</i>	100	Tinggi gambar dalam piksel
27	2	<i>BiPlanes</i>	1	Jumlah bidang gambar
29	2	<i>BiBitCount</i>	8	Jumlah bit per piksel
31	4	<i>BitCompression</i>	0	Jenis kompresi
35	4	<i>BiSizeImage</i>	0	Ukuran data gambar
39	4	<i>BiXPelsPerMeter</i>	0	Resolusi horizontal dalam piksel meter
43	4	<i>BiYPelPerMeter</i>	0	Resolusi vertical dalam piksel meter
47	4	<i>BiClrUsed</i>	0	Jumlah warna yang digunakan
51	4	<i>BiClrImportant</i>	0	Jumlah warna penting

(Prasetyo, 2011 : 11)

Jumlah warna yang terdapat pada gambar ditentukan oleh *BiBitCount*. Kemungkinan nilai untuk *BiBitCount* adalah 1 (hitam/putih), 4 (16 warna), 8 (256 warna), dan 24 (16,7 juta warna). Elemen data *BiBitCount* sekaligus menentukan apakah pada *file* BMP terdapat tabel warna atau tidak, sekaligus susunan dari tabel warnanya.

Untuk gambar 1 *bit*, tabel warna hanya berisi dua warna (biasanya hitam dan putih). Jika setiap *bit* dari data gambar bernilai 0 maka warna yang

ditunjuknya adalah warna pertama di dalam tabel warna. Jika setiap *bit* dari data gambar bernilai 1 maka warna yang ditunjuknya adalah warna kedua yang terdapat di dalam tabel warna.

Pada gambar 4 *bit*, tabel warnanya berisikan 16 warna. Setiap *byte* yang terdapat pada data gambar mewakili dua piksel. *Byte-byte* tersebut dibagi menjadi dua bagian, masing – masing 4 bit. *Bit – bit* tadi menunjukkan ke warna – warna yang terdapat pada tabel warna.

Pada gambar 8 bit, setiap *byte* mewakili satu piksel. Nilai dari setiap *byte* tadi menunjuk ke salah satu warna yang terdapat pada tabel warna yang di dalamnya berisi 256 warna.

Untuk gambar 24 *bit*, 3 *byte* digunakan untuk mewakili satu piksel. *Byte* yang pertama mewakili unsur warna merah, *byte* yang kedua mewakili unsur warna hijau, dan *byte* ketiga mewakili unsure warna biru. Pada gambar 24 *bit*, tabel warna tidak dibutuhkan karena setiap piksel mengandung unsur warna merah, hijau dan biru yang sebenarnya.

Tabel warna sendiri dibentuk dari struktur *RGBQUAD* yang disusun dalam bentuk *array*. Struktur dari *RGHQUAD* dapat dilihat pada tabel dibawah ini.

Tabel II.3 Struktur *RGBQUAD*

Mulai	Ukuran (Byte)	Nama	Nilai Standar	Keterangan
1	1	<i>RGBBlue</i>	?	Intensitas warna biru
2	1	<i>RGBGreen</i>	?	Intensitas warna hijau
3	1	<i>RGBBlue</i>	?	Intensitas warna merah
4	1	<i>RGBReserved</i>	0	Tidak digunakan

II.3.8 Format File GIF

Format file GIF (Graphics Interchange Format) merupakan hasil rancangan *CompuServe Incorporated*. *Format* ini dirancang untuk memudahkan pertukaran citra *bitmap* antarkomputer. GIF hanya mendukung resolusi warna sampai 256 warna (8-bit).

Format file GIF memiliki dua versi yaitu GIF 87a dan GIF89a. Versi GIF89a diperkenalkan pada bulan Juli 1989 merupakan perbaikan dari versi GIF87a. Pada GIF89a ditambahkan kemampuan untuk menampilkan citra dengan latar belakang transparan (*background transparency*), penyimpanan data citra secara *interlaced* dan kemampuan untuk menampilkan citra animasi.

GIF menggunakan *variable-length code* yang merupakan modifikasi dari algoritma LZW (*Lemple-Ziv Welch*) untuk mengkompresi data citra. Teknik kompresi data ini mampu menghasilkan kompresi yang baik dan merupakan teknik kompresi yang mampu mengembalikan data sama persis dengan aslinya (*lossless data comperssion*).

Format lengkap *file* citra GIF dapat dilihat pada tabel berikut ini.

Tabel II.4 Format *File* GIF

<i>Name</i>	<i>Size</i>	<i>Description</i>
<i>Signature</i>	6 bytes	'GIF87a' or 'GIF89a'
<i>Global Descriptor</i>	7 bytes	Always present
<i>Width</i>	2 bytes	Width in pixels
<i>Height</i>	2 bytes	Height in pixels
<i>Flags</i>	1 bytes	Global descriptor flags
<i>Global Color Map</i>	Bit 7	=1 if Global ColorMap exists (should be true in almost all cases)

			=0 if default map is used. Or if every citra has a Local Color Map
		<i>Color Resolution Bits</i>	<i>Bit 6-4</i> +1 = significant bits per color in Global Color Map
		<i>Reserved</i>	<i>Bit 3</i> = 0
		<i>Pixel Bits</i>	<i>Bit 2-0</i> +1 = Color Depth, Number of Global Colors := $2^{\text{colordepth}}$
		<i>Background Color</i>	<i>1 byte</i> Background color number (from Global Color Map or default map)
		<i>Aspect Ratio</i>	<i>1 byte</i> Usually =0
		<i>Global Color Map</i>	<i>Number of Global Colors * 3</i> Global color table, present only when Global Descriptor Flags Global Color Map =1
		<i>Red</i>	<i>1 byte</i> Red intensity of color (not necessarily 8 significant bits)
		<i>Green</i>	<i>1 byte</i> Green intensity of color (not necessarily 8 significant bits)
		<i>Blue</i>	<i>1 byte</i> Blue intensity of color (not necessarily 8 significant bits)
<i>Repeated Number Of Global Colors Times</i>			
		<i>Extension Blocks</i>	<i>Cannot be pre calculated</i> Optional, may be repeated any number of times. Read first byte to check its precense
		<i>Header</i>	<i>1 byte</i> = \$21
		<i>Function Code</i>	<i>1 byte</i> There is a list of known function codes
		<i>Length</i>	<i>1 byte</i> >0 !
		<i>Data</i>	<i>Length bytes</i> Interpretation of this data depends on its function code
<i>Repeated any number of times. Read first byte to check for terminator</i>			
		<i>Terminator</i>	<i>1 byte</i> = 0
		<i>Local Descriptor</i>	<i>10 bytes</i> Local descriptor, always present

	<i>Header</i>	<i>1 bytes</i>	<i>=$\\$2C$</i>
	<i>Pos X</i>	<i>2 bytes</i>	<i>Horizontal position of citra in global citra</i>
	<i>Pos Y</i>	<i>2 bytes</i>	<i>Vertical position of citra in global citra</i>
	<i>Width</i>	<i>2 bytes</i>	<i>Width of citra</i>
	<i>Height</i>	<i>2 bytes</i>	<i>Height of citra</i>
	<i>Flags</i>	<i>1 bytes</i>	<i>Local descriptor Flags</i>
	<i>Local Color Map</i>	<i>Bit 7</i>	<i>=1 if Local Color Map exists =0 if Global Color Map (or default map) is used</i>
	<i>Interlaced Citra</i>	<i>Bit 6</i>	<i>=1 Interlaced! =0 Non Interlaced</i>
	<i>Sorted</i>	<i>Bit 5</i>	<i>Usually =0</i>
	<i>Reserved</i>	<i>Bit 4-3</i>	<i>=0</i>
	<i>Pixel Bits</i>	<i>Bit 2-0</i>	<i>-1 = Color Depth, Number of Local Colors := $2^{\text{color depth}}$</i>
<i>Local Color Map</i>		<i>Number of Local Color * 3</i>	<i>Local color table, present only when local Descriptor Flags. Local Color Map =1</i>
	<i>Red</i>	<i>1 byte</i>	<i>Red intensity of color</i>
	<i>Green</i>	<i>1 byte</i>	<i>Green intensity of color</i>
	<i>Blue</i>	<i>1 byte</i>	<i>Blue intensity of color</i>
<i>Repeated number of Local Color times</i>			
<i>Rasted DataBlock</i>		<i>Cannot be pre calculated</i>	<i>Always present</i>
	<i>Initial Code Size</i>	<i>1 byte</i>	<i>Usually = Color Depth, except for balck & white, where it is 2!</i>
	<i>Length</i>	<i>1 byte</i>	<i>>0</i>
	<i>Data</i>	<i>Length bytes</i>	<i>The pixel data bit stream. See decompression and compression schemes</i>
<i>Repeated any number of times, read first byte to check for terminator</i>			
	<i>Terminator</i>	<i>1 byte</i>	<i>=0!</i>

Extension Blocks	<i>Cannot be pre calculated</i>	<i>Optional. Read first byte to check its presence</i>
<i>Format is identical to the Extension Block above</i>		
Terminator	<i>1 byte</i>	<i>=\$3B</i>

(sumber : Prasetyo, 2011 : 12)

II.4 *Depth First Search*

Pada *Depth First Search* (DFS), proses akan dilakukan pada semua anaknya sebelum dilakukan pencarian ke *node-node* (titik) yang selevel. Pencarian dimulai dari *node* akar ke *level* yang lebih tinggi. Proses ini diulangi terus hingga ditemukannya solusi. Stack atau tumpukan adalah struktur data yang setiap proses baik penambahan maupun penghapusan hanya bisa dilakukan dari posisi teratas tumpukan. Cara kerja stack adalah LIFO (*Last In First Out*), dimana data yang terakhir masuk akan keluar pertama.

Berikut analisis ruang dan waktu untuk metode pencarian DFS :

1. Diasumsikan :
 - Pohon pelacakan memiliki cabang yang selalu sama, yaitu sebanyak b .
 - Tujuan dicapai pada level ke- d
2. Analisis Ruang
 - Setelah berjalan 1 langkah, *stack* akan berisi b *node*.
 - Setelah berjalan 2 langkah, *stack* akan berisi $(b-1) + b$ *node*.
 - Setelah berjalan 3 langkah, *stack* akan berisi $(b-1) + (b-1) + b$ *node*.
 - Setelah berjalan d langkah, *stack* akan berisi $(b-1) * d + 1$ *node*, mencapai maksimum.

3. Analisis Waktu

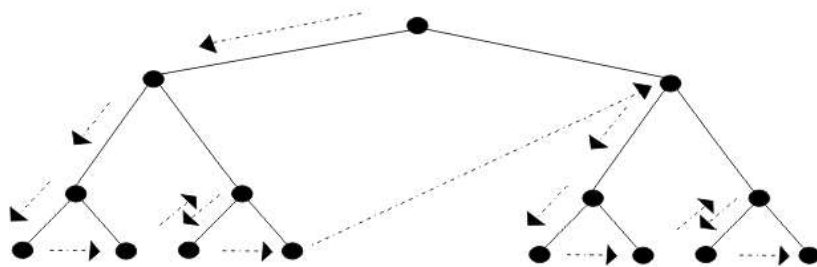
- Pada kasus terbaik, DFS akan mencapai tujuan pada kedalaman d pertama, sehingga dibutuhkan pencarian sebanyak $d + 1$ *node*.
- Pada kasus terburuk, DFS akan mencapai tujuan pada kedalaman d pada *node* terakhir, sehingga dibutuhkan pencarian sebanyak $1 + b + b^2 + b^3 + \dots + b^d = (b^{d+1} - 1) / (b - 1)$

Keuntungan dari metode ini adalah :

1. Membutuhkan memori yang relatif kecil, karena hanya *node-node* pada lintasan yang aktif saja yang disimpan.
2. Secara kebetulan, metode DFS akan menemukan solusi tanpa harus menguji lebih banyak lagi dalam ruang keadaan.

Kelemahan dari metode ini adalah :

1. Memungkinkan tidak ditemukannya tujuan yang diharapkan.
2. Hanya akan mendapatkan 1 solusi pada setiap pencarian.



Gambar II.17 Pohon Pencarian *Depth First Search*
(Prasetyo, 2011 : 23)

II.4.1 Penerapan DFS Dalam Permainan *Halma*

Pada permainan *halma* proses pencarian langkah terpendek digunakan metode *Depth First Search*, proses pencarian dari awal dengan menelusuri kedalam sejauh mungkin sebelum kembali ke pencarian awal. Metode ini digunakan untuk mencari langkah terpendek yang ditempuh oleh biji pemain ke daerah tujuan.

II.5 Sejarah *Halma*

Halma ditemukan oleh seorang profesor berkebangsaan Amerika dari Boston, Dr. *George Howard Monks* (1853 – 1933) diantara tahun 1883 dan 1884. *Monk* adalah seorang ahli bedah spesialisasi bagian dada di *Harvard Medical School*. Saudara *George, Robert Monks* berada di Inggris pada tahun 1883 atau 1884 dan *Robert* menulis surat kepada saudaranya dan mendeskripsikan permainan Inggris '*Hoppity*'. G.W. *Monks* mengambil beberapa ketentuan dari *Hoppity* dan membangun *Halma*.

Dr. *Thomas Hill* (1818 – 1891), seorang ahli matematika, guru dan pendeta, membantu merancang permainan ini. Dr. *Thomas Hill* menamakan permainan ini '*Halma*' yang berasal bahasa Yunani yang berarti melompat. *Hill* adalah presiden dari *Harvard College* antara tahun 1862 – 1868 dan merupakan ayah mertua dari *Robert Monks*.

Halma pertama kali dipublikasi di Amerika Serikat (AS) pada tahun 1885 oleh *E.I. Horsman Company*. Terdapat kontroversi pada *Halma* di AS, dimana *Milton Bradley Company* juga mengklaim hak atas permainan ini. Tidak jelas

apakah terjadi perang hak legal, tetapi kemudian *Milton Bradley* memproduksi dan memasarkan versi modifikasi dari *Halma* dengan nama *Eckha* pada tahun 1889.

Parker Bros mengklaim bahwa *George H. Monks* menjual hak paten atas *Halma* kepada mereka, namun hal itu tidak diverifikasi.

Di Inggris, *Spears Co.* memproduksi permainan *Halma* pada bulan Juli 1893. *Halma* adalah satu-satunya permainan klasik pada abad 19 yang berasal dari Amerika Serikat dan dikenal secara internasional.