BAB II

LANDASAN TEORI

2.1. Jaringan LAN

Jaringan LAN adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung dan biasanya jangkauannya tidak lebih dari 1 kilometer persegi. Beberapa model konfigurasi LAN biasanya berupa sebuah komputer yang dijadikan sebagai *file server* yang digunakan untuk menyimpan perangkat lunak ataupun sebagai perangkat lunak yang dapat digunakan oleh komputer-komputer yang terhubung ke dalam jaringan lokal.

Komputer-komputer yang terhubung dengan suatu *file server* biasanya disebut *workstation*. Biasanya kemampuan *workstation* lebih kurang di bawah dari *file server*-nya dan mempunyai aplikasi lain di dalam media penyimpanannya selain aplikasi untuk jaringan. Kebanyakan LAN menggunakan media kabel untuk menghubungkan antara satu komputer dengan komputer lainnya. LAN merupakan jaringan komunikasi yang terbatas pada daerah yang kecil.

2.1.1. Media Transmisi

Kabel adalah alat penghubung untuk mengirim informasi dari satu komputer ke komputer yang lain. Ada beberapa macam tipe kabel yang umum digunakan pada LAN. Dalam beberapa kasus sebuah jaringan hanya menggunakan satu macam tipe kabel sedangkan pada jaringan lain menggunakan

kabel yang berbeda. Pemilihan kabel berdasarkan dengan topologi jaringan, protokol jaringan, dan ukurannya.

Adapun jenis-jenis kabel tersebut adalah :

1. Kabel *Unshielded Twisted Pair* (UTP)

Kabel *twisted pair* terdiri dari dua tipe yaitu *shielded* dan *unshielded*. *Unshielded twisted pair* (UTP) adalah yang paling populer dan umumnya merupakan pilihan yang terbaik untuk jaringan sederhana.

Kualitas kabel UTP berbeda dengan kabel telepon, kabel jenis ini mempunyai empat pasangan kabel di dalamnya. Setiap pasangan adalah jenis kabel kembar. Jenis konektor untuk kabel ini adalah konektor RJ-45.

2. Kabel *Shielded Twisted Pair* (STP)

Kekurangan kabel jenis ini adalah sangat sensitif terhadap sinyal radio dan listrik. Kabel seperti ini sangat baik digunakan dimana lingkungan pengaruh listrik kurang, serta biasanya digunakan pada jaringan yang menggunakan topologi *Token Ring*.

3. Kabel *Coaxial*

Kabel koaksial adalah kabel yang memiliki satu *Copper Conductor* pada bagian tengahnya. Sebuah lapisan plastik menutupi di antara konduktor dan lapisan pengaman serat besi. Lapisan serat besi tersebut membantu menutupi gangguan dari arus listrik, lalu lintas kendaraan atau mesin dan komputer.

Selain sangat sulit untuk konfigurasi, kabel ini pula sangat tidak tahan terhadap serangan dari sinyal-sinyal tertentu. Tetapi memiliki kelebihan karena dapat mendukung penggunaan kabel yang panjang di antara jaringan daripada

kabel *Twisted Pair*. Ada dua jenis tipe kabel ini yaitu kabel *thick coaxial* dan kabel *thin coaxial*.

Kabel *thin coaxial* disebut juga dengan 10Base2 (*thinnet*) dimana angka 2 menunjukan pada panjang maksimum untuk setiap segmen kabel tersebut yaitu 200 meter, namun kenyataannya hanya dapat menjangkau sampai 185 meter. Kabel ini sangat populer terutama pada penggunaan jaringan yang linear.

Kabel *thick coaxial* disebut juga dengan 10Base5 (*thicket*) dimana angka 5 menunjukan pada panjang maksimum untuk setiap segmen kabel tersebut yaitu 500 meter, dan satu kekurangan dari kabel jenis ini adalah tidak lentur dan sangat relatif sulit untuk mengkonfigurasinya. Tipe konektor untuk kabel jenis ini adalah konektor *Bayone-Neill-Concelman* (BNC).

4. Kabel *Fibre Optic*

Kabel *fibre optic* (serat optik) mempunyai kemampuan mentransmisikan sinyal melalui jarak yang relatif jauh daripada kabel *coaxial* ataupun kabel *twisted*, serta memiliki kecepatan yang baik. Kabel ini sangat baik digunakan untuk fasilitas konferensi radio atau layanan interaktif. 10BaseF merujuk pada spesifikasi untuk kabel *fibre optic* dengan membawa sinyal *Ethernet*.

Ketika akan melakukan konfigurasi kabel ini ada beberapa petunjuk yang bisa dilakukan, diantaranya :

a. Hendaknya menggunakan kabel yang relatif panjang untuk menghindari adanya kekurangan.
b. Ujilah tiap-tiap bagian dari jaringan yang akan dikonfigurasi, meskipun hal tersebut dalam kondisi baru.

c. Jika kabel melintasi lantai sebaiknya dilindungi dengan *protector* kabel.
d. Berikan label pada masing-masing ujungnya.
e. Gunakan pengikat kabel untuk menyatukan kabel-kabel jangan menggunakan isolasi atau plester.

2.1.2. Topologi LAN

Topologi secara fisik dari suatu jaringan lokal merujuk kepada konfigurasi kabel, komputer dan perangkat lainnya. Adapun jenis-jenis topologi LAN adalah sebagai berikut :

1. *Linear Bus* (Garis Lurus)

Topologi *linear Bus* terdiri dari satu jalur kabel utama dimana pada masing-masing ujungnya diberikan sebuah *terminator*. Semua *nodes* pada jaringan terkoneksi pada sebuah kabel utama (*backbone*). Jaringan-jaringan *Ethernet* dan *Local Talk* menggunakan topologi ini.

Kelebihan dari topologi *linear Bus* adalah :

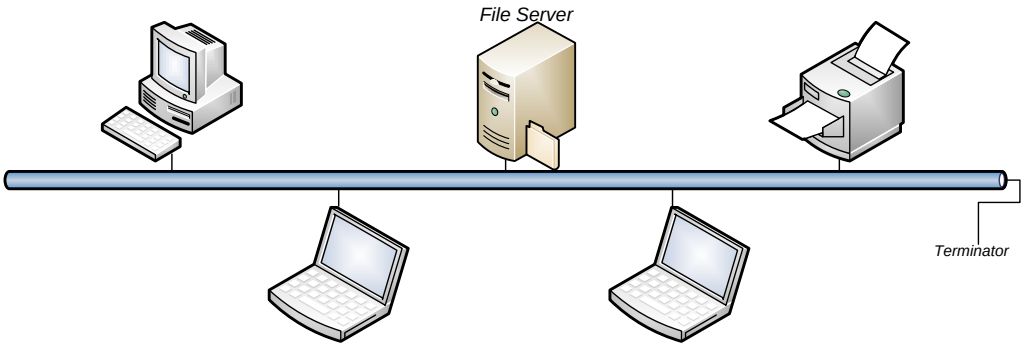
a. Mudah dalam mengkonfigurasi komputer atau perangkat lain ke dalam sebuah kabel utama.
b. Tidak terlalu banyak menggunakan kabel dibandingkan dengan topologi *star* / bintang.

Kekurangan dari topologi *linear Bus* adalah :

a. Seluruh jaringan akan mati jika ada kerusakan pada kabel utama.
b. Membutuhkan *terminator* pada kedua sisi kabel utamanya.

c. Sangat sulit mengidentifikasi permasalahan jika jaringan sedang *down* atau rusak.

d. Sangat tidak disarankan dipakai sebagai salah satu solusi pada penggunaan jaringan di gedung besar.



Gambar 2.1 Topologi *Linear Bus*

Pada gambar 2.1 memperlihatkan topologi jaringan *linear bus*, pada gambar tersebut kita dapat melihat *backbone* dan *terminator* dari *backbone*.

2. *Star* (Bintang)

Topologi model ini dirancang yang mana setiap *nodes* terkoneksi ke jaringan melewati sebuah *concentrator*.

Data yang dikirim ke jaringan lokal akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya, *concentrator* akan mengatur dan mengendalikan keseluruhan fungsi jaringan dan juga bertindak sebagai *repeater*. Konfigurasi jaringan model ini menggunakan kabel *Twisted Pair* dan dapat digunakan pada kabel *coaxial* atau kabel *fibre optic*.

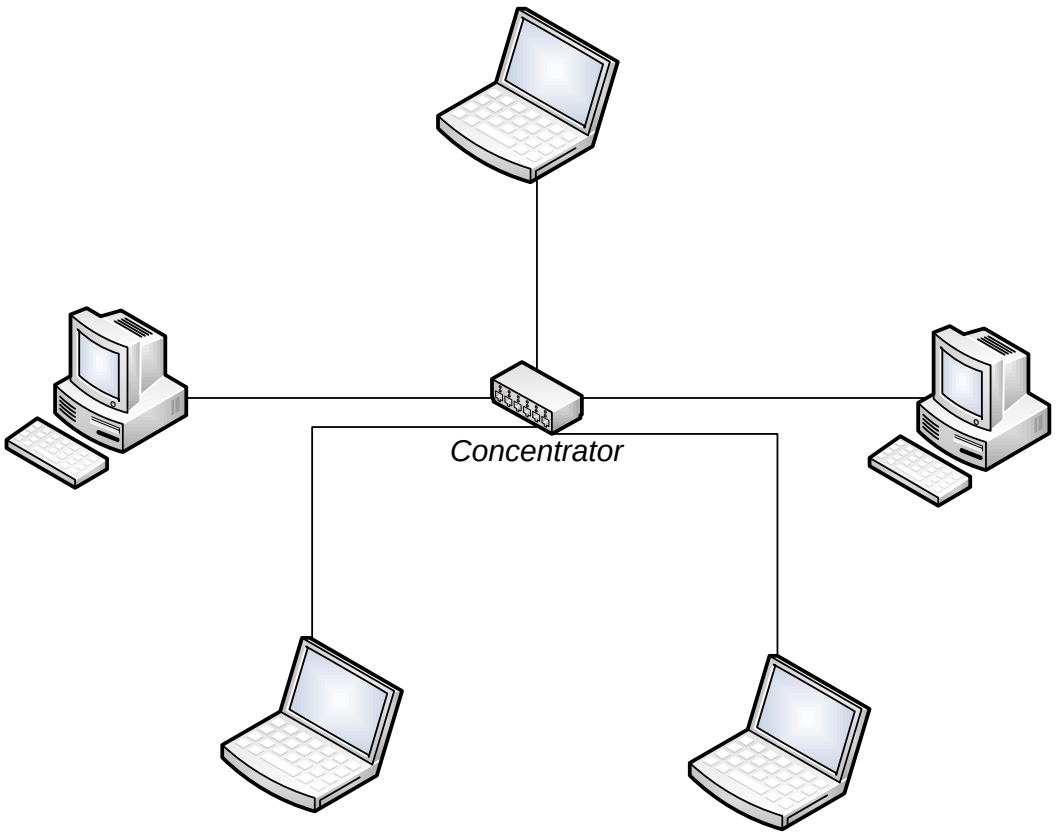
Kelebihan topologi *Star* (bintang) adalah :

a. Mudah dalam pemasangan dan pengkabelan.

b. Tidak mengakibatkan gangguan pada jaringan ketika akan memasang atau memindahkan perangkat jaringan lainnya.
c. Mudah untuk mendeteksi kesalahan dan memindahkan perangkat-perangkat lainnya.

Kekurangan dari topologi *Star* (bintang) adalah

a. Membutuhkan lebih banyak kabel daripada topologi *linear bus*.
b. Membutuhkan *concentrator* dan apabila *concentrator* rusak maka semua *node* yang terkoneksi tidak dapat terdeteksi.
c. Lebih mahal daripada topologi *linear bus* karena biaya untuk pembelian *concentrator*.

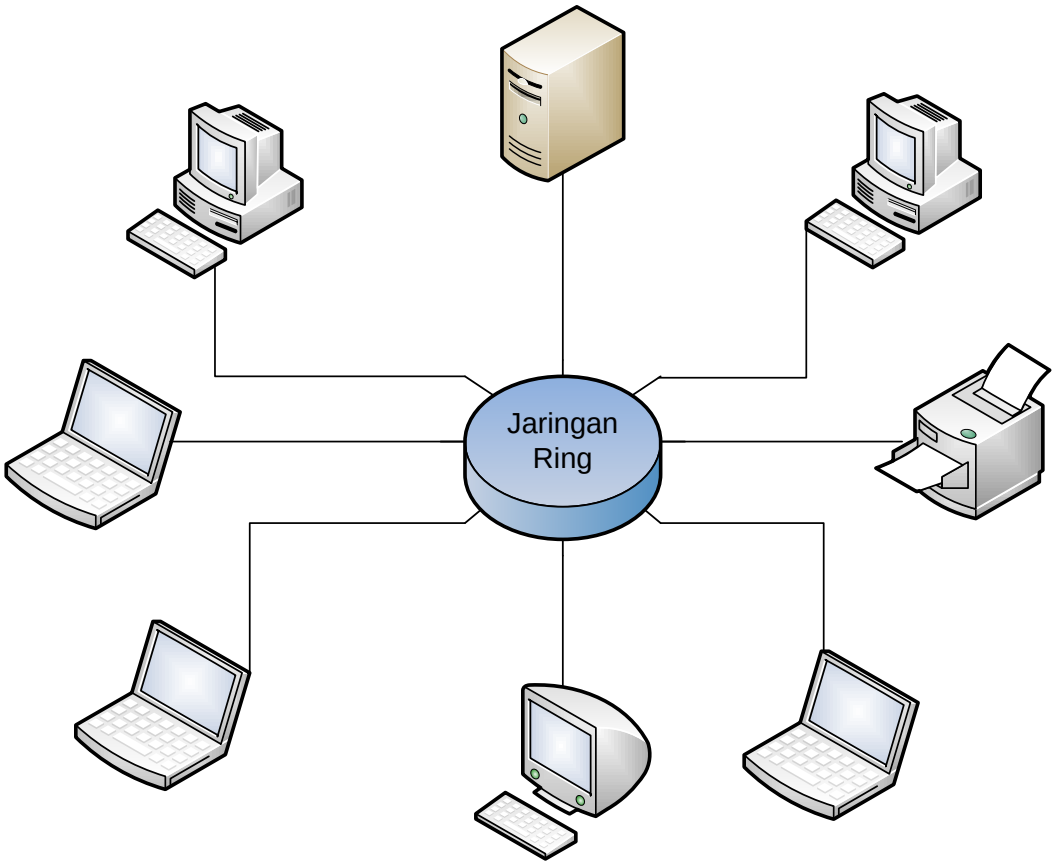


Gambar 2.2 Topologi *Star*

Pada gambar 2.2 memperlihatkan topologi jaringan *star*, pada gambar tersebut kita dapat melihat *concentrator* yang merupakan bagian paling vital dari topologi ini.

3. *Ring* (Cincin)

Topologi *Ring* (cincin) menggunakan teknik konfigurasi yang sama dengan topologi *star* tetapi pada topologi ini terlihat bahwa jalur media transmisi menyerupai suatu lingkaran tertutup menyerupai cincin sehingga diberi nama topologi bintang dalam lingkaran atau *star-wired ring*.



Gambar 2.3 Topologi *Ring*

Pada gambar 2.3 di atas terlihat bahwa *concentrator* dari topologi *ring* berbentuk lingkaran tetapi sebenarnya yang berbentuk lingkaran itu adalah kabel untuk menghubungkan dari kartu jaringan ke *concentrator*

4. *Tree* (Pohon)

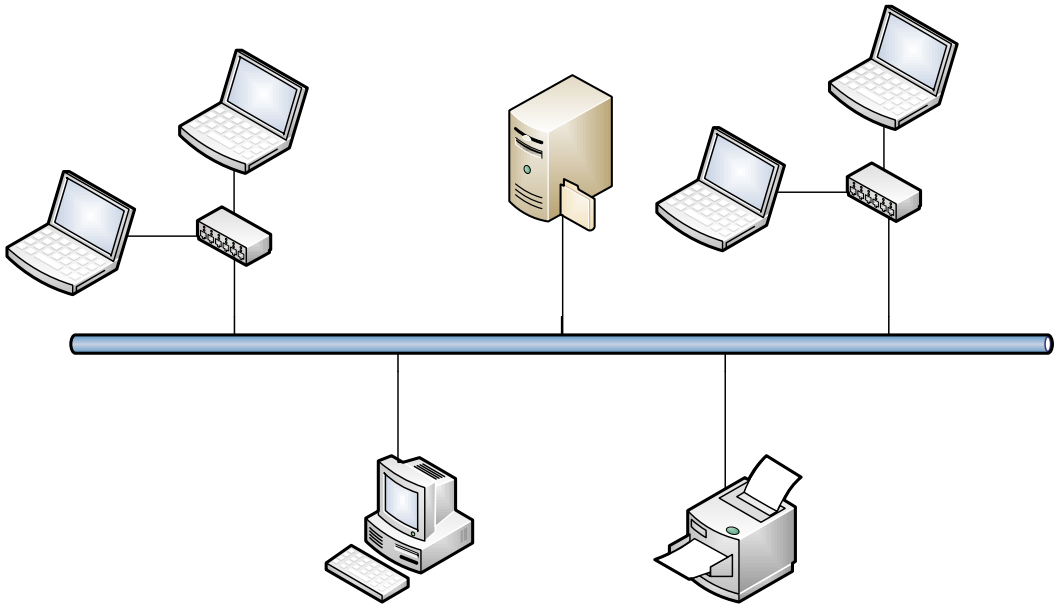
Topologi model ini merupakan perpaduan antara topologi *linear bus* dan *star*, yang mana terdiri dari kelompok-kelompok *workstation* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *Linear Bus*. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada dan memungkinkan untuk mengkonfigurasi jaringan sesuai dengan kebutuhan.

Kelebihan dari topologi *Tree* adalah

a. Proses konfigurasi jaringan dilakukan dari titik ke titik pada masing-masing segmen.
b. Didukung oleh banyak perangkat keras dan perangkat lunak

Kekurangan dari topologi *Tree* adalah

a. Keseluruhan panjang kabel pada tiap-tiap segmen dibatasi oleh tipe kabel yang digunakan.
b. Jika jaringan utama rusak maka keseluruhan segmen ikut rusak juga.
c. Sangat relatif sulit untuk dikonfigurasi dan proses pengkabelannya dibandingkan dengan topologi jaringan yang lain.



Gambar 2.4 Topologi *Tree*

Pada gambar 2.4 di atas terlihat bahwa topologi *tree* merupakan gabungan dari beberapa topologi.

2.1.3. Model Hubungan Pada LAN

1. *Peer-to-peer*

Model hubungan *peer to peer* memungkinkan *user* membagi sumber dayanya yang ada di komputernya baik berupa *file*, layanan *printer* dan lain-lain serta mengakses sumber daya yang terdapat pada komputer lain. Namun model ini tidak mempunyai sebuah *file server* atau sumber daya yang terpusat. Pada model ini seluruh komputer adalah sama, yang mana mempunyai kemampuan yang sama untuk memakai sumber daya yang tersedia di dalam jaringan. Model jaringan ini didesain untuk jaringan berskala kecil dan menengah.

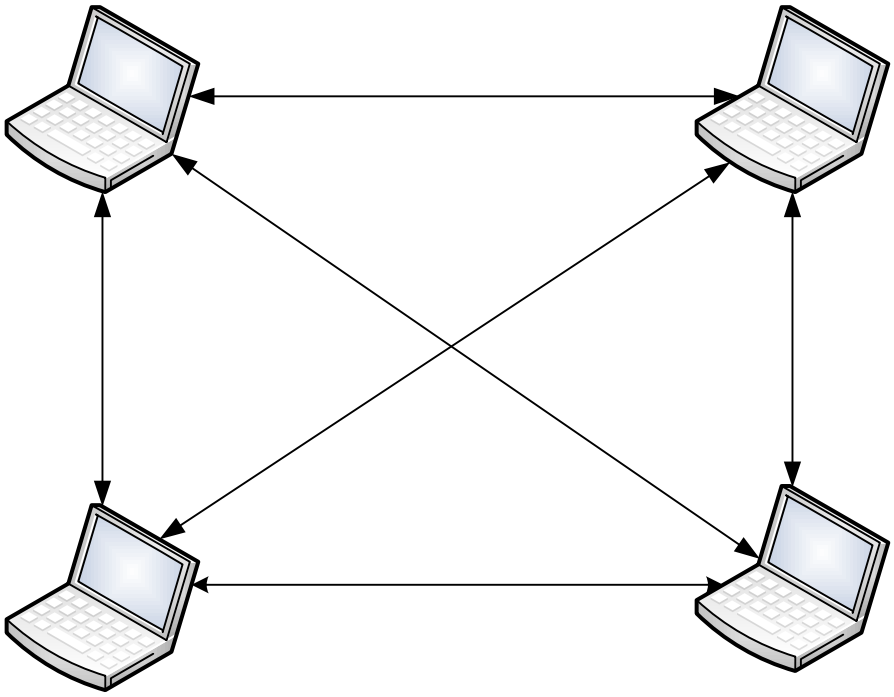
Kelebihan model hubungan *peer to peer* adalah :

a. Tidak terlalu mahal, karena tidak membutuhkan *dedicated file server*.

b. Mudah dalam konfigurasi programnya, hanya tinggal mengatur untuk operasi model hubungan *peer to peer*.

Kekurangan model hubungan *peer to peer* adalah :

a. Tidak terpusat, terutama untuk penyimpanan data dan aplikasi.
b. Tidak aman, karena tidak menyediakan fasilitas untuk keperluan itu.



Gambar 2.5 Model Hubungan *Peer ro Peer*

Pada gambar 2.5 memperlihatkan model hubungan *peer to peer* pada suatu jaringan.

2. *Client / Server*

Model hubungan *Client / Server* memungkinkan jaringan untuk mensentralisasi fungsi dan aplikasi kepada satu atau dua *dedicated file server*. Sebuah *file server* menjadi jantung dari keseluruhan sistem, memungkinkan untuk mengakses sumber daya dan menyediakan keamanan. *Workstation* yang berdiri

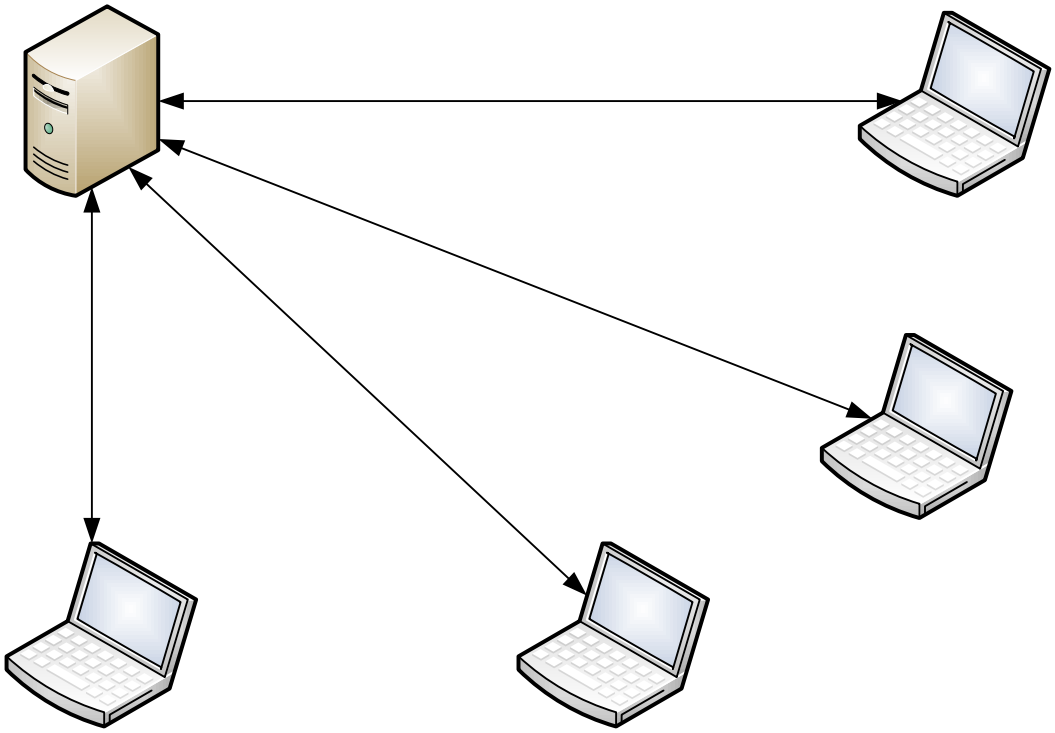
sendiri dapat mengambil sumber daya yang ada pada *file server*. Model hubungan ini menyediakan mekanisme untuk mengintegrasikan seluruh seluruh komponen yang ada di jaringan dan memungkinkan banyak pengguna secara bersama-sama memakai sumber daya pada *file server*.

Kelebihan model hubungan *Client Server* adalah :

a. Terpusat (sumber daya dan keamanan data terkontrol melalui server)
b. Skalabilitas
c. Fleksibel
d. Teknologi baru dengan mudah terintegrasi ke dalam sistem
e. Keseluruhan komponen (*client / network / server*) dapat bekerja bersama

Kekurangan model hubungan *Client Server* adalah :

a. Mahal
b. Membutuhkan investasi untuk *dedicated file server*
c. Perbaikan (Jaringan besar membutuhkan seorang staff untuk mengatur agar sistem berjalan secara efisien)
d. Berketergantungan
e. Ketika *server* jatuh, mengakibatkan keseluruhan operasi pada *network* akan jatuh pula



Gambar 2.6 Model Hubungan *Client Server*

Pada gambar 2.6 memperlihatkan model hubungan *client server* pada suatu jaringan.

2.2. Jaringan Internet

Internet adalah suatu jaringan komputer global yang terbentuk dari jaringan-jaringan komputer lokal dan regional yang memungkinkan komunikasi data antar komputer yang terhubung ke jaringan tersebut. Dimulai dari suatu proyek yang dinamakan ARPANET atau *Advanced Research Project Agency Network* beberapa universitas diantaranya UCLA, Stanford, UC Santa Barbara dan University of Utah diminta bantuan dalam mengerjakan proyek ini dan awalnya telah berhasil menghubungkan empat komputer di lokasi universitas yang berbeda tersebut. Perkembangan ARPANET ini cukup pesat jika dilihat

perkembangan komputer pada saat itu. Sebagai gambaran pada tahun 1977 ARPANET telah menghubungkan lebih dari 100 *mainframe* komputer dan pada saat ini terdapat sekitar 4 juta *host* jaringan yang terhubung pada jaringan ini.

Jumlah sebenarnya dari komputer yang terhubung tidak dapat diketahui dengan pasti karena perkembangan jumlah komputer yang terhubung dengan suatu jaringan semakin lama semakin besar. Karena perkembangannya sangat pesat, jaringan komputer ini tidak dapat lagi disebut sebagai ARPANET karena semakin banyak konputer dan jaringan-jaringan regional yang terhubung. Konsep ini yang kemudian berkembang dan dikenal sebagai konsep *internetworking* (jaringan antar jaringan). Oleh karena itu istilah internet menjadi semakin populer, dan orang menyebut jaringan besar komputer tersebut dengan istilah internet.

2.2.1. Protokol

Agar komputer-komputer dalam suatu jaringan dapat berkomunikasi, maka dibutuhkan suatu protokol atau suatu aturan standar komunikasi baik antar komputer maupun antara jaringan komputer. Saat ini terdapat banyak jenis protokol jaringan komputer seperti IPX/SPX yang biasa digunakan oleh NOVELL Netware, NETBIOS/NETBEUI biasa digunakan oleh jaringan Microsoft LAN Manager ataupun Microsoft Window Networking, AppleTalk yang biasa digunakan oleh jaringan komputer Apple Macintosh dan sebagainya.

Ada satu protokol yang dikembangkan oleh DARPA (*Defense Advanced Research Projects Agency*) dalam pengembangan dari ARPANET dan juga digunakan oleh jaringan komputer berbasis sistem operasi UNIX yaitu protokol

TCP/IP (*Transmission Control Protocol / Internet Protocol*). Protokol TCP/IP ini menjadi standar protokol yang digunakan pada jaringan Internet karena TCP/IP dikembangkan untuk dapat diterapkan di hampir segala jenis *platform* komputer yang biasa dikenal dengan konsep *open system*.

2.2.2. Standar TCP/IP

Internet dapat terbentuk karena sekumpulan besar jaringan komputer memiliki kesepakatan untuk berbicara dalam bahasa yang sama. Kesepakatan ini semata-mata merupakan kesepakatan yang bersifat teknis karena tidak ada suatu badanpun di dunia ini yang berhak mengatur jalannya internet secara keseluruhan. Yang dapat diatur dalam internet adalah penggunaan protokolnya. TCP/IP dikembangkan sebelum model OSI ada. Namun demikian lapisan-lapisan pada TCP/IP tidaklah cocok seluruhnya dengan lapisan-lapisan OSI. Protokol TCP/IP hanya dibuat atas lima lapisan saja yaitu *physical, data link, network, transport* dan *application*.

Hanya lapisan aplikasi pada TCP/IP mencakupi tiga lapisan OSI teratas. Khusus layer keempat, protokol TCP/IP mendefinisikan 2 buah protokol yakni *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP). Sementara itu pada lapisan ketiga TCP/IP mendefinisikan *Internetworking* (IP), namun ada beberapa protokol lain yang mendukung pergerakan data pada lapisan ini.

2.2.3. Kelas IP *Address*

Untuk mempermudah proses pembagiannya, IP *address* dikelompokkan dalam kelas-kelas. Dasar pertimbangan pembagian IP *Address* ke dalam kelas-kelas adalah untuk memudahkan pendistribusian pendaftaran IP *address*.

Dengan memberikan sebuah ruang nomor jaringan (beberapa blok IP *address*) kepada ISP (*Internet Service Provider*) di suatu area diasumsikan penanganan komunitas lokal tersebut akan lebih baik dibandingkan dengan jika setiap pemakai individual harus meminta IP *address* ke otoritas pusat yaitu *Internet Assigned Numbers Authority* (IANA). IP *address* ini dikelompokan dalam lima kelas yaitu kelas A, kelas B, kelas C, kelas D dan kelas E. perbedaan pada tiap kelas tersebut adalah pada ukuran dan jumlahnya. IP kelas A dipakai oleh sedikit jaringan namun jaringan ini memiliki anggota yang besar. Kelas C dipakai oleh banyak jaringan, namun anggota masing-masing jaringan sedikit. Kelas D dan E juga didefinisikan tetapi tidak digunakan dalam pengunaan normal. Kelas D diperuntukkan bagi jaringan *multicast* dan kelas E untuk keperluan eksperimental.

2.2.3.1. *Network* ID dan *host* ID

Pembagian kelas-kelas IP *address* didasarkan pada dua hal yaitu *nerwork ID* dan *host ID* dari suatu IP *address*. *Network* ID ialah bagian dari IP *address* yang digunakan untuk menunjukkan jaringan tempat komputer ini berada. Sedangkan *host* ID ialah bagian dari IP *address* yang digunakan untuk menunjukkan *workstation*, *server*, *router*, dan semua *host* TCP/IP lainnya dalam

jaringan tersebut. Dalam suatu jaringan *host* ID ini harus unik (tidak boleh ada yang sama).

Kelas A

Karakteristik :

Format : 0nnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh

Bit Pertama : 0

Panjang Net ID : 8 bit

Panjang Host ID : 24 bit

Byte Pertama : 0 – 127

Jumlah : 126 kelas A (0 dan 127 dicadangkan)

Range IP : 1.xxx.xxx.xxx sampai 126.xxx.xxx.xxx

Jumlah IP : 16.777.214 IP *address* pada tiap kelas A

Bit pertama dari IP *address* kelas A selalu diset 0 sehingga byte terdepan dari IP *address* kelas A bernilai antara angka 0 hingga 127. *Network* ID ialah delapan bit pertama, sedangkan *host* ID ialah 24 bit berikutnya. Dengan demikian cara membaca IP *address* kelas A dengan IP 113.46.5.6 ialah :

Network ID = 113

Host ID = 46.5.6

Sehingga IP *address* di atas berarti *host* nomor 46.5.6 pada *network* nomor 113.

Kelas B

Karakteristik :

Format : 10nnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh

Bit Pertama : 10

Panjang Net ID : 16 bit

Panjang Host ID : 16 bit

Byte Pertama : 128 – 191

Jumlah : 16.184 kelas B

Range IP : 128.0.xxx.xxx sampai 191.255.xxx.xxx

Jumlah IP : 65.532 IP *address* pada tiap kelas B

Bit pertama dari IP *address* kelas B selalu diset 10 sehingga byte terdepan dari IP *address* kelas B bernilai antara angka 128 hingga 191. *Network* ID ialah enam belas bit pertama, sedangkan *host* ID ialah enam belas bit berikutnya. Dengan demikian cara membaca IP *address* kelas B dengan IP 132.92.121.6 ialah

Network ID = 132.92

Host ID = 121.6

Sehingga IP *address* di atas berarti *host* nomor 121.6 pada *network* nomor 132.92.

Kelas C

Karakteristik :

Format : 110nnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh

Bit Pertama : 110

Panjang Net ID : 24 bit

Panjang Host ID : 8 bit

Byte Pertama : 192 – 223

Jumlah : 2.097.152 kelas C

Range IP : 192.0.0.xxx sampai 223.255.255.xxx

Jumlah IP : 254 IP *address* pada tiap kelas B

IP *address* kelas C awalnya digunakan untuk jaringan berukuran kecil (misalnya LAN). Tiga bit pertama dari IP *address* kelas C selalu berisi 111 bersama dengan 21 bit berikutnya, angka ini membentuk *network* ID 24 bit. *Host* ID ialah 8 bit terakhir. Dengan konfigurasi ini bisa dibentuk sekitar dua juta *network* dengan masing-masing *network* memiliki 256 IP *address*.

Kelas D

Karakteristik :

Format : 1110nnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh

4 Bit Pertama : 1110

Bit Multicast : 28 bit

Byte Inisial : 224-247

Deskripsi : Kelas D adalah ruang alamat *multicast* (RFC 1112)

IP *address* kelas D digunakan untuk keperluan IP *multicasting*. 4 bit pertama IP *address* kelas D diset 1110. Bit-bit berikutnya diatur sesuai keperluan *multicast group* yang menggunakan IP *address* ini. Dalam *multicasting* tidak dikenal *network* bit dan *host* bit.

Kelas E

Karakteristik :

Format : 1111rrrr.rrrrrrrr.rrrrrrrr.rrrrrrrr

4 Bit Pertama : 1111

Bit Multicast : 28 bit

Byte Inisial : 248-255

Kelas E adalah ruang alamat yang dicadangkan untuk keperluan eksperimental. IP *address* kelas E tidak digunakan untuk umum. Empat bit pertama IP *address* diset 1111.

2.2.4. Metode pengalamatan di Internet

Tiap-tiap komputer yang terhubung dengan jaringan internet dapat saling berkomunikasi. Untuk itu diperlukan tata cara pengalamatan pada jaringan internet yang sistemnya hampir sama dengan tata cara pengalamatan nomor telepon dimana setiap telepon mempunyai nomor telepon yang unik.

Dengan konsep dari protokol TCP/IP, setiap komputer yang terhubung pada jaringan TCP/IP secara teori harus mempunyai suatu alamat yang unik. Alamat ini dikenal sebagai *Internet Protocol Number* (IP *Number* / IP *Address*) sebesar 32 bit dan direpresentasikan dalam bentuk desimal dibagi menjadi 4 bagian dipisahkan dengan titik.

Pada saat ini ada konsep-konsep baru yang memungkinkan satu IP dipakai oleh lebih dari satu komputer. Satu IP *Address* mempunyai suatu nama untuk dapat mempermudah mengidentifikasikan suatu komputer di internet atau yang biasa disebut *host*. *Host* ini dapat memberikan fasilitas layanan kepada jaringan yang terhubung kepadanya. Satu nama *host* dapat mempunyai lebih dari satu nama alias.

2.2.4.1. *Domain*

Tata cara penamaan suatu *host* dikenal dengan istilah *domain* yang digunakan untuk menentukan posisi hirarki *host* dari jaringan ini. Di bawah nama

domain dalam hirarki ini dimungkinkan adanya nama *subdomain*. Penentuan IP *address* dan nama *domain* tidak dapat sembarangan. Permohonan harus diajukan kepada *Internet Network Information Center* (InterNIC). Badan ini bertugas mengelola pemakaian alamat IP dan nama *domain*. Berikut ini adalah beberapa nama *domain* yang ada di internet.

Com , co : untuk badan komersil

Contoh : dotnetnuke.com, yahoo.co.id

Edu , ac : untuk lembaga pendidikan

Contoh : unikom.ac.id , ucla.edu

Gov , go : untuk lembaga pemerintahan

Contoh : esdm.go.id , fbi.gov

Net : untuk gateway jaringan, ISP

Contoh : telkom.net , indosat.net.id

Mil : untuk militer

Contoh : af.mil

Org , or : untuk organisasi

Contoh : wikipedia.org , asean.or.id

Pada perkembangan saat ini internet sudah mencakup banyak negara, maka agar mempermudah dalam mengidentifikasi lokasi *host* internet maka dibuat hirarki nama *domain* negara. Berikut ini adalah beberapa nama *domain* negara :

Au : Australia

Ca : Canada

Fr : France

Id : Indonesia

My : Malaysia

Sg : Singapura

2.2.4.2. Identitas di Internet

Sesuai dengan tata cara penamaan *domain* di internet, maka tata cara penamaan *host* di internet sesuai dengan posisi pada hirarki *domain* dan *subdomain* dari *host* tersebut. Bentuk penamaan *host* di internet adalah

host.subdomain.domain.

Contoh : www.unikom.ac.id

nama *host* pada contoh adalah : www

nama *subdomain* pada contoh adalah : unikom

dan nama *domain* adalah : ac.id

Pengalamatan nama pemakai atau *user* pada *host* di internet atau yang biasa disebut alamat email adalah

nama@host.subdomain.domain

contoh : wrdc@sda.kimpraswil.go.id

2.2.4.3. *Bandwidth*

Dalam sistem komunikasi data komputer dikenal istilah *bandwidth* atau kecepatan transmisi data dalan satuan bit perdetik. Semakin cepat *bandwidth* maka semakin cepat transmisi datanya.

Semakin besar *bandwidth* maka semakin cepat transmisi data yang dapat dilakukan antara kedua lokasi komputer tersebut. Semakin besar *bandwidth* maka semakin besar biaya yang harus dikeluarkan. Karena tarif sewa *bandwidth* tidak murah. Sebagai gambaran paling sederhana ialah jika kita menggunakan saluran telepon sebagai media komunikasi maka *bandwidth* modem yang digunakan tarif biaya sewanya sama dengan tarif menggunakan telepon lokal selama waktu yang digunakan untuk melakukan koneksi.

2.3. *Streaming* video

Usaha untuk menampilkan media pada komputer telah dilakukan sejak pertengahan abad dua puluh. Bagaimanapun juga sedikit proses yang telah dilakukan dalam beberapa dekade diakibatkan biaya yang tinggi dan keterbatasan kemampuan *hardware* komputer.

Eksperimen akademik pada tahun 1970 berhasil menemukan konsep dasar dan kemungkinan dalam melakukan *streaming* media di komputer. Pada akhir tahun 1980 komputer sudah cukup mampu untuk menampilkan berbagai media. Permasalahan utama teknik *streaming* adalah

a. Tenaga CPU yang cukup dan *bandwidth* bus untuk mendukung *data rate* yang diperlukan
b. Menciptakan jalur *interupt latency* rendah pada sistem operasi untuk menghindari *buffer underrun*

Bagaimanapun juga pada saat itu jaringan komputer masih terbatas dan media biasanya dikirimkan secara *non-streaming* seperti CDROM.

Pada akhir tahun 1990 terlihat berbagai fenomena berikut :

a. *Bandwidth* jaringan yang lebih besar
b. Meningkatnya akses ke jaringan seperti internet
c. Penggunaan protokol standar dan format seperti TCP/IP, HTTP, dan HTML
d. Komersialisasi di internet

Keuntungan dari jaringan komputer ini digabungkan dengan komputer rumah yang bertenaga sistem operasi yang moderen membuat *streaming* media menjadi mudah. Secara umum konten multimedia sangat besar, sehingga media penyimpanan dan biaya transmisi masih sangat signifikan. Untuk mengatasinya maka media biasanya dikompresi baik untuk penyimanan dan *streaming*.

Suatu media *stream* dapat *on demand* atau langsung. Media *Streaming* yang dilakukan secara *on demand* disimpan dalam suatu server untuk waktu yang lama dan dapat ditransmisikan sesuai dengan permintaan *user*. *Live stream* hanya bisa dilihat pada waktu tertentu seperti *stream* video dari acara olahraga secara langsung.

Ukuran media penyimpanan dapat dihitung dari *streaming bandwidth* dan durasi dari media dapat dirumuskan sebagai berikut

Ukuran media penyimpanan (MB) = (durasi (detik) * bit rate (kbit/s) / 8.388,608)

Contohnya video dengan durasi 1 jam yang diencoding pada 300 kbit/s maka ukuran media penyimpanannya adalah

(3600 s * 300 kbit/s) / (8*1024) maka penyimpanannya sekitar 130 MB

Jika file disimpan di server untuk *on demand streaming* dan *stream* ini dilihat oleh 1000 orang dalam waktu yang bersamaan menggunakan protokol *unicast* maka diperlukan

300 kbit/s * 1000 = 300.000 kbit/s = 300Mbit/s *bandwidth*

Jumlah tersebut setara dengan 125 GB per jam. Tentu saja, menggunakan protokol *multicast* server mengirim hanya satu *stream* ke seluruh *user*. Karena itu sebuah *stream* hanya menggunakan 300 kbit/s dari *bandwidth* yang tersedia.

Mendesain suatu protokol jaringan untuk mendukung media *streaming* meningkatkan banyak permasalahan seperti :

a. Protokol datagram seperti *User Datagram Protocol* (UDP), mengirimkan media *stream* dalam seri paket kecil. Metode ini sederhana dan efisien, bagaimanapun juga tidak ada mekanisme dari suatu protokol untuk menjamin pengiriman. Mekanisme ini dimiliki aplikasi penerima untuk mendeteksi *kehilangan* atau *korupnya* dan memulihkan data menggunakan teknik koreksi error. Jika data hilang maka dapat menyebabkan *dropout*.
b. *Real-time Streaming Protocol* (RTSP), *Real-time Transport Protocol* (RTP), dan *Real-time Transport Control Protocol* (RTCP) didesain khusus untuk *streaming* media melalui jaringan.
c. Protokol yang diandalkan seperti *Transmission Control Protocol* (TCP), menjamin pengiriman yang sempurna untuk tiap bit dari media *stream*. Bagaimanapun juga metode ini menyelesaikan pengiriman dengan suatu sistem *timeouts* dan *retries*, yang mana membuatnya menjadi lebih kompleks untuk diimplementasikan. Ini juga berarti ketika terjadi kehilangan data pada

jaringan, media *stream* tertampung ketika pengendalian protokol mendeteksi kehilangan dan mentransmisikan kembali data yang hilang. *Client* dapat meminimalisasi efek ini dengan *buffering* data untuk ditampilkan.

d. Protokol *unicast* mengirimkan salinan terpisah media *stream* dari server ke tiap *client*. Dalam masa kesulitan dalam teknik implementasi protokol ini merupakan protokol yang paling sederhana. Dengan sederhananya protokol maka memungkinkan terjadinya duplikasi data yang dikirmkan di jaringan.
e. Protokol *multicast* dikembangkan dalam usaha untuk mengurangi duplikasi yang disebabkan oleh protokol *unicast*. Protokol ini hanya mengirim satu salinan media *stream* melalui koneksi jaringan yang diberikan sepanjang jalur antara dua *router* jaringan. Kebanyakan protokol ini membutuhkan hardware *routing* khusus yang mampu mengirimkan *stream*. *Multicast* merupakan suatu cara koneksi yang mana sangat mendekati cerminan fungsional melalui televisi yang kehilangan kemampuan melihat acara secara *on-demand*. Beberapa kemampuan yang hilang ini termasuk membalikan dan mempercepat suatu file media.
f. IP *Multicast*, protokol yang paling menonjol dari protokol *multicast* harus mengimplementasikan di setiap *node* antara *server* dan *client* termasuk *router* jaringan. Kebanyakan *router* di internet bagaimanapun juga tidak mendukung IP *multicast* dan banyak *firewall* membloknya. IP *multicast* sangat berguna untuk organisasi yang menjalankan jaringan mereka sendiri seperti universitas dan perusahaan. Dikarenakan mereka membeli *router* sendiri dan menjalankan

jaringannya sendiri, mereka dapat menentukan apakah harga dan usaha dari dukungan IP *multicast* sebanding dengan hasil penghematan *bandwidth*.

g. Protokol *Peer-to-peer* (P2P) mengatur media yang akan dikirimkan *client* yang sudah memiliki media ke *client* yang belum memilikinya. Protokol ini mencegah server dan koneksi jaringan menjadi bottleneck. Bagaimanapun juga protokol ini meningkatkan teknik, performansi, kualitas, bisnis, dan permasalahan legal.

Permasalahan untuk *streaming* media adalah beberapa *firewall* memblok protokol UDP dasar memberikan keamanan tambahan untuk pemiliknya. Pemblokan ini dilakukan karena UDP *stateless* dan membuat sulit bagi *firewall* untuk menentukan apakah perlu mengijinkan koneksi.

Penyebaran secara meluas *streaming* media meningkatkan permasalahan *scaling* dan *quality of service*. Pengetesan layanan penyebaran merupakan masalah yang sangat signifikan. *Vendor* menawarkan peralatan untuk mengetes layanan *streaming* melalui sejumlah pengujian dari pengujian umum termasuk *Scalability, Quality of Service*, *Quality of Experience*, dan penyesuaian jaringan.

2.4. *Video chat*

Video chat merupakan suatu teknologi telekomunikasi yang interaktif yang mana mengijinkan dua atau lebih lokasi untuk berinteraksi secara dua arah melalui transmisi video dan audio secara serempak. Video *chat* pertama kali didemonstrasikan pada tahun 1968.



Gambar 2.7 Gambar video *chat* yang pertama kali

Pada gambar 2.7 memperlihatkan tampilan dari aplikasi video *chat* yang pertama kali dibuat.

Komponen yang diperlukan untuk melakukan *video chat* ini adalah

a. Masukan video : kamera video atau webcam
b. Keluaran video : monitor komputer, televisi, atau proyektor
c. Masukan audio : *microphone*
d. Keluaran audio : biasanya *loudspeaker* yang diasosiasikan dengan alat tampilan atau telepon
e. Transfer data : telepon jaringan analog ataupun digital, LAN atau Internet.

Teknologi inti yang digunakan pada sistem *video chat* adalah kompresi digital video dan audio secara *realtime*. *Hardware* atau *software* yang melakukan kompresi disebut *codec* (*coder*/*decoder*). Kompresi dapat mencapai 1 : 500, hasil *stream* digital dari 1 detik dan 0 detik dibagi ke dalam paket berlabel yang mana akan ditransmisikan melalui jaringan digital.

Pada dasarnya terdapat dua macam sistem video *chat*. Sistem tersebut adalah *Dedicated system* dan *Desktop system*. Pada *Dedicated system* memiliki semua komponen yang diperlukan yang disatukan dalam sebuah alat, biasanya

suatu alat dengan video kamera berkualitas tinggi yang dapat dikendalikan dari jarak jauh. Kamera ini dapat dikendalikan dari jarak jauh untuk digerakan ke kiri dan kanan, atas dan bawah dan dapat melakukan *zoom*. *Microphone* sudah terkoneksi dengan alat tersebut seperti monitor TV dengan *loudspeaker* atau video proyektor. *Desktop system* adalah alat tambahan yang digunakan pada PC yang dapat mengubah PC menjadi alat video *chat*. Kebanyakan sistem *desktop* bekerja dengan standar H.323.

Fitur fundamental dari sistem video *chat* profesional adalah *acoustic echo cancellation* (AEC). AEC adalah suatu algoritma yang dapat mendeteksi kapan suara masuk kembali ke audio input pada *codec* video *chat*. Jika tidak dicek maka dapat menyebabkan beberapa masalah termasuk lawan bicara ketika melakukan video *chat* mendengar suaranya sendiri kembali (biasanya disebabkan *delay* yang sangat besar), gema yang kuat, merender jalur audio sia-sia karena sulit untuk dimengerti, deru tercipta oleh umpan balik. *Echo cancellation* merupakan tugas intensif prosessor yang biasanya dikerjakan melalui serangkaian *delay* suara.

Penggunaan modem audio dalam jalur transmisi memungkinkan penggunaan POTS atau *Plain Old Telephone System* pada aplikasi dengan kecepatan rendah seperti *videotelephony* karena modem tersebut mengkonversi gelombang digital dari atau ke gelombang analog pada jangkauan spektrum audio.

Video chat pada saat ini telah banyak berkembang. Hal ini dapat terlihat dengan adanya fasilitas *video chat* yang terdapat pada aplikasi *chating* seperti NetMeeting, MSN Messenger, Yahoo Messenger, SightSpeed, Skype dan lainnya.

Permasalahan utama dalam *video chat* ini adalah meminimalisasi waktu *delay* dalam penerimaan transmisi audio dan video. Ketika melakukan video *chat* sering kali suara dan video tidak diterima secara bersamaan seperti halnya dalam melakukan komunikasi langsung. Hal ini diakibatkan semakin banyaknya pengguna internet sehingga menyebabkan kecilnya *bandwidth* transmisi.

2.5. *Visual basic*

Visual basic merupakan generasi ketiga bahasa pemrograman *event driven* dan berasosiasi dengan pengembangan lingkungan dari microsoft untuk model pemrograman COM. *Visual basic* berasal dari BASIC memungkinkan *Rapid Application Development* (RAD) dengan aplikasi berbasis *Graphical User Interface* (GUI), akses ke database menggunakan DAO, RDO, atau ADO dan membuat *ActiveX* kontrol dan objek.

Seorang programmer dapat meletakan komponen yang disediakan *visual basic* pada aplikasinya. Program yang ditulis dengan *visual basic* dapat juga menggunakan windows API tetapi memerlukan deklarasi fungsi eksternal. Dalam bisnis programming *visual basic* memiliki banyak pengguna. Pada survey yang dilakukan pada tahun 2005, enam puluh dua persen dari pengembang melaporkan menggunakan *visual basic*. *Visual basic* berkompetisi dengan C++, C#, dan java untuk dominasi di dunia bisnis.

Visual basic didesain agar lebih mudah digunakan dan dipelajari. Bahasanya tidak hanya memungkinkan *programmer* untuk menciptakan aplikasi GUI yang sederhana, tetapi juga dapat mengembangkan aplikasi yang kompleks.

Programming pada VB merupakan suatu kombinasi dari pengaturan secara visual komponen atau kontrol pada sebuah *form*, menspesifikasi atribut dan aksi komponen tersebut, dan menuliskan tambahan baris kode agar lebih fungsional. Program sederhana dapat dibuat tanpa *programmer* harus menulis banyak baris *code*. Masalah performansi dialami pada versi sebelumnya tetapi dengan komputer yang lebih cepat dan kompilasi *native code* telah dapat mengurangi masalah tersebut.

Form dibuat dengan teknik *drag* dan *drop*. Sebuah tool digunakan untuk menempatkan kontrol (seperti textbox,tombol dan lainnya) pada *form* (*window*). Kontrol memiliki atribut *event handler* yang diasosiasikan dengannya. Nilai awal diberikan ketika kontrol tercipta, tetapi nilai tersebut dapat dirubah oleh *programmer*. Banyak nilai atribut yang bisa dirubah selama *runtime* berdasarkan kepada aksi *user* atau perubahan lingkungan, menjadikan aplikasi yang dinamis. Contohnya suatu kode dapat dimasukan di *event handler resize form* untuk memposisikan suatu kontrol agar tetap berada di tengah-tengah *form*. Dengan memasukan kode ke *event handler* untuk *keyprees* yang terdapat pada *textbox*, program dapat secara otomatis menterjemahkan *case* dari teks yang dimasukan atau bahkan mencegah karakter tertentu untuk dimasukan ke *textbox*.

Visual basic dapat menciptakan file *executeables* (*.EXE), kontrol *ActiveX*, file DLL, tetapi penggunaan utamanya untuk mengembangkan aplikasi *windows* dan untuk *interface* sistem *web database*. Kotak dialog dengan sedikit fungsi (seperti tidak ada kontrol minimize/maximize) dapat digunakan untuk menyediakan kemampuan *pop-up*. Kontrol menyediakan fungsi dasar dari suatu

aplikasi, ketika *programmer* dapat memasukan logika tambahan pada suatu *event handlers*. Contohnya suatu *combo box* (*Drop down list*) akan secara otomatis menampilkan *list* data dan memungkinkan user untuk memilih setiap elemennya. *Event handlers* dipanggil ketika suatu *item* dipilih yang mana akan mengeksekusi kode tambahan yang diciptakan *programmer* untuk melakukan suatu aksi berdasarkan elemen yang dipilih.

2.6. *ActiveX*

ActiveX merupakan suatu *Component Object Model* (COM) yang dikembangkan oleh *Microsoft* untuk *platform Windows*. Software yang berdasarkan teknologi *ActiveX* umumnya terdapat dalam bentuk *internet explorer*, *plugins*, dan kebanyakan dalam bentuk *ActiveX controls*.

Sebelum adanya teknologi *ActiveX*, terdapat dua standar yang umum untuk *Microsoft Windows*. Yang pertama adalah OLE (*Object Linking Embeded*) dan yang lainnya adalah COM (*Component Object Model*). Kedua skema tersebut diciptakan untuk operasi antar *software*. Yang lebih dahulu difokuskan pada komunikasi dan selanjutnya difokuskan pada implementasi. Pada tahun 1996, *Microsoft* menggabungkan kedua teknologi tersebut ke dalam bentuk *ActiveX*.

Untuk menjadi suatu komponen ActiveX suatu objek harus mengimplementasikan antarmuka *IDispatch*. Antarmuka ini memungkinkan setiap objek meminta suatu *list pointer* ke antarmuka objek lain yang mendukung. *Developer* dapat beriterasi melalui *list* ini dan membuat *references* untuk antarmuka. *Microsot Office* merupakan contoh *software* yang menerapkan

ActiveX. Menggunakan bahasa pemrograman yang khusus, *developer* dapat memanipulasi dokumen dan *spreadsheets* dari kode sama seperti jika mereka menggunakan aplikasinya secara langsung.

Suatu *ActiveX controls* merupakan suatu komponen yang dapat digunakan berulang-ulang yang mengimplementasikan antarmuka *IDispatch*. Komponen tersebut jumlahnya tidak banyak pada seluruh aplikasi. Cukup dengan blok pembangun kecil yang dapat digunakan oleh aplikasi yang berbeda. Faktanya *command button* terlihat sama pada setiap program pada suatu *platform* merupakan suatu contoh dari komponen yang dapat digunakan berulang-ulang dan tidak hanya terbatas pada kontrol *ActiveX*.

Kontrol ActiveX dapat dibandingkan dalam satu pengertian dengan *java applet*. Karena kedua teknologi tersebut sama-sama bertindak sebagai *abstraction layer* antara *developer* dengan sistem operasi. Dari sana jelas terlihat perbedaannya. Perbedaannya adalah *Java applets* dapat berjalan pada setiap *platform* sementara komponen *ActiveX* hanya kompatibel dengan *web browser Microsoft* dan sistem operasi *Microsoft*. *ActiveX control* juga memungkinkan pengendalian *Windows* dengan level yang lebih tinggi dibandingkan dengan *Java applets*, membuatnya lebih kuat dan berbahaya. *Malware* seperti virus komputer dan *spyware* sering disebarkan sebagai *ActiveX applets* pada suatu halaman *web*.

ActiveX controls dapat ditulis dengan MFC, ATL, C++, *Borland Delphi* dan *Visual Basic*. Contoh umum *ActiveX controls* adalah *command button*, *list box*, *dialog boxes* dan bahkan *browser Internet Explorer*.