BAB II
TINJAUAN PUSTAKA

2.1 Jaringan Cerdas

Semakin pesatnya perkembangan teknologi telekomunikasi akan melahirkan suatu jaringan yang disebut jaringan cerdas atau lebih dikenal sebagai *Intelligent Network*, jaringan cerdas ini dapat diintegrasikan pada jaringan telepon yang sudah ada, yaitu dengan menerapkan konsep pemusatan *software* untuk mengontrol layanan-layanan khusus yang ditawarkan oleh jaringan telekomunikasi. Sehingga dapat disimpulkan bahwa jaringan cerdas adalah suatu arsitektur jaringan di atas jaringan (*transport*) telekomunikasi yang merupakan *platform* untuk manajemen servis secara terpusat baik untuk servis *eksisting* maupun servis baru, dan mampu memberikan nilai tambah pada jasa pelayanan telekomunikasi dan melaksanakan fungsi pengawasan serta pencaturan pada pelayanan tersebut. Hal ini dikarenakan dua sifat mendasar yang dimiliki jaringan cerdas, yaitu yang pertama jaringan cerdas tidak terikat dengan bentuk pelayanan, dan yang kedua jaringan cerdas tidak dipengaruhi oleh sistem jaringan yang ada, jenis pelayanan yang sudah ada, dan oleh komponen jaringan yang digunakan dalam sistem jaringan tersebut.

Arsitektur dasar jaringan cerdas, disusun berdasarkan modul dasar fungsi pelayanan yang disebut komponen fungsional (*Standar Bellcore*) atau *Service Independent Building Blocks*-SIB (Standar CCITT) yang bisa dipindahkan antar berbagai jaringan fisik. Dengan kata lain terjadi pemisahan antara logik *routing* jaringan sebagai fungsi penyambungan dengan logik *intelligent* sebagai fungsi pelayanan. Sehingga informasi untuk penanganan panggilan tidak perlu lagi harus berada dalam komponen *switching*, tetapi bisa juga berada pada komponen lain.

Dengan prinsip diatas maka dapat dihasilkan suatu jenis pelayanan baru dengan menggabungkan atau mengkombinasikan beberapa *feature* servis tunggal, dari blok servis yang sudah ada. Pengaturan set blok servis ini akan memperkaya jenis pelayanan jaringan cerdas yang pengembangan lebih lanjutnya diusahakan dapat melibatkan peran pelanggan secara interaktif.

2.2 Perkembangan Teknologi Jaringan ke Arah Jaringan Cerdas

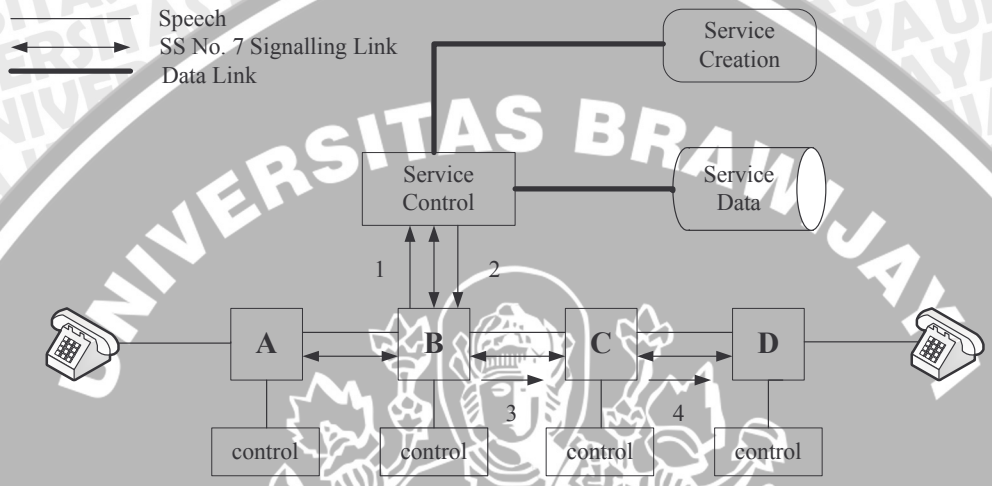
Perkembangan teknologi jaringan ke arah jaringan cerdas dimulai dengan penggunaan sistem *Stored Program Control* (SPC) yang dilengkapi dengan fasilitas yang mempunyai salah satu sifat *intelligent* dalam bentuk respon terhadap permintaan fasilitas pelanggan oleh pelanggan. Hal di luar dari fasilitas-fasilitas dasar yang dimiliki oleh jaringan penyambungan yang mempunyai tugas utama dalam menghubungkan komunikasi antar pelanggan.

Tetapi dengan kemampuan di atas, jaringan ini belum bisa sepenuhnya disebut jaringan yang ”cerdas”, karena baru merupakan blok-blok *intelligent* yang bekerja sendiri-sendiri dan hanya dapat beinteraksi dengan pelanggan, belum dapat melakukan interaksi dengan jaringan SPC lain dalam bentuk pertukaran pelayanan komunikasi.

Sehingga muncul gagasan untuk menyatukan sistem-sistem tersebut ke dalam suatu jaringan yang dapat berkomunikasi dalam bentuk transfer data dan pelayanan. Untuk penghubung antar sistem digunakan pensinyalan CCS (*Common Channel Signalling*) dan dilengkapi fasilitas basis data yang dapat digunakan secara bersama-sama oleh semua SPC. Basis data ini, ditujukan untuk memperbesar kapasitas penyimpanan data serta pengontrol pelayanan pada jaringan tanpa harus memperbesar kapasitas penyimpanan pada masing-masing SPC. Dengan konsep ini maka dimungkinkan terjadi pertukaran servis logik dari basis data ke *switching network* (SPC) yang membutuhkannya.

Konsep dasar pengendalian panggilan melalui jaringan cerdas dijelaskan pada gambar 2.1. Panggilan diasumsikan berasal dari sentral A. Untuk panggilan khusus yang membutuhkan pengendalian jaringan cerdas, urutan pengendalian pangilannya akan ditunda pada sentral B dan meneruskan panggilan permintaan panggilan khusus tersebut pada SCP (*Service Control Point*) yang akan menentukan bagaimana jenis panggilan tersebut dilayani (pesan 1). Selanjutnya SCP mengirimkan informasi pengendalian panggilan tersebut pada sentral B (pesan 2), untuk kemudian diteruskan pada sentral tujuan dengan menggunakan jaringan telekomunikasi konvensional (pada contoh gambar melalui sentral C dan sentral D atau pesan 3 dan pesan 4).

Pada kasus tertentu yang lebih rumit, pesan 2 mungkin memerlukan informasi khusus dari pemanggil. Untuk itu sentral B akan mengirimkan pemberitahuan pada pemanggil. Pemberitahuan tersebut berupa pesan-pesan dalam bentuk suara yang dihasilkan oleh unit pengendalian terpisah yang disebut *Intelligent Peripheral* (IP). Sedangkan tanggapan dari pemanggil berupa sinyal DTMF (*Dual Tone Multi Frequency*) dari pesawat telepon.



Gambar 2.1 Konsep dasar jaringan cerdas

Sumber: Uke Kurniawan

Elemen lain yang lebih rumit pada jaringan cerdas adalah adanya fasilitas *services creation* yang berupa sistem antarmuka manusia dan mesin (*man-machine interface*), yang memungkinkan penggunaan perangkat lunak secara menyeluruh tanpa operator manual. Sehingga prosedur pengendalian panggilan dapat dilakukan lebih cepat dibandingkan dengan penggunaan jaringan telekomunikasi konvensional.

2.3 Arsitektur Jaringan Cerdas

Standar CCITT, dalam rekomendasi Q. 1200, membagi arsitektur jaringan cerdas menjadi dua, yaitu arsitektur fungsional dan arsitektur fisik, yang disusun berdasarkan penggunaan fungsi-fungsi modular dari SIB yang terkandung dalam *Global Function Plane* (GFP) yang memiliki karakteristik utama sebagai berikut:

1. *Service Independent Building Blocks*, SIB dapat dimasukkan ke dalam sistem jaringan apapun tanpa tergantung jenis pelayanan yang sudah ada

pada jaringan tersebut

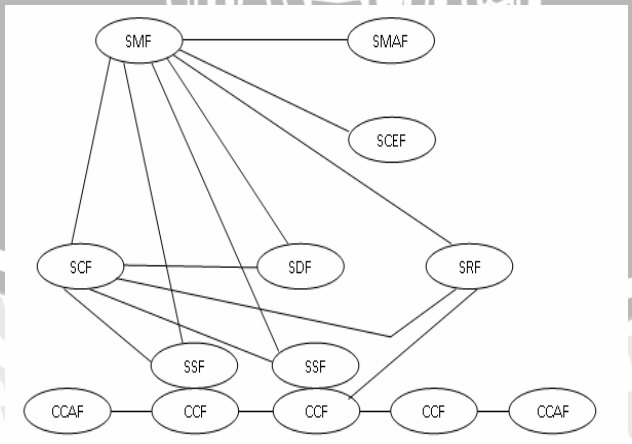
2. *Network Implementation Independent*, yang memungkinkan kemudahan penerapan ke dalam berbagai system jaringan
3. dapat menjadi interface tunggal dan stabil
4. merupakan modul *reuseable* yang dapat dipisah-pisahkan dan digabungkan kembali dalam berbagai kombinasi.

2.3.1 Arsitektur Fungsional

Pada arsitektur fungsional, pengembangan fungsi jaringan cerdas didasarkan pada kebutuhannya untuk memenuhi keinginan pelanggan dan kemudahan operasi pelayanan yang kemudian diselaraskan melalui pengembangan unsur fungsional jaringan cerdas untuk memenuhi lima kebutuhan yaitu:

1. kreasi pelayanan
2. manajemen pelayanan
3. pengolahan pelayanan
4. manajemen jaringan
5. *internetworking* pelayanan.

Distributed Functional Plane (DFP) yang dijelaskan dalam rekomendasi CCITT Q. 1204 tentang arsitektur dan Q. 1214 tentang CS 1, terdiri dari beberapa *Functional Entities* (FE) yang membentuk suatu group unik dengan fungsi khusus yang merupakan bagian dari konsep jaringan pintar secara keseluruhan, seperti gambar 2.2.



Gambar 2.2 Distributed Functional Plane

Sumber: Uke Kurniawan

Hubungan antar FE dilakukan melalui *Information Flow* (IF). Dalam suatu DFP, *functional entities* dapat terdiri dari:

1. CCAF (*Call Control Access Function*) yaitu memberikan dukungan khusus kepada *resource network* khusus yang berinteraksi dengan *caller.*
2. CCF (*Call Control Function*) yaitu mengendalikan pemrosesan *call* dan dan menyediakan layanan koneksi ke *network.*
3. SSF (*Service Switching Function*) yaitu memberikan dukungan *trigger* selama pemrosesan dan akses ke fungsionalitas jaringan cerdas.
4. SCF (*Service Control Function*) yaitu mengeksekusi *service logic* dan mempengaruhi *call processing* pada *switch* melalui *interface* ke SSF.
5. SDF (*Service Data Function*) yaitu mengelola *customer* dan data *network* untuk akses *real time* pada SCF dalam mengeksekusi layanan-layanan jaringan cerdas.
6. SRF (*Special Resource Function*) yaitu memberikan dukungan interaksi antara s/w pada *switch* dan *service control function*.
7. SCEF (*Service Creation Environment Function*) yaitu tempat dimana kreasi layanan dibangun, diuji, dan dimasukkan ke dalam SMF.
8. SMAF (*Service Management Access Function*) yaitu *interface* antara *service manajer* dan SMF (dapat diimplementasikan terpisah secara fisik atau SMAP)
9. SMF (*Service Management Function*) yaitu menyediakan layanan jaingan cerdas dan menyediakan dukungan kepada operasi yang sedang berjalan.

2.3.2 Arsitektur Fisik

Berdasarkan rekomendasi CCITT Q. 1205 dan Q. 1215 dijelaskan tentang *physical plane* yang menunjukkan bagaimana fungsi-fungsi yang ada pada DFP dapat diimplementasikan secara fisik pada jaringan, yaitu bagaimana suatu *functional entities* dapat ditempatkan pada suatu *physical entities* (PE).

Suatu PE minimal terdiri dari satu FE, tetapi dalam satu PE tidak boleh menggandung beberapa FE yang sama. *Physical plane* ini terdiri dari node dan antarmuka yang dapat berupa *software* maupun gabungan *software* dan *hardware*

yang memiliki fungsi-fungsi untuk menciptakan, mengeola dan mengoperasikan jaringan cerdas yang digunakan oleh penyediaan jasa telekomunikasi.

Physical Entities yang termasuk- pada tingkat ini meliputi :

1. SSP (*Service Switching Point*)

   Fungsi utama SSP adalah:

   1.1 fungsi deteksi *trigger*

   1.2 melakukan komunikasi dengan SCP

   1.3 melaksanakan perintah SCP

   1.4 membangkitkan rekaman panggilan

   1.5 fungsi manajemen trafik.

2. SCP (*Service Control Point*)

   Bertanggung jawab atas logik serves dan pengendaliannya. SCP berisi modul logik servis yang disebut *Service Logic Program* (SLP) dan data pemakai servis yang digunakan, yang gabungan keduanya disebut *service script*.

3. *Adjunct*

   Adjunct memiliki fungsi yang sama dengan SCP tetapi jalur komunikasi ke SSP dilakukan secara langsung. Sehingga *adjunct* dapat berupa antarmuka berkecepatan tinggi.

4. IP (*Intelligent Peripheral*)

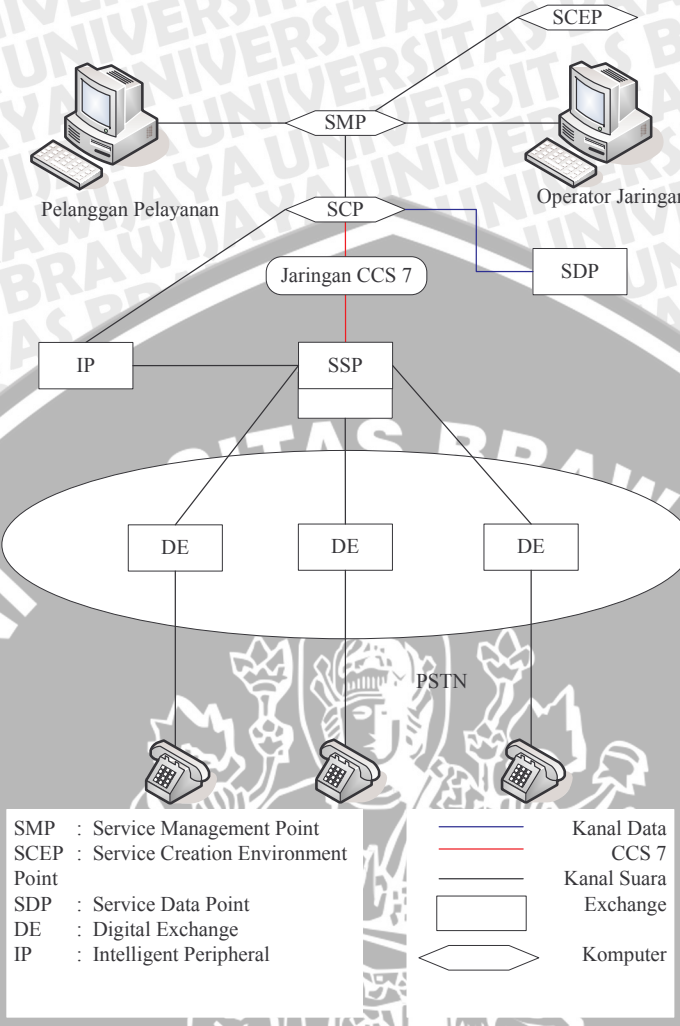
   IP adalah sistem yang menangani pengawasan dan manajemen fungsi-fungsi sebagai berikut:

   4.1 *Voice synthesis* (pembangkitan suara)

   4.2 *Announcement* (pemberitahuan)

   4.3 *Voice recognition* (pengenalan suara)

5. SMP (*Service Management Point*)



Gambar2.3 Arsitektur fisik jaringan cerdas

Sumber: Ambrosch

2.4 Proses Operasi

Proses operasi jaringan cerdas terdiri dari beberapa tahap seperti usulan pelayanan, pembentukan SLP, percobaan pelayanan baru, implementasi pelayanan baru, pemakaian pelayanan oleh pelanggan, dan pelaksanaan pelayanan. Yang tahap pelaksanaannya adalah sebagai berikut :

1. Usulan Pelayanan

   Pada tahap usulan suatu pelayanan baru, akan dianalisis kelayakan pelayanan tersebut sebelum dipasarkan secara luas. Hal-hal penting yang dijadikan pertimbangan antara lain kemungkinan pelaksanaan, pengaruh yang diberikan terhadap sistem jaringan dan sistem operasi yang telah ada,

pendapat pelanggan tentang pelayanan baru ini, serta segi keuangan dan keuntungan yang dibutuhkan dan dihasilkannya.

2. Pembuatan Pelayanan Baru

   Penciptaan pelayanan baru dilalui dengan penggabungan atau pengkombinasian beberapa SIB, yang merupakan fungsi dasar dari jaringan cerdas. Yang perlu diperhatikan dalam tahap ini adalah SIB yang tersedia, batasan, karakteristik, dan performansinya. Selain itu juga harus dibuat model dan simulasi SLP untuk mengetahui apakah fungsi pelayanan ini berjalan dengan baik.

3. Percobaan Pelayanan Baru

   Setelah dibuat model dan simulasinya berjalan baik, maka pelayanan ini akan dicobakan secara langsung ke seluruh jaringan untuk membuktikan apakah simulasi yang dilakukan sesuai penerapannya di lapangan. Untuk itu basis data di jaringan dan tabel pengenalnya disesuaikan dan SLP baru diletakkan pada jaringan dengan pengaksesan yang masih terbatas.

4. Implementasi Pelayanan Baru

   Setelah proses percobaan berjalan sesuai seperti yang dikehendaki, proses berikutnya adalah: mengimplementasikannya ke seluruh jaringan. Sementara itu dilakukan pula pelatihan pengoperasian, pengenalan ke pelanggan, penyusunan buku petunjuk pemakaian, dan sebagainya.

5. Penggunaan Pelayanan oleh Pelanggan

   Disini pelanggan sudah dapat mengakses langsung pelayanan yang ditawarkan. Dipersiapkan pula komponen penunjang yang akan membantu implementasi pelayanan tersebut. Perlu diperhatikan pula jenis peralatan terminal yang harus dimiliki pelanggan untuk memanfaatkan pelayanan ini.

6. *Interface* umum untuk Pelanggan

   Untuk melayani keinginan pelanggan dalam pengaturan sendiri parameter-parameter pelayanan yang dibutuhkan, perlu disediakan suatu *interface* umum untuk pelanggan (UCI-*Universal Customer Interface*) yang dapat berkomunikasi dengan berbagai terminal pelanggan, sehingga pelanggan dapat berkomunikasi secara langsung dengan SMS serta menerjemahkan

perintah-perintah SMS ke dalam bentuk suara yang dikirim ke telepon pelanggan atau teks ke layar monitor terminal komputer pelanggan.

7. Pelaksanaan

   Pelayanan baru ini sudah dapat dipergunakan oleh pelanggan sewaktu-waktu. Jaringan akan mendeteksi adanya permintaan pelayanan melalui tabel pengenal pada SSP. SSP memantau saluran pelanggan dan trunk untuk membandingkan keadaan yang masuk dengan tabel pengenal. Jika sesuai, SSP mengirim *query* (daftar pertanyaaan) ke SCP sesuai pengalamatan yang ada dalam SSP. SCP memproses SLP yang digabung dengan data pemakai yang ada pada basis data SCP untuk menangani *query* tersebut. Setelah *service script* sesuai seperti yang diinginkan, akan dikirim ke SSP untuk menanggapi permintaan pelayanan pelanggan. Proses ini akan berakhir bila sudah terjadi aksi dari pemanggil atau jika eksekusi SLP selesai.

2.5 Pelayanan Jaringan Cerdas

Beberapa jenis pelayanan yang ditawarkan dalam pengimplementasiannya pada jaringan cerdas, antara lain:

1. *Advanched Freephone Service*

   Pelayanan 0800, dimana biaya panggilan dibebankan kepada pelanggan dipanggil, pelanggan pelayanan ini. Biasa digunakan oleh pengelola bisnis untuk menarik minat pelanggan bisnis yang ditawarkan.

2. *Pay Per View*

   Memberi fasilitas pelanggan pemakai pelayanan ini untuk menyelenggarakan suatu jasa melalui hubungan telepon dengan biaya tambahan yang dibebankan ke pelanggan pengguna jasa tersebut, seperti konsultasi kesehatan atau kecantikan, ramalan cuaca, saran perpajakan, dan sebagainya. Biaya dipungut melalui pemakaian pulsa dengan pembagian yang telah disetujui bersama dengan pihak pengelola jasa telekomunikasi.

3. *Emergency Response Service*

   Memungkinkan penggunaan satu nomor untuk melayani seluruh panggilan di seluruh daerah, tanpa kode wilayah. Jaringan cerdas akan secara otomatis

menghubungkan panggilan ke cabang terdekat dari pelanggan pemanggil akses ini. Sering digunakan oleh perusahaan *delivery fast food*.

4. *Alternate Billing Service*

   Menawarkan sistem pembayaran alternatif yang mampu memindahkan rekening tagihan pemakaian pulsa pada suatu nomor atau rekening sesuai permintaan. Yang termasuk didalamnya adalah *calling card*, *collect call*, dan *third party call*.

Khusus untuk pelayanan *collect call* dan *third party call* akan dibahas dalam buku ini secara lebih mendalam. Proses hubungan, logika pelayanan hingga pembuatan simulasi proses bubungan kedua pelayanan tersebut akan dibahas pada bab IV.

2.6 Pengertian Dasar *Third Party Call* dan *Collect Call*

Third party call adalah sebuah layanan telepon yang pembayarannya dilakukan oleh salah satu dari ketiga pemakai layanan ini yaitu pemanggil telepon atau penerima telepon. Ketika penerima telepon menerima telepon, operator atau *automated service* akan memberitahu bahwa itu adalah layanan third party call dan memberitahu nama pemanggil telepon dan akan menanyakan apakah penerima telepon mau membayar tagihan atau berkurang pulsanya (*balance*). Jika penerima telepon menerima tawaran tersebut, maka pemanggil telepon akan dihubungkan dengan nomor yang lain, dan kemudian operator akan *disconnect.* Pada umumnya jika operator tidak mampu untuk mendapatkan seseorang yang akan mau membayar tagihan, maka percakapan tidak akan terjadi. Penerima telepon tidak akan terhubung pada suatu percakapan jika penerima telepon menolak untuk membayar tagihan. Ada dua cara untuk melakukan layanan *third party call*. Yang pertama yaitu mendial operator jarak jauh dan meminta pembayaran tagihan dilakukan oleh si penerima telepon. *Dial* 00, atau 0 ditambah dengan kode area dan nomor telepon yang dituju, atau menggunakan 10-10 kode akses yang disediakan oleh *preferred long distance company.* Yang kedua, dapat mendial nomor special yaitu 800 yang disediakan oleh *the long distance company*. Itu biasanya memberi harga rata-rata lebih murah dibandingkan mendial melalui operator.

Sedangkan *collect call* adalah layanan telepon yang pembayarannya dilakukan oleh penerima telepon. Jadi jika biasanya penelpon yang akan membayar tagihan atau berkurang pulsanya (*balance*), kebalikan dari itu fasilitas *collect call* berarti pembayaran atau pengurangan pulsa telepon dilakukan pada sisi penerima (*called party*). Fasilitas ini sering juga disebut *reverse charge*. Fasilitas ini muncul pada dasarnya karena setiap operator ingin mengingkatkan keuntungannya (ARPU), oleh karena itu operator akan selalu mencari cara agar keberhasilan pada saat pemanggilan (*call completion*) meningkat. Ketidakberhasilan saat pemanggilan bisa jadi karena masalah teknis atau hanya karena si pemanggil tidak memiliki cukup *balance*. Lalu dicarilah cara agar penelpon tetap bisa melakukan percakapan tanpa perlu membayar tapi uang diambil dari orang lain yaitu orang yang ditelpon. Hampir sama dengan layanan *third party call*, ada dua cara untuk melakukan layanan *collect call*. Yang pertama yaitu mendial operator jarak jauh dan meminta untuk melakukan layanan *collect call* ke nomor yang akan dituju. Dial 00 atau 0 ditambah kode area dan nomor telepon yang dituju, atau menggunakan 10-10 kode akses yang disediakan oleh *preferred long distance company*. Yang kedua, dapat mendial nomor special yaitu 800 yang disediakan oleh *the long distance company*. Itu biasanya memberi harga rata-rata lebih murah dibandingkan mendial melalui operator.

Kedua layanan tersebut, yaitu *third party call* dan *collect call* dapat ditolak atau diblok di jalur telepon penerima telepon, dimana akan mencegah pemanggil telepon untuk melakukan *collect call* dan/ atau pembayaran telepon layanan *third party call* pada penerima telepon. Biasanya tidak dikenakan biaya untuk melakukan penolakan atau pemblokingan. Bisa juga dengan menghubungi perusahaan telepon terdekat untuk selalu membloking atau selalu tidak membloking kedua layanan tersebut.

Seperti telah disebutkan pada bab I, standar yang ada dalam pelayanan *collect call* dan *third party call* masih menggunakan sistem pelayanan operator (OSS) untuk pelayanan yang khusus seperti dijelaskan di atas dan penggunaan suatu basis data terpusat yang terletak dalam SCP. SSP dan OSS masing-masing mempunyai akses langsung ke basis data melalui jaringan CCS-7.

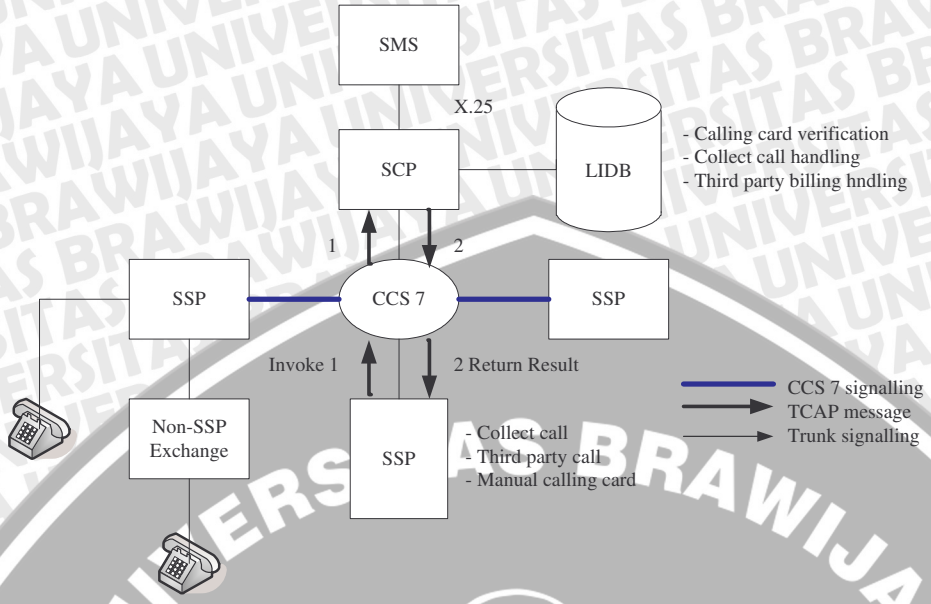
Semua basis data yang terpusat dalam SCP dikendalikan oleh suatu SMS (*Service Management System*). SMS dapat menerima informasi-informasi pelayanan secara otomatis (untuk pelayanan terjadwal) ataupun menerima pembaharuan informasi-informasi pelayanan tersebut dari pelanggan yang mempunyai akses langsung ke SMS.

Pada saat seorang pelanggan menghubungi nomor akses khusus pelayanan *collect call* atau *third party call*, maka SSP akan segera mendeteksi panggilan tersebut dan mencari tahu kondisi beban di SCP dengan sinyal ACG (*Automatic call Gapping*). Sinyal ACG mengindikasikan terjadinya beban lebih pada SCP dan SSP akan mengakhiri permintaan pelayanan dengan *busy tone*. Apabila tidak terdeteksi sinyal ACG tersebut, maka SCP tidak dalam kondisi beban lebih sehingga hubungan memungkinkan untuk dilanjutkan.

Dari sini proses pembangunan hubungan dapat diteruskan. SSP akan meminta instruksi SCP tentang hubungan yang diminta pelanggan pemanggil dan mengirimkan *message* ke SCP untuk mengakses basis data pelanggan dan fasilitas yang dibutuhkan untuk melayani permintaan hubungan tersebut. SCP akan memberikan informasi-informasi yang dibutuhkan untuk penanganan hubungan yang diinginkan dan dikirim kembali ke SSP melalui jaringan CCS-7.

Untuk memberikan *announcement* yang dibutuhkan pada kasus yang membutuhkan persetujuan pelanggan dipanggil tentang suatu hubungan yang dikehendaki (*collect call* atau *third party call*), jaringan cerdas menstandarkan suatu sistem operator pelayanan atau OSS. OSS disini mendapat perintah dari SCP melalul SSP dengan aliran informasi tertentu dalam jaringan CCS-7.

Proses terbentuknya hubungan untuk pelayanan *collect call* dan *third party call* dapat dilihat dari gambar 2.4.



Gambar 2.4 Proses pelayanan third party call dan collect call

Sumber: ejlp12

Proses pembangunan hubungan pelayanan *collect call* dan *third party call* dijelaskan di bawah ini beserta alokasi kejadiannya:

1. pengguna pelayanan mendial prefiks akses pelayanan *collect call* atau *third party call*. SSP mendeteksi adanya permintaan pelayanan khusus tersebut (*collect call* atau *third party call*)
2. SSP akan meminta informasi yang dibutuhkan untuk penanganan hubungan tersebut ke SCP. CCS 7 ke SCP dimana terkandung basis data pelayanan seluruh pelanggan.
3. melalui SSP, OSS akan mengirim pertanyaan-pertanyaan (TCAP *message*) dengan menggunakan jaringan.

2.7 Status Pelayanan Pelanggan

Dalam simulasi yang akan dibuat untuk kedua pelayanan tersebut, basis data pada SCP akan berisi status pelayanan pelanggan yang merupakan suatu pilihan yang diberikan kepada pelanggan pelayanan, yang nantinya dapat diakses secara langsung oleh pelanggan yang bersangkutan apabila terdapat perubahan status sesuai yang diinginkannya melalui SMP.

Status pelayanan pelanggan yang digunakan khusus untuk kedua pelayanan ini diantaranya adalah status penagihan biaya penggunaan pelayanan pada saat pelanggan memanggil dan dipangil. Pemberian status pelayanan ini diharapkan dapat mempermudah operasional pembangunan hubungan dan member keleluasaan pelanggan akan status pelayanan yang diinginkannya sesuai kondisi pelanggan yang bersangkutan pada suatu saat.

Pada kondisi pelanggan pemangggil, status penagihan biaya penggunaan pelayanannya terdiri dari dua pilihan, yakni pelanggan pemanggil sebagai penanggung biaya penggunaan pelayanan dan penagihan biaya penggunaan pelayanan dibebankan ke pelanggan dipanggil.

Bertindak sebagai pelanggan dipanggil, pelanggan memiliki tiga pilihan sebagai status pelayanan untuk penagihan biaya, yakni :

1. selalu menerima pembebanan biaya penggunaan pelayanan
2. selalu menolak semua pembebanan biaya penggunaan pelayanan
3. melalui konfirmasi terlebih dahulu.

Pemberian status penagihan penggunaan pelayanan seperti disebut di atas tentu saja masih memiliki peluang memberikan implikasi yang kurang menguntungkan bagi pengelola jaringan dan juga masih membuka kondisi dimana pelanggan tidak merasakan kelebihan yang ditawarkan. Apabila semua pelanggan pemanggil selalu membebankan biaya penggunaan pelayanan ke pelanggan dipanggil dan dalam saat yang bersamaan semua pelanggan dipanggil selalu menolak pembebanan biaya penggunaan pelayanan yang masuk, maka dalam kondisi tersebut selamanya tidak akan pernah terjadi hubungan yang menggunakan pelayanan tersebut.

Demikian juga kepada pelanggan yang selalu menolak penagihan biaya penggunaan pelayanan, pelanggan tidak merasakan kelebihan pelayanan yang ditawarkan pengelola jaringan malahan akan merasakan kerugiannya jika pada saat yang bersamaan terdapat panggilan masuk dari kerabat yang sebenarnya diinginkan melakukan hubungan dengan pembebanan biaya, namun status pelayanan yang langsung saja memblokir semua panggilan yang menggunakan pelayanan tersebut.

Kondisi-kondisi di atas merupakan suatu perumpamaan yang sebenarnya dapat terjadi meskipun dengan probabilitas yang kecil. Untuk itu masih perlu

diperhitungkan kondisi-kondisi lain yang akan timbul dan dicari jalan keluar untuk masing-masing kondisi untuk mendapatkan suatu momen optimum yang dapat dirasakan pelanggan pengguna pelayanan dan keuntungan yang memadai bagi pengelola pelayanan.

2.8 Proses Pensinyalan

Standar pensinyalan yang digunakan dalam penanganan hubungan dan pelayanan pada jaringan cerdas menggunakan CCS 7. Secara umum CCS 7 akan menyediakan prosedur penyaluran informasi menggunakan sejumlah data dalam interkoneksi dua atau lebih node. Pensinyalan dalam jaringan telekomunikasi digolongkan dalam 2 klasifikasi, yaitu:

1. Pensinyalan *User-network*

   Adalah pensinyalan antara terminal pelanggan dengan sentral pelayanan. Pada PSTN penanganannya masih menggunakan pensinyalan *Dial-Pulse* atau *Dual Tone Multi Frequency*.

2. Pensinyalan *Interoffice*

   Pensinyalan antar *signaling point* atau *signaling transfer point* untuk aplikasi jaringan cerdas berupa pertukaran SCCP(*Signaling Connection Control Part*)/TCAP (*Transaction Capability Application Part*).

Format umum SCCP yang mempunyai fungsi mengendalikan hubungan pensinyalan dalam jaringan CCS 7 secara logika dan mengirimkan data pensinyalan tersebut. Bagian-bagian dalam SCCP adalah :

1. *Routing Label*

   Didalamnya terdapat *Destination* dan *Originating Point Code* yang menunjukkan kode nomor node yang dituju dan pengirim *message*. *Signaling Link Selection* berguna dalam pemilihan satu *signaling link* dari suatu set *signaling link* untuk proses pengiriman *message* tersebut.

2. *Message Type Code*

   Terdiri dari satu oktet yang mutlak harus ada dalam setiap *message* yang berguna dalam penentuan fungsi dan format tiap *message*.

3. *Mandatory Fixed Part*

   Memuat parameter-parameter yang bersifat *mandatory* dengan posisi, panjang dan urutannya ditentukan oleh tipe *message* secara unik.

4. *Mandalory Variabel Part*

   Parameter-parameter dengan panjang tidak tetap dengan nama dan urutan pengiriman pointernya terdapat dalam tipe *message* yang dinyatakan secara implisit. Parameter yang ada dan sering digunakan dalam jaringan cerdas antara lain :

   4.1 *called Party Address*

   4.2 *calling Party Address*

   4.3 data/TCAP

5. *Optional Part*

   Berisi parameter-parameter yang boleh ada maupun tidak dalam suatu *type message* tertentu.

Dalam proses pembentukan hubungan pensinyalan, *service Switching Point* (SSP) akan mengenali suatu panggilan yang membutuhkan penanganan dengan jaringan cerdas. SSP akan menunggu sinyal *Automatic Call Gapping (ACG)* dari *Service Control Part (SCP)* yang menyatakan beban lebih SCP. Jika tidak terdapat ACG, SSP akan mengirimkan *message* dengan format suatu permintaan bantuan penanganan pelayanan dan data-data pelanggan pengguna untuk penanganan pelayanan selanjutnya. SCP dengan segera mengirimkan perintah penanganan penyambungan hubungan beserta semua data yang dibutuhkan untuk melaksanakan perintah tersebut.

2.9 Perhitungan Trafik

2.9.1 Intensitas Trafik

Intensitas trafik merupakan nilai trafik dari sekelompok kanal yang menyatakan lamanya waktu pendudukan yang diolah oleh sekelompok kanal dalam suatu interval waktu (1jam) tertentu jam sibuk. Untuk melihat karateristik trafik, terdapat dua parameter yang penting yaitu jumlah panggilan selama periode waktu pengamatan (1jam) dan rata-rata waktu pembicaraan setiap panggilan (lamanya

kanal diduduki untuk pembicaraan). Jika trafik yang ditawarkan adalah A yang dinyatakan dalam Erlang, maka dapat dinyatakan:

$$A = X.t_h \qquad (2.1)$$

Sumber: Ericson Lind

Dimana:

A = trafik yang ditawarkan (erlang)

X = rata-rata kedatangan (panggilan/ jam)

t_h = rata-rata waktu pendudukan (jam/ panggilan)

2.9.2 Derajat Pelayanan

Parameter trafik yang penting dalam sistem seluler yang harus diperhatikan adalah tingkat pelayanan atau *Grade of Service* (GOS). *Grade of Service* (B) adalah probabilitas suatu hubungan tidak berhasil karena tidak tersedianya fasilitas seperti kanal radio. Dalam prakteknya GOS adalah perbandingan antara jumlah panggilan yang tidak dapat dilayani dengan segera dengan jumlah panggilan yang ditawarkan. Panggilan yang hilang atau harus menunggu tersebut adalah disengaja karena adanya pembatasan jumlah server (peralatan *switching*, saluran) karena pertimbangan ekonomis. Biasanya GOS dinyatakan dalam persen.

Dalam sistem hilang (*loss system*) GOS (B) dapat dinyatakan dengan perbandingan antara trafik yang hilang dengan trafik yang ditawarkan pada jam sibuk, yaitu:

$$B = \frac{R}{A} = \frac{A - Y}{A} \qquad (2.2)$$

Sumber: Ericson Lind

Dimana:

B = tingkat pelayanan (%)

A = trafik yang ditawarkan (erlang)

Y= trafik yang dilayani (erlang)

R= trafik yang hilang (erlang)

Dengan asumsi pola kedatangan panggilan mengikuti distribusi *poisson* dengan *rate* kedatangan panggilan konstan dan waktu layanan panggilan

mengikuti distribusi eksponensial negatif, probabilitas bloking mengikuti distribusi Erlang dengan hubungan sebagai berikut:

$$B = \frac{\frac{A^N}{N!}}{\sum_{i=1}^{N} \frac{A^i}{i!}} \qquad (2.3)$$

Sumber: Ericson Lind

Dimana:

A= trafik yang ditawarkan (erlang)

B= probabilitas bloking suatu sistem (tanpa satuan)

N= jumlah kanal pembicaraan (tanpa satuan)

Dari hubungan diatas, untuk harga B dan N yang diketahui, kapasitas trafik yang ditawarkan pada prinsipnya dapat ditentukan. Tetapi secara matematis harga A sukar ditentukan. Hubungan dari ketiga parameter B,A dan N diatas telah disusun dalam sebuah tabel yang biasa dikenal tabel rugi Erlang B. Dari tabel Erlang tersebut dapat ditentukan trafik A untuk harga B dan N yang diketahui.

2.10 Sistem Pensinyalan No. 7

2.10.1 Perkembangan Pensinyalan dalam Standarisasi ITU-T

Common Channel System No. 7 yang dikembangkan oleh ITU dioptimalkan digunakan pada lingkungan jaringan digital yang menggunakan jalur pensinyalan dengan kecepatan 64 Kbps. Sistem didesain untuk pemakaian multi-fungsi yaitu untuk *signaling call control*, untuk layanan telepon dan data, serta manajemen dan perawatan jaringan. Perkembangan teknik pensinyalan dapat dilihat pada tabel 2.1 dibawah ini :

Tabel 2.1 Perkembangan Teknik Pensinyalan
Sumber: Cisco System

2.10.2 Keunggulan dan Kelemahan CCS7

Keunggulan CCS7

Pensinyalan CCS7 mempunyai beberapa keunggulan diantaranya:

1. Mampu menangani trafik yang tinggi
2. Mempunyai fleksibilitas yang lebih tinggi dalam menerima aplikasi pelayanan baru dengan tersedianya kemungkinan pendefinisian sinyal yang cukup banyak
3. Status jalur pensinyalan tidak tergantung pada status panggilan karena jalur pensinyalan terpisah dari jalur suara/ data. Hal ini menyebabkan penggegaman sirkit lebih singkat sehingga penggunaan sirkit lebih efisien.
4. Tersedianya kecepatan transmisi sinyal yang tinggi (64 Kbps) akan memperpendek *post dialing delay.*

Kelemahan CCS7

Disisi lain CCS7 mempunyai beberapa aspek yang kurang menguntungkan antara lain:

1. Biaya investasi awal yang sangat tinggi untuk pengadaan perangkat baru baik perangkat transmisi data maupun perangkat control
2. Diperlukan persyaratan keamanan yang lebih handal untuk pensinyalan jalur data

3. Jaringan CCS7 sangat kompleks sehingga perlu dibuat konsep perancangan jaringan pensinyalan CCS7 secara nasional.

2.10.3 Struktur Jaringan CCS7

2.10.3.1 Konsep Dasar

Jaringan telekomunikasi yang dilayani dengan CCS7 dibentuk dari sejumlah titik penyambungan dan processing yang diinterkoneksikan dengan jalur transmisi. Untuk berkomunikasi dengan menggunakan CCS7, titik-titik tersebut memerlukan bentuk layanan CCS7 sehingga dapat berfungsi sebagai SP dalam jaringan CCS7. Agar informasi dapat dikirim atau diterima antara SP diperlukan interkoneksi melalui jalur pensinyalan. Kombinasi dari beberapa SP dan beberapa jalur pensinyalan membentuk jaringan CCS7.

2.10.3.2 *Signaling Point*

Signaling point adalah semua titik dalam jaringan yang mampu menangani *control message* CCS7. Titik ini dapat berupa suatu *signaling end point* atau *signaling transfer point*. SEP digunakan untuk mengontrol pesan dan tidak dapat memproses pesan yang tidak secara langsung ditujukan ke titik tersebut. Sedangkan STP adalah suatu titik yang mampu mentransfer pesan yang diterima pada suatu jalur pensinyalan ke jalur pensinyalan yang lain.

Contoh-contoh titik dalam suatu jaringan pensinyalan yang dapat berfungsi sebagai SP adalah:

1. Sentral
2. Pusat operasi, administrasi dan pemeliharaan
3. *Control Service Point*
4. *Signaling Transfer Point*

Semua pensinyalan dalam jaringan CCS7 diidentifikasikan oleh kode khusus yang disebut *Signaling Point Code*.

2.10.3.3 Link Pensinyalan

Pada sistem CCS7, jalur pensinyalan digunakan untuk membawa pesan pensinyalan antara dua signaling point.

2.10.3.4 Mode Pensinyalan

CCS7 mempunyai dua mode hubungan pensinyalan, yaitu mode *associated* dan *non-associated*. Mode *non-associated* dapat berbentuk *quasi-associated* dan *fully-non-associated*. Pada mode *associated*, pesan-pesan antara dua sentral (SP) ditransfer melalui suatu *common signaling link* yang terhubung ke sentral yang sama sebagaimana sirkit grup pembicaraan disambungkan, sedangkan pada mode *non-associated* pesan-pesan antara dua sentral (SP) ditransfer melalui dua atau lebih *common signaling link* dan melalui satu atau lebih STP menuju sentral (SP) yang dituju. Mode *quasi-non-associated* adalah salah satu bentuk dari mode *non-associated*, pada mode ini, pesan-pesan antara dua sentral (SP) ditransfer melalui dua atau lebih *common signaling link* pada tandem melalui lintasan dan STP yang telah ditentukan sebelumnya. Pada mode *fully-non-associated* pesan-pesan antara dua sentral (SP) ditransfer melalui lintasan manapun yang tersedia dalam jaringan pensinyalan.

2.10.3.5 Rute Pensinyalan

Rute pensinyalan adalah lintasan pesan yang telah ditentukan sebelumnya. Lintasan tersebut terdiri dari STP dan jalur pensinyalan yang berada diantara *originating point* dan *destination point*. Semua rute pensinyalan yang dapat digunakan dalam satu jaringan pensinyalan disebut rute pensinyalan.

2.10.4 Konsep Arsitektur CCS7

CCS7 didefinisikan sebagai tipe pensinyalan multifungsi yang memungkinkan untuk membawa seluruh data informasi seperti *speech*, data komputer dengan kecepatan tinggi, dan dapat mendukung beberapa aplikasi seperti IN, PSTN, ISDN, dan PCN.

Struktur 4 level digunakan apabila memakai hubungan *circuit related* yaitu pensinyalan yang berhubungan dengan terjadinya hubungan suatu panggilan. Sedangkan struktur 7 layer OSI digunakan bila mengadakan hubungan *non-circuit related* yaitu pensinyalan yang tidak langsung berhubungan dengan suatu panggilan, misalnya pemeliharaan jaringan dan mencari terjemahan perutean.

2.10.4.1 Struktur 4 level CCS7

Objek utama struktur 4 level CCS7 adalah penerapan hubungan *circuit related*. Contoh pelayanan yang dapat diambil untuk aplikasi 4 level ini adalah ISDN. Hal ini dimungkinkan karena konsep ISDN User Part (level 4) membelah fungsinya kedalam *Call Control Part* dan *Bearer Connection Control Part*. *Call Control Part* menangani aspek yang berhubungan dengan panggilan dan tidak tergantung dengan trafik sirkit yang dipakai, sedangkan B*earer Connection Control Part* menangani fungsi pengendalian trafik sirkit yang dimaksud, misalnya fungsi *circuit related*. Fungsi yang terdapat pada masing-masing level dapat dijelaskan sebagai berikut:

1. Level 1

   Mendefenisikan karakter fisik, listrik dan fungsional dari jalur pensinyalan yang berisi kanal transmisi digital untuk pertukaran sinyal pada dua arah yang berbeda secara serentak. Blok dari *switching* digital atau peralatan terminal digunakan untuk melewatkan sinyal diantara terminal-terminal.

2. Level 2

   Mendefenisikan fungsi dan prosedur untuk mengirim sinyal informasi jalur data, serta jalur pensinyalan termasuk pengendalian *error* dan pemantauan jalur. Fungsi level 2 bersama dengan level 1 bertanggung jawab untuk menjamin keandalan transmisi dan pemindahan informasi diantara dua *signaling point*.

3. Level 3

   Secara umum mendefenisikan fungsi-fungsi sinyal informasi secara langsung dalam jaringan pensinyalan (tergantung dari kondisi jaringan). Untuk keperluan pengetesan dan perawatan juga dilakukan pada level ini. Level 3 mendefenisikan penanganan berita (*message handling function*) yang bertanggung jawab terhadap perutean dan manajemen jaringan pensinyalan (manajemen jaringan pensinyalan) yang berfungsi untuk menangani pengendalian konfigurasi jaringan.

4. Level 4

   Level 4 merupakan *user part* yang berfungsi untuk mendefenisikan arti dan urutan dari pesan yang disalurkan melalui level 1, 2 dan 3 untuk pemakai

yang berbeda. Sebagai contoh adalah pensinyalan untuk telepon atau data. *user part* didefenisikan berdasarkan format pesan dan prosedurnya.

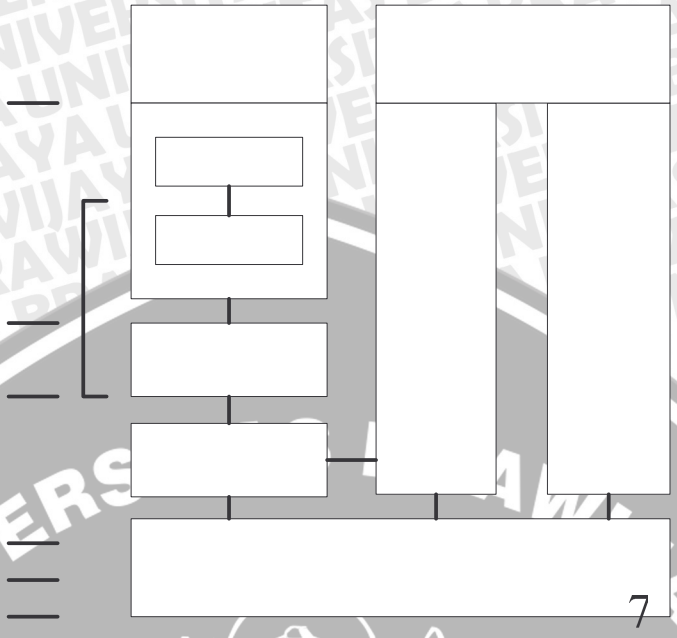
Kombinasi level 1, 2 dan 3 disebut MTP (*Message Tranfer Part*) secara umum bertanggung jawab terhadap mekanisme pemindahan informasi meliputi jumlah pesan, jalur dan fungsi manajemen jaringan. Namun demikian MTP tidak memperdulikan arti pesan yang dibawa dari level 4. MTP hanya mengarahkan pesan yang berhubungan dengan perutean (*message addressing*) yang tercantum dalam *field* label perutean yang berdasarkan format pesan dan prosedurnya terdiri atas alamat asal, alamat tujuan dan kode bit untuk mencari jalur yang akan dipergunakan.

2.10.4.2 Struktur 7 layer OSI CCS7

Objek utama struktur 7 layer OSI adalah penerapan hubungan *non-circuit related* seperti komunikasi data antar prosesor. Tujuan dari konsep layer ini adalah untuk mendefenisikan fungsi yang diperlukan diantara dua komputer dalam lingkungan 7 layer OSI. Contoh layanan yang dapat diaplikasikan seperti IN dan GSM. Dalam jaringan telekomunikasi, *costumer*, sentral dan titik (*node*) disebut *entity*. *Entity* yang melakukan komunikasi dalam hal ini komputer.

Konsep 7 layer OSI pada dasarnya hampir sama dengan struktur 4 level CCS7 namun fungsinya dikatagorikan dalam layer 7. Tiap lapisan dapat berkomunikasi dengan lapisan yang setingkat melalui layer yang lebih rendah dan setiap layer tersebut fungsi yang terdefenisi, setiap layer melakukan komunikasi dengan menggunakan *primitive*.

Gambar 2.5 adalah arsitektur CCS7 secara umum yang dibangun atas struktur 4 level dan 7 layer OSI.



Gambar 2.5 Arsitektur CCS7 yang terdiri dari 4 level dan 7 layer OSI
Sumber: Cisco System

Berdasarkan basis pelanggan ke pelanggan (*circuit related*), jaringan telekomunikasi dianggap telah mampu menyediakan fungsi dari layer 1 sampai layer 3. fungsi layer 4 sampai layer 7 disediakan oleh pelanggan (user). Untuk penerapan 7 layer OSI, fungsi layer 1 sampai layer 3 perlu ditambahkan suatu elemen fungsional yakni *Signaling Connection Control Part* (SCCP) karena MTP tidak dapat menyediakan seluruh fungsi layanan jaringan OSI. Kombinasi antar MTP dan SCCP disebut *Network Service Part* (NSP). NSP mampu memenuhi layanan-layanan layer 3 seperti didefenisikan pada referensi model OSI. Fungsi layer 4 sampai laver 7 (Uver Part) ditentukan oleh elemen funsgsional yang bersifat umum yaitu *transaction capability* (TC). TC dibelah fungsinya menjadi 2 bagian yaitu *Intermediate Service Part* (ISP) dan *Transaction Capability Application Part* (TCAP) untuk menyediakan fungsi dari layer 4 sampai layer 6 OSI.

OSI
Layer 4 - 6
3
2
1

Seperti terlihat pada gambar 2.5 sistem CCS7 dibagi dalam blok-blok fungsional, yaitu :

1. *Message Transfer Part* (MTP)

   MTP berisi fungsi-fungsi dasar yang diinginkan untuk membawa pesan-pesan pada masing-masing level-nya. Bersama dengan SCCP menyediakan

fungsi tiga layer terendah mode OSI yaitu *phisical*, *data link* dan *network layer*.

2. *Signaling Connection Control Part* (SCCP)

   Bersama dengan MTP, SCCP menyediakan fungsi-fungsi *network* menggunakan layanan berupa *connectionless* dan *connection oriented* untuk membawa pesan-pesan. SCCP mengontrol hubungan lojik pada layanan *connection oriented*. SCCP juga melakukan perutean pesan-pesan dan melakukan *encoding* dan *decoding* untuk keperluan pemakai yang sesuai.

3. *Telephone User Part* (TUP)

   TUP menggunakan MTP untuk menyediakan fungsi-fungsi guna menunjang telepon. TUP menyediakan subset fungsional yang disediakan ISUP.

4. *Data User Part* (DUP)

   DUP mendefenisikan protokol untuk mengontrol sirkit *inter exchange*. DUP dipergunakan untuk data panggilan, data panggilan *registration* dan *cancellation*. seperti halnya TUP, DUP merupakan subset ISUP.

5. ISDN *User Part* (ISUP)

   ISUP merupakan *superset* dari TUP. Meskipun ISUP menggunakan *message set* yang berbeda dengan TUP tetapi seluruh aplikasi yang ditunjang oleh TUP dapat pula ditunjang oleh ISUP.

6. *Transaction Capability* (TC)

   TC berisi *Intermediate Service Part* (ISP) dan *Transaction Capability Aplication Part* (TCAP). TCAP menyediakan kemampuan pensinyalan yang cangfih berbasis *dialogue*. Sampai saat ini hanya layanan *connection less* yang didefenisikan ITU-T. TCAP dipergunakan untuk mendukung aplikasi IN dan GSM.

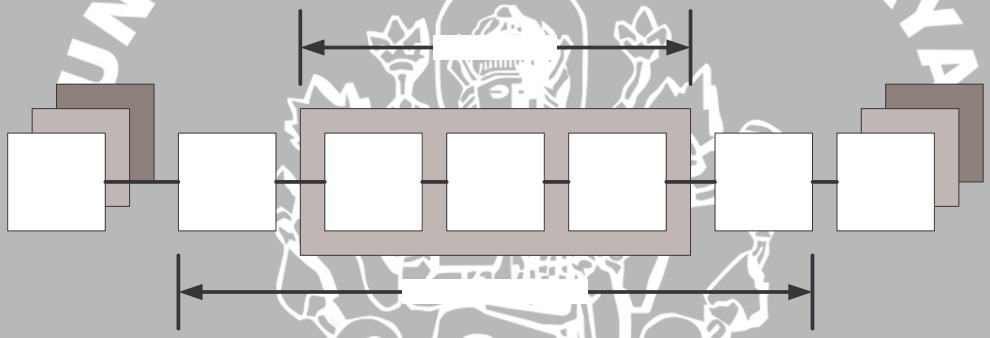
7. *Application Entities* (AEs) dan *Aplication Service Entities* (ASEs)

   AEs merupakan user dari CCS7 yang berfungsi menyediakan layanan komunikasi untuk aplikasi tertentu.

2.10.5 *Message Transfer Part* (MTP)

Teknologi komunikasi data paket yang dikenal dengan nama *High Level Data Link Control* (HDLC) merupakan dasar dari CCS7. Sasaran yang ingin dicapai dalam menyampaikan informasi adalah tanpa *loss*, duplikasi, bebas *error* dan urutan pesan sesuai dengan yang telah diatur. MTP bertanggung jawab atas pemindahan pesan diantara titik-titik pensinyalan. MTP dapat melakukan mekanisme pemindahan pesan, menyusun kembali pesan-pesan dan mengendalikan trafik sehingga dapat mengatasi kegagalan pada jaringan. Fungsi-fungsi MTP dibagi dalam tiga layer:

1. *Signaling Data Link Part* (Level 1)
2. *Signaling Link Function* (level 2)
3. *Signaling Network Functions* (Level 3)



Gambar 2.6 Struktur dasar CCS7 yang memerlukan pembagian fungsi MTP dan UP
Sumber: Cisco System

Gambar 2.6 memperlihatkan struktur dasar CCS7 yang memerlukan pembagian fungsi MTP dan beberapa *User Part* (UP) yang berbeda.

2.10.5.1 *Signaling Data Link* (MTP Level 1)

Signaling data link (SDL) adalah jalur transmisi dua arah (*bidirectional*) yang dipergunakan untuk pensinyalan. SDL mengandung dua kanal data yang dioperasikan bersama-sama dalam arah berlawanan pada kecepatan data yang sama.

Si

Usage Message Processing

Common Transfer Function

Link Control Function

2.10.5.2 *Signaling Link Function* (MTP Level 2)

Fungsi level ini menyediakan sebuah *frame network* untuk informasi yang akan dipindahkan pada setiap jalur, prosedur pendeteksian *error* dan perbaikan *error*.

Informasi pensinyalan yang akan dikirim merupakan *intra layer primitive* dalam komunikasi komputer. Informasi ini dibagi dalam beberapa unit sinyal dengan panjang bervariasi. Pada CCS7 *intra layer primitive* biasanya juga disebut juga *Basic Signal Unit* (SU). SU berisi *Signal Information Field* (SIF) dengan panjang informasi bervariasi dan nomor *field* dengan panjang informasi tetap dimana informasi yang dibawa diperlukan untuk mengontrol pesan yang dikirim. Terdapat tiga tipe dari SU, yaitu;

1. *Message Signaling Unit* (MSU)

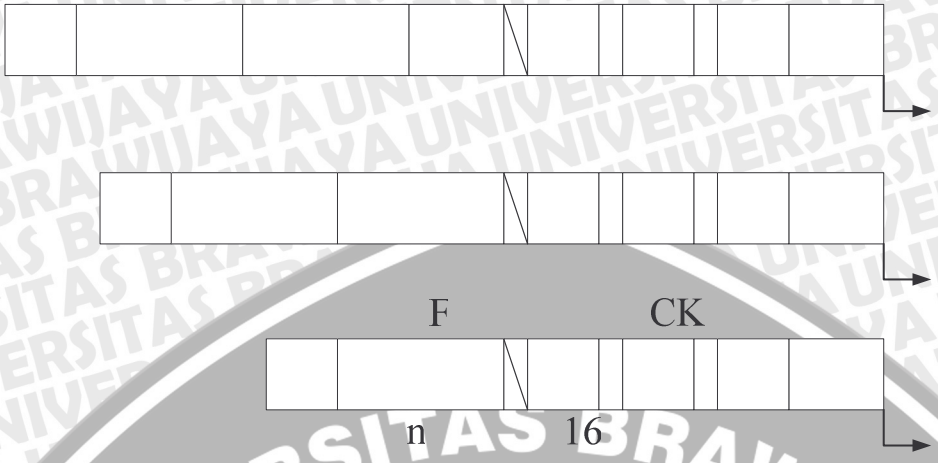
   Message signaling unit digunakan untuk membawa informasi pensinyalan baik berupa pengontrolan panggilan, manajemen jaringan, dan informasi perawatan dalam *signaling information field* (SIF).

2. *Fill in Signaling Unit* (FTSU)

   Lebih dikenal sebagai filter, biasanya dikirim bila tidak terdapat pengiriman MSU, LSSU, atau tidak terdapat alir informasi dari level 3 atau 4. Dengan demikian memungkinkan jaringan untuk menerima pemberitahuan kerusakan jalur pensinyalan dengan segera. FISU berfungsi untuk menjaga hubungan antara dua SP. FISU dikirim bila tidak ada pertukaran informasi sehingga FISU disebut juga dengan *supervisiory signaling*.

3. *Link Status Signaling Unit* (LSSU)

   Link status signalling unit digunakan untuk mengindikasikan suatu jalur ke *remote end signaling link*, Baik berupa indikasi normal, *out off alligment*, *out off service*, dan status darurat. Sinyal ini dikirim selama terjadi aktifitas jalur, pembenahan setelah kerusakan alur dan bila ditransfer informasi ke level 3 atau 4 tidak dapat dilaksanakan.



(a) Format Dasar Sebuah MSU

F CK SIF

n 16 8 atau 16

(b) Format LSSU

F CK

n 16

(c) Format FISU

BIB : Backward Indicator Bit
BSN : Backward Sequence Number
CK : Check Bit
F : Flag
FIB : Forward Indicator Bit

FS
LI
N
SF
SIO

Gambar 2.7 Format Sinyal Unit

Sumber: Cisco System

2.10.5.3 *Signaling Network Function* (MTP Level 3)

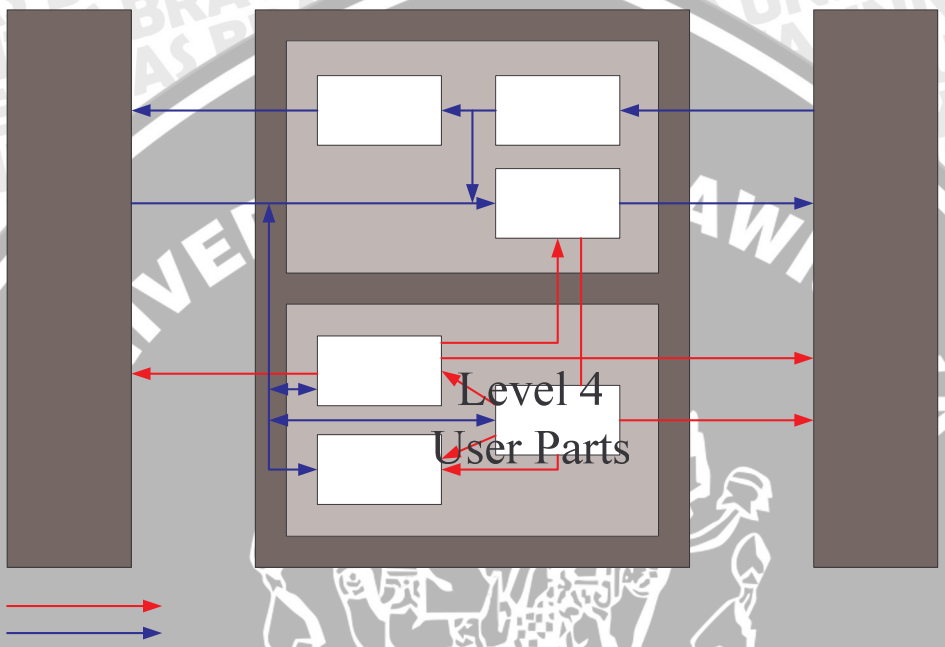
Signaling network fuinction didasarkan pada ITU-T Req.Q.704 dan diadaptasikan untuk berbagai jaringan pensinyalan. MTP level 3 berisi berbagai fungsi-fungsi dan prosedur-prosedur yang berhubungan dengan transfer pesan dari satu titik ke titik yang lain diantaranya:

1. *Signaling message handling.*
2. *Signaling network management.*
3. *Signaling network testing and maintenance.*

Untuk mentransfer informasi fungsi-fungsi tersebut ke level yang lain, digunakan sejumlah perintah *primitive* sebagai berikut:

1. *MTP- TRANSFER*, digunakan antara level 3 dan 4 untuk membangun mekanisme transport MTP.
2. *MTP-PAUSE*, digunakan untuk mengindikasikan ke pemakai (misal ISUP) bahwa *Point Code* tertentu tidak dapat diakses.
3. *MTP-RESUME*, digunakan untuk mengindikasikan ke *user* bahwa *point code* tertentu sudah dapat diakses.

4. *MTP-STATUS*, digunakan untuk mengindikasikan sebagian dari kemampuan untuk menyediakan layanan ke tujuan tidak dapat di bangkitkan. Juga dapat untuk mengindikasikan bahwa *remote user* tidak dapat dihubungi.



Gambar 2.8 Fungsi Jaringan Signaling (MTP Level 3)

Sumber: Cisco System

2.10.5.3.1 *Signaling Message Handling*

Signaling message handling seperti yang tertera pada gambar 2.8 terdiri atas 3 blok utama yang mempunyai fungsi sebagai berikut :

1. *Message Routing*, digunakan untuk menentukan *outgoing link* yang digunakan untuk mengirim pesan ke arah tujuannya.
2. *Message Discrimination*, digunakan untuk menentukan jika pesan tersebut untuk SP (pada saat pesan tersebut ditransfer ke *message distribution function*) atau jika pesan tersebut ditujukan ke tujuan lainnya (pada saat *message* tersebut ditransfer ke *message routing function*).
3. *Message Distribution*, digunakan untuk mengirim pesan untuk *user part* yang sesuai dengan fungsi level 3.

2.10.5.3.2 *Signaling Network Management*

Signaling network management menangani rekonfigurasi jaringan pensinyalan bila terjadi kerusakan dan untuk mengontrol trafik bila terjadi kemacetan. *Signaling network management* terdiri atas 3 fungsi utama:

1. *Signaling Traffic Management Function*, digunakan untuk membagi-bagi trafik pensinyalan dari jalur pensinyalan atau *link set* atau untuk mereduksi trafik bila terjadi kemacetan.
2. *Signaling Link Management Function*, digunakan untuk membenahi jalur pensinyalan yang rusak dan untuk memberikan informasi tentang jalur ke bagian lain MTP.
3. *Signaling Route Management Function*, digunakan untuk memberikan informasi tentang status set/ rute jalur pensinyalan antara SP atau STP, atau dapat dikatakan untuk memonitor status set/ rute jalur pensinyalan.

Untuk status yang dapat digunakan pada suatu *signaling route* terdiri dari :

1. *Unavailable* bila pesan *Tranfer Prohibit* telah diterima. *Transfer prohibit* mengindikasikan bahwa alir trafik menuju suatu tujuan tidak dapat ditransfer melalui STP yang mengirim pesan *transfer prohibit*.
2. *Available*, bila pesan *Transfer Allowed* telah diterima dari STP yang sebelumnya tidak dapat mentransfer pesan ke SP tertentu.
3. *Restricted*, mengindikasikan bahwa suatu STP mendapatkan kesulitan untuk mentransfer pesan ke tujuan tertentu.

2.10.5.3.2.1 *Signaling Trafik Management*

Digunakan untuk memastikan bahwa semua SP tujuan dapat diakses dengan memodifikasi pesan perutean, sebagai respon dari informasi yang didapatkan pada *signalling route management* dan *signaling link management*. *Signaling traffic managemet* terdiri dari beberapa prosedur :

1. *Changeover*

   Bila suatu jalur rusak maka *traffic management* akan melakukan prosedur *changeover* untuk mengalihkan trafik ke jalur lainnva dalam set jalur yang sama.

2. *Change back.*

   Bila suatu jalur rusak telah dapat digunakan kembali. Prosedur *change back* akan digunakan untuk mengembalikan jalur ke operasi normal. SP dapat melakukan prosedur ini dan akan mengirim pesan *Change Back Declaration* (CBD) ke SP lainnya melalui sembarang jalur yang dapat digunakan. CBD menyatakan bahwa jalur dapat digunakan dan prosedur *change back* dapat dilakukan.

3. *Forced Rerouting*

   Forced rerouting dilakukan pada saat menerima pesan TFP (*Transfer Prohibit*) yang mengindikasikan bahwa rute tersebut tidak dapat digunakan. Semua trafik ke tujuan tersebut akan dikirim pada suatu rute alternatif.

4. *Controlled Rerouting*

   Digunakan untuk mengembalikan trafik pensinyalan dari rute pensinyalan alternatif ke rute yang normal (sebenarnya). *Controlled rerouting* dilaksanakan oleh SP yang menerima pesan transfer *Allowed* (TFA).

5. *Management Inhibiting*

   Prosedur ini digunakan untuk membatasi penggunaan suatu jalur tertentu untuk pengetesan pesan dan perawatan dengan *command*.

6. *Signaling Traffik Flow Control*

   Berfungsi untuk membatasi trafik pada saat jaringan tidak dapat mentransfer semua trafik, yang disebabkan oleh kerusakan atau kemacetan jaringan. Bila tidak terdapat rute yang dapat digunakan ke suatu tujuan tertentu, MTP akan menggunakan perintah *primitive MTP-PAUSE* untuk memberitahukan ke *user part* lokal bahwa MTP tidak menangani pesan dari *user part* tersebut. Selanjutnya *user part* akan menghentikan trafik pesan ke tujuan tersebut.

2.10.5.3.2.2 *Signaling Link Management*

Fungsi ini digunakan untuk membenahi jalur rusak, untuk mengaktifkan jalur yang *idle* dan untuk menon-aktifkan jalur yang mengalami *alignment*.

2.10.5.3.2.3 *Signaling Route Management*

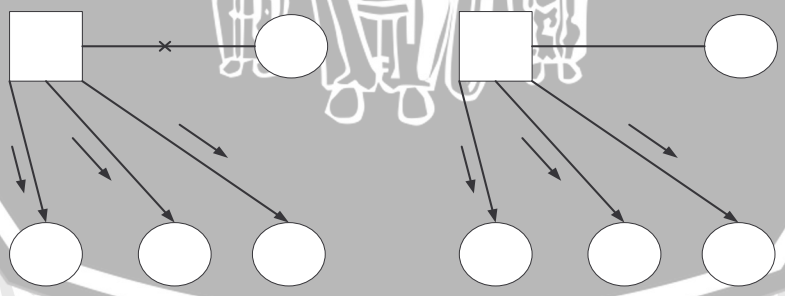
Signaling route management hanya berhubungan dengan operasi jaringan dengan mode *quasi-associated*, dengan demikian melingkupi fungsi STP untuk mengirim pesan yang sesuai melalui jaringan pensinyalan, sehingga memungkinkan SP dapat melaksanakan langkah yang sesuai. Prosedur *signaling route management* adalah sebagai berikut:

1. *Transfer Prohibit*

   Dilaksanakan pada suatu STP bila tidak ada rute trafik ke suatu SP yang dapat digunakan. STP tersebut akan mengirim pesan *Tranfer Prohibit* (TFP) melalui sembarang jalur yang dapat digunakan ke SP/ STP lainnya yang terhubung ke STP tersebut. Pesan tersebut berisi *Point Code* dari SP yang tidak dapat dihubungi. Pesan TFP hanya dikirim jika STP menerima pesan untuk ditransfer ke SP lain dan SP tersebut tidak dapat diakses. Bila suatu titik (*node*) menerima pesan TFP, titik tersebut akan melakukan prosedur *forced rerouting*, dan *MTP-PAUSE* dikirim ke level 4.

2. *Transfer allowed*

   Hal ini merupakan proses kebalikan dari TFP, dengan demikian tujuan yang tidak dapat diakses dapat diakses kembali dari suatu STP. STP mengirim pesan *Tranfer Allowed* (TFA) ke semua SP/STP yang terhubung ke titik *Point Code* yang dapat diakses kembali tersebut. Penerimaan TFA pada suatu SP/STP merupakan hasil implementasi prosedur *controlled rerouting* dan juga memungkinkan pembangkitan pesan TFA dari titik tersebut.



Gambar 2.9 Transfer Prohibit dan Transfer Allowed

Sumber: Cisco System

2.10.6 *Signaling Connection Control Part* (SCCP)

SCCP merupakan suatu *software* blok fungsional diatas MTP. Bersama-sama dengan MTP, SCCP membentuk *Network Service Par*t (NSP). Transfer data dilakukan dalam bentuk blok data yang disebut dengan *Network Service* Data Unit (NSDU). MTP menyediakan suatu mekanisme transfer data yang fleksibel dan sesuai dengan perubahan data antara titik pensinyalan.

2.10.7 ISDN *User Part* (ISUP)

User part (UP) terletak dilapisan tertinggi dari struktur lapisan CCS7. Pemakai dari *user part* adalah sentral yang menginginkan proses-proses tertentu. ISUP merupakan superset dari TUP dan DUP. Pada awalnya, UP hanya dipergunakan untuk layanan telepon yang dipenuhi oleh *Telephone User Part* (TUP). Dengan berkembangnya layanan-layanan baru yang berdasar atas komunikasi data, maka muncul lagi *Data User Part* (DUP). Sejalan dengan berkembangnya jaringan yang mengintregasikan suara dan data, maka diperlukan UP yang menunjang hal tersebut. *User part* itu disebut dengan ISDN *User Part* karena ide awal penerapannya pada jaringan ISDN. Pada prinsipnya ISUP mempunyai fungsi sebagai berikut:

1. Menyediakan pesan untuk pembangunan dan pembubaran hubungan antar SP
2. Menyediakan bentuk layanan untuk jasa ISDN.

2.10.8 Parameter Unjuk Kerja Jaringan CCS7

Ada tiga jenis kategori pada unjuk kerja jaringan pensinyalan, yaitu *avaibility*, *dependability*, dan *delay*.

2.10.8.1 *Avaibility*

Set perutean ditentukan oleh *unavaibility* dari komponen jaringan pensinyalan dan struktur jaringan *unavaibility* dari suatu jaringan didefinisikan sebagai probabilitas bahwa sistem dalam *state* gagal pada sebuah *random point* pada suatu saat, hal ini dapat diekspresikan pada sistem *Mean Time To Time*

Failure dan *Mean Time To Repair* dan sebagai *unavaibility* = MTTR/ (MTTF+MTTR).

2.10.8.2 *Dependability*

Dependability berhubungan dengan kemampuan jaringan untuk mengangkut pesan yang dapat diandalkan dan menyebabkan kegagalan pemakaian (*malfunction*). Pada MTP terdapat 4 tujuan:

1. *Undetected errors*: pada setiap jalur pensinyalan, tidak lebih dari satu dalam 10^{10} dari semua unit pensinyalan salah harus tidak dideteksi oleh MTP
2. *Lost Messages*: tidak lebih dari satu dari 10^{7} pesan harus hilang berhubungan dengan kegagalan pada MTP
3. *Messages out of sequence*: tidak lebih dari satu dalam 10^{10} pesan harus dikirim *out of sequence* ke UP berekenaan dengan kegagalan pada MTP hal ini termasuk penduplikatan pesan
4. *Transmission error rate*: SDL harus mempunyai bit *error* yang rendah yang tidak melebihi dari 10^{-6}.

Pada ISDN UP terdapat 2 tujuan:

1. *Probability of false operation*: tidak lebih dari satu dalam 10^{8} dari semua unit pensinyalan yang ditransmisikan baru diterima dan berkenaan dengan *error*, menyebabkan kesalahan operasi.
2. *Probability of signaling malfunction*: ketidakberhasilan panggilan dapat disebabkan oleh tidak dideteksinya kesalahan, kehilangan pesan-pesan, atau pengiriman pesan *out of sequence* tidak lebih dari dua dalam 10^{5} pada semua panggilan ISDN harus tidak berhasil berkenaan dengan *signal malfunction*, tidak lebih dari satu dalam 10^{5} dari semua koneksi sirkit ISDN harus tidak berhasil berkenaan dengan *signaling malfunction*.

2.10.8.3 *Delay*

Unjuk kerja *delay* merupakan atribut yang sangat penting dari layanan pensinyalan. Delay end to end merupakan delay antara node sumber (server) dan

node tujuan (client). Delay end to end dihitung dengan menggunakan persamaan 2.4.

$$t_{end\ to\ end} = t_{enc} + t_{trans} + t_{prop} + t_w + t_{dec} \quad (2.4)$$

Sumber: William Staling

Batas toleransi nilai delay end to end adalah 400 ms.

Dimana:

$t_{end\ to\ end}$ = *delay* end to end (sekon)

t_{enc} = *delay* enkapsulasi (sekon)

t_{trans} = *delay* transmisi (sekon)

t_{prop} = *delay* propagasi (sekon)

t_w = *delay* antrian (sekon)

t_{dec} = *delay* dekapsulasi (sekon)

Delay enkapsulasi dihitung dengan menggunakan persamaan 2.5.

$$t_{enc} = \frac{W_{frame} - MSS}{C_{pros}} x8 \quad (2.5)$$

Sumber: William Staling

Dimana:

t_{enc} = *delay* enkapsulasi (sekon)

W_{frame} = nilai panjang frame (byte)

MSS = *Maximum Segment Size* (byte/ paket)

C_{pros} = kapasitas media transmisi (bps)

D*elay* transmisi dihitung dengan menggunakan persamaan 2.6.

$$t_i = \frac{W_{frame}}{C} x8 \quad (2.6)$$

Sumber: William Staling

Dimana:

t_i = *delay* transmisi (sekon)

W_{frame} = nilai panjang frame (byte)

C = kapasitas media transmisi (bps)

Delay propagasi dihitung dengan menggunakan persamaan 2.7.

$$t_{prop} = t_{DTE} + t_{UTP} + t_{Router} + t_{backbone} + (\frac{d\sqrt{\mu_r x \varepsilon_r}}{c}) \quad (2.7)$$

$$= N_{DTE} x25x\frac{1}{C} + 0{,}556xP_{UTP}x\frac{1}{C} + N_{Router}xt_{Router} + t_{Backbone} + (\frac{d\sqrt{\mu_r x \varepsilon_r}}{c})$$

Sumber: William Staling

Dimana:

t_{prop} = *delay* propagasi (sekon)

t_{DTE} = *delay LAN card* (sekon)

t_{UTP} = *delay* kabel UTP (sekon)

t_{Router} = *delay* masing-masing *router* (sekon)

$t_{backbone}$ = *delay backbone* (sekon)

d = panjang kabel tembaga (meter)

μ_r = konstanta permeabilitas relative (tanpa satuan)

ε_r = konstanta dielektrik relative (tanpa satuan)

c = kecepatan cahaya (meter/ sekon)

N_{DTE} = jumlah LAN *card* (tanpa satuan)

C = kapasitas media transmisi (bps)

P_{UTP} = panjang kabel UTP (meter)

N_{Router} = jumlah *router* (tanpa satuan)

Delay antrian dihitung dengan menggunakan persamaan 2.8.

$$t_w = \frac{\lambda}{\mu(\mu-\lambda)} + \frac{1}{\mu} \qquad (2.8)$$

Sumber: William Staling

Dimana:

t_w = *delay* antrian (sekon)

λ = *throughput* (paket/ sekon)

μ = kecepatan pelayanan pelanggan (paket/ sekon)

1 = panjang paket rata-rata (byte/ paket)

Delay dekapsulasi dihitung dengan menggunakan persamaan 2.9.

$$t_{dec} = \frac{W_{frame} - MSS}{C_{pros2}} x8 \qquad (2.9)$$

Sumber: William Staling

Dimana:

t_{dec} = *delay* dekapsulasi (sekon)

W_{frame} = nilai panjang frame (byte)

MSS = *Maximum Segment Size* (byte/ paket)

C_{pros2} = kapasitas media transmisi (bps)

2.10.9 Teori Unjuk Kerja Sistem Pensinyalan No. 7

2.10.9.1 Umum

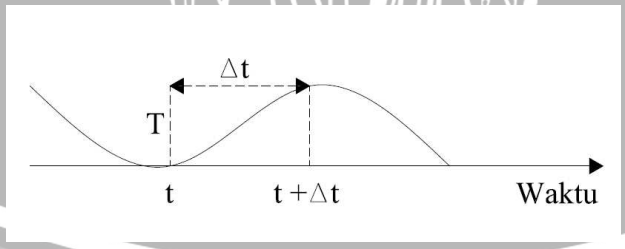
Ada beberapa model antrian yang digunakan dalam penanganan kedatangan pesan dalam satu atau beberapa server (prosesor) dimana prosesor tersebut hanya menyediakan *buffer* yang terbatas. Teori antrian ini muncul secara alami yang dapat digunakan untuk mempelajari proses panggilan, menganalisis hubungan pada suatu titik/ penyambungan sentral dan mempelajari model terhambatnya panggilan untuk sistem *blocking*. Isi dari *buffer* tersebut adalah pelanggan yang menunggu pelayanan, pelayanan paket-paket pada server dilakukan dengan disiplin pelayanan tertentu misalnya LIFO, FIFO dan FCFS.

Gambar 2.10 Sistem Antrian Sederhana dengan Sebuah Server
Sumber: Cisco System

2.10.9.2 Model Antrian M/M/1

2.10.9.2.1 Proses *Poisson*

Interval waktu yang digunakan dalam mendefinisikan proses *poisson*.



Gambar 2.11 Proses Poisson
Sumber: Cisco System

Proses kedatangan (*arrival proces*) adalah *memory less* (suatu kedatangan dalam suatu interval Δt tidak tergantung pada kejadian dalam interval sebelumnya dan sesudahya).

Buf

λ

Laju

P_1 {k kedatangan dalam T}= $p(k) = \frac{(\lambda T)^k e^{-\lambda T}}{k!}$ (2.10)

Distribusi *Poisson* k=0,1,2,…..

Sumber: William Staling

Dimana:

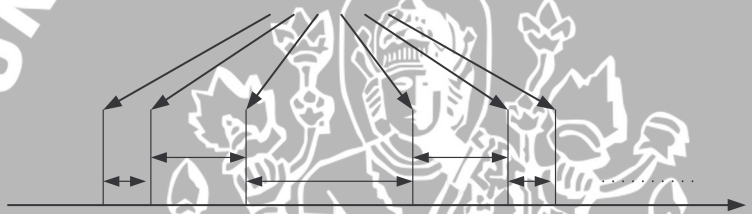
P_1 : peluang munculnya kedatangan pelanggan (tanpa satuan)

k : kedatangan palanggan (tanpa satuan)

λ : rate kedatangan pelanggan (pelanggan/ sekon)

T : interval waktu (sekon)

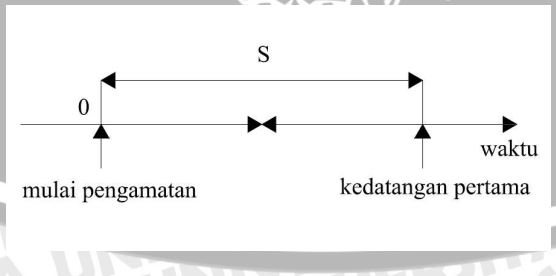
2.10.9.2.2 Kedatangan *Poisson*



Gambar 2.12 Proses Kedatangan Poisson
Sumber: Cisco System

Waktu kedatangan yang berurutan adalah variabel acak yang terdistribusi secara *continue*.

2.10.9.2.3 Distribusi Eksponensial



Gambar 2.13 Distribusi Eksponensial
Sumber: Cisco System

Waktu pela

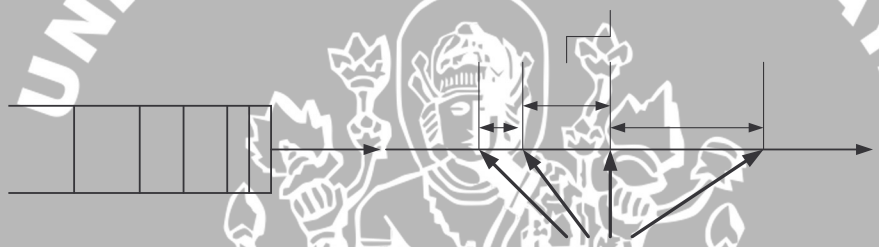
r2

r1 r3

τ adalah variabel acak yang menunjukan saat kedatangan, dihitung dari titik awal (*time origin*) tertentu.

2.10.9.2.4 Waktu Pelayanan disiplin pelayanan FIFO

Misal ada suatu antrian dari sejumlah pelanggan yang menunggu untuk mendapat pelayanan, τ adalah variabel acak yang menunjukkan jarak waktu antara panggilan yang datang berurutan dan mendapat pelayanan, r merupakan waktu pelayanan jika tidak ada waktu yang kosong (di dalam antrian selalu ada paket/ data/ *call* yang menunggu untuk dilayani, sehingga server tidak pernah menunggu masuknya panggilan baru). Sesuai dengan statiska *Poisson* maka r adalah variabel acak yang terdistribusi secara eksponensial negatif.



Gambar 2.14 Kedatangan Poisson
Sumber: Cisco System

p.d.f (probabilitas density function):

$$f_r(r) = \mu e^{-\mu r} \quad (2.11)$$

$$E(r) = 1/\mu \quad (2.12)$$

Sumber: William Staling

Dimana:

$f_r(r)$: peluang waktu pelayanan (tanpa satuans)

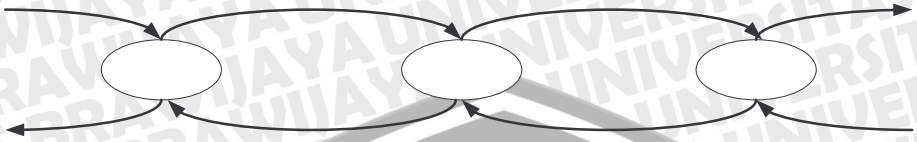
$E(r)$: waktu transmisi rata-rata (sekon)

μ : waktu transmisi (sekon)

r : waktu pelayanan (sekon)

Buffer

2.10.9.2.5 *State Diagram*



Gambar 2.15 State Diagram Untuk M/M/1
Sumber: Cisco System

Syarat kestabilan dalam system M/M/1 akan tercapai bila:

1. Jika $\lambda < \mu \rightarrow \rho = \lambda / \mu < 1$ sistem akan stabil
2. Jika $\lambda \geq \mu \rightarrow \rho = \lambda / \mu \geq 1$ sistem tidak stabil (akan terjadi kemacetan total)

Jika λ yang masuk ke server meningkat maka tingkat kedatangan meningkat sehingga terjadi peningkatan jumlah antrian di dalam *buffer*. Dalam kondisi dimana $\lambda \geq \mu$ maka server tidak mampu lagi mengimbangi laju kedatangan paket sehingga terjadi kemacetan total. Maka probabilitas *state*-nya untuk sistem yang stabil adalah:

$$P_k = (1-\rho)\rho^k \rightarrow \rho = \lambda / \mu < 1, k \geq 0 \qquad (2.13)$$

Sumber: William Staling

Dimana:

P_k : probabilitas state (tanpa satuan)

ρ : tingkat intensitas pelayanan (tanpa satuan)

k : kedatangan palanggan (tanpa satuan)

Solusi probabilitas kesetimbangan untuk model antrian M/M/1 jika kondisi $\lambda \geq \mu$ tercapai maka kondisi kesetimbangan tidak akan pernah tercapai.

2.10.9.2.6 Kepadatan Antrian yang Terbatas

Misal K adalah jumlah paket yang dapat ditampung oleh *queque buffer* maka:

1. *State* probabilitas untuk M/M/1 dengan *finite buffer* adalah:

$$P_k = \frac{(1-\rho)\rho^k}{1-\rho^{K+1}} \qquad (2.14)$$

Sumber: William Staling

Dimana:

P_k : probabilitas *state* (tanpa satuan)

ρ : tingkat intensitas pelayanan (tanpa satuan)

k : kedatangan palanggan (tanpa satuan)

Karena *buffer* antrian hanya dapat menampung K paket maka paket ke (K+1) dan seterusnya akan ditolak oleh sistem

2. Probabilitas *block up*

$$P_K = \frac{(1-\rho)\rho^K}{1-\rho^{K+1}} \qquad (2.15)$$

Sumber: William Staling

Dimana:

P_K : probabilitas *block up* (tanpa satuan)

ρ : tingkat intensitas pelayanan (tanpa satuan)

k : kedatangan palanggan (tanpa satuan)

3. Hubungan antara *throughput* dan beban adalah sebagai berikut:

Gambar 2.16 Antrian yang menunjukkan hubungan antara throughput dan beban
Sumber: Cisco System

P_b = perbandingan antara λ_B dengan λ (*loss* dengan total kedatangan).

2.10.9.3 Prosedur Pensinyalan

2.10.9.3.1 Umum

CCS7 menyediakan prosedur penyaluran informasi menggunakan sejumlah data *link* dalam interkoneksi antar dua atau lebih titik. Dalam aplikasinya, proses yang terjadi adalah pertukaran informasi antara dua prosesor dalam jaringan

telekomunikasi. CCS7 menyediakan semua fungsi-fungsi *interworking* baik untuk protokol DSS-1 ke sisi terminal pada jaringan ISDN dan untuk *Dual Tone Multi Frequency* (DTMF) pada terminal PSTN, maupun untuk pensinyalan non CCS7 (misal R2) ke sisi sentral non CCS7.

2.10.9.3.2 Metode Pensinyalan

Terdapat dua macam metode pensinyalan untuk hubungan antar sentral, melalui beberapa sentral transit yaitu;

1. Pensinyalan *End to End*

   Pada metode ini sentral asal mengirimkan informasi pensinyalan ke sentral transit untuk menginformasikan permintaan hubungan ke sentral tujuan. Pada CCS7, metode ini tidak digunakan dalam prosedur pembangunan hubungan, hanya digunakan untuk menangani *supplementary service* atau pensinyalan *user to user*.
2. Pensinyalan *Link by link*.

   Prosedur pensinyalan pada jaringan CCS7 hanya memungkinkan menggunakan metode *link by link*. Informasi permintaan hubungan dikirim oleh sentral asal ke sentral transit, selanjutnya sentral transit mengirimkan informasi tersebut sampai ke sentral tujuan.

2.10.9.3.3 Tipe Pesan

Pesan-pesan yang terdapat pada ISUP diperlukan untuk pembangunan hubungan, pembubaran hubungan dan keperluan lain termasuk fasilitas untuk *supplements services*. Secara umum pesan-pesan ISUP tersebut dapat dikelompokan, diantaranya adalah:

1. Pesan-pesan pembangunan hubungan (*Call Setup Messages*)
2. Pesan-pesan pembubaran hubungan (*Call Release Messages*);
3. Pesan-pesan pengawasan sirkit (*Circuit Supervision Messages*)

ISUP Pesan (*Message*)	Uraian
***Initial Address* (IAM)**	Pesan yang dikirim ke arah *forward* untuk mengawali pendudukan suatu sirkit *outgoing* , berisi informasi digit dan informasi lainnya yang berkenaan dengan perutean dan penanganan panggilan.
***Address Complete* (ACM)**	Pesan yang dikirim kearah *backward* menyusul dikirimnya pesan IAM, untuk membawa tambahan informasi yang masih diperlukan untuk hubungan yang diinginkan.
***Answer* (ANM)**	Pesan yang dikirim ke arah *backward* menaandakan bahwa panggilan telah dijawab oleh pelanggan yang dipanggil.

Tabel 2.2 Call Setup Message
Sumber: Fred Hashal

***Release* (REL)**	Pesan yang dikirim ke arah *forward* dan/atau *backward* sebagai indikasi bahwa sirkit yang sedang tergenggam dilepaskan. Pesan ini juga berisi tentang penyebab kenapa sirkit tersebut dilepaskan, dan menjadikannya pada status *idle* (setelah menerima pesan RLC).
***Release Complete* (RLC)**	Pesan yang dikirim arah *forward* dan/atau *backward* sebagai balasan pesan REL, atau RSC jika statusnya sesuai. Dalam hal sirkit yang bersangkutan akan dibawa ke status *idle* .

Tabel 2.3 Call Release Message
Sumber: Fred Hashal

***Reset Circuit* (RSC)**	Pesan yang dikirim untuk melepaskan suatu sirkit bila pesan yang seharusnya diterima tidak datang.

Tabel 2.4 Circuit Supervision Message
Sumber: Fred Hashal

2.10.9.3.4 Urutan Proses Pensinyalan

Fungsi utama pensinyalan CCS7 adalah untuk membangun hubungan kanal informasi 64 Kbps untuk suara serta mendukung *supplementary service* yang disediakan oleh kedua pemakai. Pensinyalan untuk pembangunan hubungan dilakukan melalui hubungan pensinyalan secara permanen sedangkan pensinyalan untuk mengaktifkan fasilitas *supplementary service* dilakukan melalui hubungan pensinyalan *end-to-end* secara sementara.

2.10.9.3.4.1 Proses Pembangunan Hubungan

Dalam proses pembangunan hubungan terdapat dua alternatif:

1. Pembangunan hubungan yang berhasil

1. Sentral asal menerima informasi (*off-hook* dan *dialing number*) dari pemakai, sentral asal selanjutnya merubah informasi tersebut ke dalam format pesan ISUP IAM serta mengirimkannya ke sentral transit.
2. Sentral transit setelah menerima IAM merubah pesan menjadi satu informasi (*ringing tone*) ke terminal tujuan. Pada saat yang sama sentral transit mengirim ACM sebagai tanda telah diterimanya IAM. Selanjutnya sentral transit mengirim pesan *ringing back tone* ke sentral asal.
3. Pada saat terminal tujuan menjawab (*off-hook*), sentral transit menerima informasi tersebut, mengubahnya ke dalam pesan ANM dan mengirimkan pesan tersebut ke sentral asal.
4. Setelah menerima ANM dari sentral transit, sentral asal memberitahukan ke terminal asal sebagai indikasi bahwa hubungan kanal informasi telah siap dilaksanakan.

2. Pembangunan hubungan yang tidak berhasil

   Terdapat berbagai macam penyebab ketidakberhasilan pembangunan hubungan, antara lain disebabkan oleh kongesti pada jaringan dan pelanggan yang dipanggil sibuk.

   1. Ketidakberhasilan karena kongesti pada jaringan, bila:

      Setelah menerima IAM dan terdapat kongesti maka IAM tersebut akan dibalas dengan pesan REL. Sentral yang menerima pesan REL akan mengirimkan pesan RLC untuk membebaskan jalur pensinyalan dan sirkit yang akan diduduki. *Tone* kongesti akan dikirim sentral asal ke terminal asal.
   2. Ketidak berhasilan karena pelanggan sibuk (*user busy*)

      Bila sentral asal menerima informasi sibuk dari sentral transit, maka sentral asal akan mengirim pesan REL ke sentral transit dan sentral transit membalasnya dengan pesan RLC untuk membebaskan jalur pensinyalan dan sirkit yang diduduki. Nada sibuk juga akan dikirim ke terminal asal dari sentral asal.

2.10.9.3.4.2 Proses Pemutusan Hubungan

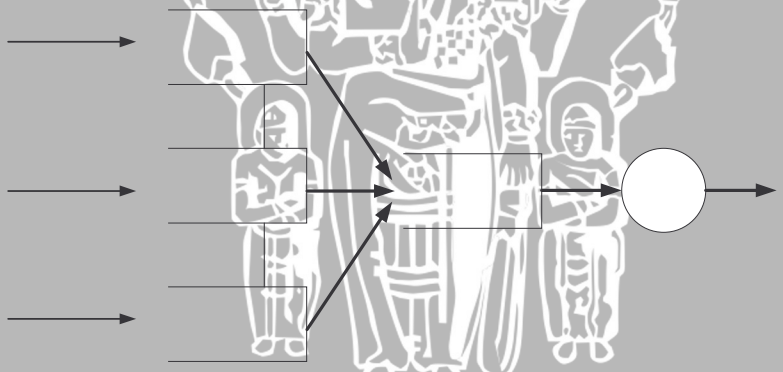
Dalam proses pemutusan hubungan terdapat dua alternatif tergantung kepada yang menutup hubungan terlebih dahulu, yaitu yang memanggil atau yang dipangil.

1. Pemanggil menutup hubungan, dan
2. Pelanggan yang dipanggil menutup hubungan.

Bila pemanggil lebih dahulu menutup hubungan (*on-hook*). Sentral asal akan mengirim pesan REL, dan sentral transit akan mengirim pesan RLC. Dalam hal pelanggan yang dipanggil lebih dahulu menutup hubungan (*on-hook*), sentral transit akan mengirimkan pesan REL terlebih dahulu, dan sentral asal akan mengirim pesan RLC.

2.10.9.4 Deskripsi STP

Pada gambar 2.17 dapat dilihat arsitektur STP dengan sebuah pusat pemrosesan untuk melayani pesan-pesan yang datang ke STP.



Gambar 2.17 Arsitektur STP dengan 1 buah prosesor

Sumber: Cisco System

Pesan-pesan tersebut diterima oleh prosesor tingkat 2. Prosesor ini bertanggung jawab atas pengecekan *Cyclic Redudancy Check* (CRC), pengiriman *acknowledgement*, menyimpan *track* dari pewaktuan pada tingkat 2. Semua pesan yang telah melewati prosesor tingkat 2 akan menuju sebuah prosesor tingkat 3 (biasanya melalui beberapa bentuk bus komunikasi internal) yang bertanggung jawab atas penyelenggaraan perutean dari pesan-pesan tersebut ke *outgoing link*.

Level 2 Recieve

SS7 Links Buffer

Prosesor tingkat 3 menempatkan pesan-pesan tersebut ke dalam *buffer*, dan dilayani dengan disiplin pelayanan FIFO. Berdasarkan atas jumlah kedatangan pesan pada *buffer* perutean, prosesor tingkat 3 akan memutuskan untuk menerima pesan-pesan tersebut atau membuangnya segera (*Message Discard*), berdasarkan prioritas yang terdapat pada setiap pesan. Pada aplikasinya, STP tidak mengevaluasi pembangunan pangilan yang berhubungan dengan pesan untuk menentukan tipe pesan tersebut, tetapi lebih sering merutekannya ke tujuan dari pesan-pesan tersebut. Sebagai bagian dari prosedur pengontrolan kemacetan tingkat 2, prosesor tingkat 3 harus menguji prioritas setiap pesan untuk menentukan apakah status *current route congestion* mengijinkannva untuk ditransfer.

2.10.9.5 Skema Alir Pembangunan Hubungan CCS7

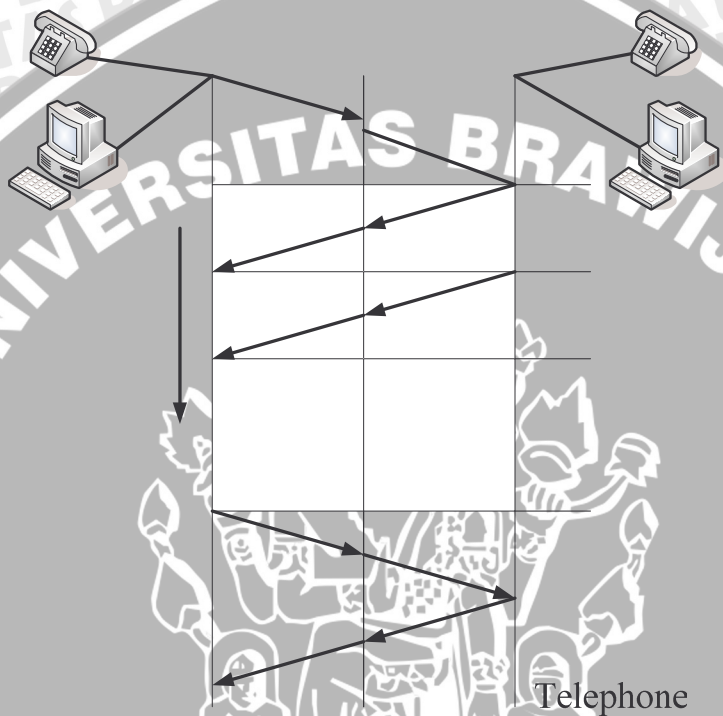
Pada bagian ini, dijabarkan pembangunan hubungan menggunakan jaringan CCS7 dan tindakan *recoverv* terhadap pembangunan hubungan yang diambil ketika pesan jatuh (*droped*). Dengan asumsi sentral yang melayani *Called Party* sebagai sentral asal dan sentral yang melayani *called party* sebagai sentral yang dituju. Penggunaan istilah ini tanpa memperhatikan sentral yang mana yang sebenarnya mengirim atau menerima pesan CCS7 yang terliput dalam *call setup* atau pemutusan (*teardown*).

2.10.9.5.1 Skema Alir Pesan dan Prioritas Pesan

Sebuah panggilan lengkap yang normal akan meliputi lima atau enam *message* tergantung pada siapa terlebih dahulu yang *hangs-up* (menutup/ mengakhiri hubungan) (gambar 2.18 menunjukan diagram aliran *message*). Kami mengasumsikan bahwa semua panggilan akan ke *line* yang kosong dan dijawab. Sebagai contoh dengan memasukkan juga panggilan-panggilan ke jalur sibuk, tetapi dengan mempelajari kasus ini secara lengkap akan lebih mudah untuk menginpretasikan bentuk layanan dari protokol dan lokasinya yang mempengaruhi unjuk kerja CCS7.

Ketika sentral asal menyeleksi *trunk* dengan lebih mengarahkan ke sentral tujuan daripada melalui sentral asal akan membuat sebuah sinyal *Initial Address Message* (IAM) yang kemudian mengirimkannya ke sentral tujuan melalui STP.

Pesan ini berisi informasi pembangunan hubungan panggilan seperti nomor pemanggilan yang dipanggil dan kode pengindetifikasian sirkit. Ketika sentral tujuan menerima IAM, maka sentral tujuan akan mengirim balik sinyal *Address Complete Message* (ACM), yang pada intinya mengkonfirmasikan pemilihan sirkit. Sentral tujuan kemudian menghidupkan jalur daya (*power ring*) ke *called party's phone*.



Gambar 2.18 Diagram Alir Call Setup untuk Panggilan yang Dijawab
Sumber: Fred Hashal

Setelah ada jawaban sentral tujuan akan mengirimkan sebuah sinyal *Answer Message* (ANM). Pada saat itu jalur *forward speech* disambung dan percakapan dapat dimulai. Pewaktuan pada penyambungan jalur *forward speech* dapat bervariasi tergantung dari vendor dan penerapannya. Pada *call completion*, skenario berikut dapat terjadi yaitu :

1. Jika *party* asal memutuskan hubungan pertama kali maka *Release Message* (REL) dikirim ke sentral tujuan dan *Release Complete* (RLC) dikirim balik kesentral asal. Prosedur ini akan membebaskan pesanan sirkit, oleh karenanya sirkit dapat digunakan untuk panggilan yang lain.

2. Jika *party* yang dipanggil memutuskan hubungan terlebih dahulu, maka sebuah sinyal SUS (*Suspend Message*) dikirim dari sentral tujuan ke sentral asal. Maka sentral asal akan mengrimkan pesan REL dan sentral tujuan akan merespon dengan pesan RLC, membebaskan *trunk* dan menyelesaikan pemrosesan.

2.10.9.5.2 Tindakan *Call Setup Recovery*

Pada bagian ini, dijabarkan tentang tindakan yang akan diambil oleh sentral ketika pesan dijatuhkan oleh STP. Ketika sebuah sinyal IAM dikirim oleh sentral asal, maka pewaktuan TIAM (20-30 detik) mulai aktif untuk panggilan tersebut. Jika pewaktuan ini habis sebelum menerima pesan ACM ataupun ANM, maka sentral asal akan memutuskan bahwa pengulangan panggilan telah gagal, menghubungkan pelanggan ke *service announcement*, dan prosedur pembebasan sirkit dijalankan, ini akan terjadi ketika pesan IAM jatuh (*dropped*) oleh STP.

Jika pesan ACM hilang dan *called party* mengambil lebih dari TIAM detik untuk menjawab panggilan, sentral asal akan mengirim *calling party* untuk menugaskan kembali dan membebaskan sirkit, dan kemungkinan akan terjadi pemanggilan kembali. Jika pesan ACM jatuh oleh STP, maka tidak ada tindakan *recovery* yang diambil oleh sentral dan jika pesan ANM jatuh oleh STP maka jalur *Forward Speech* tidak akan disambungkan. Ini akan dikenal oleh parties (sehingga yang satu akan mendengar percakapan dan yang lainnya tidak), dan keduanya akan menutup telepon. Prosedur pembebasan sirkit kemudian dijalankan.

Ketika sebuah sinyal REL dikirim oleh sentral asal, maka pewaktuan TREL (4-6 detik) dan TREL (60 detik) akan mulai aktif. Jika pewaktuan TREL habis sebelum pengirimaan pesan RLC diterima, sebuah pesan REL yang baru dan pewaktuan akan mulai diaktifkan kembali. Prosedur ini diulang sampai pewaktuan TREL habis. Pada saat itu, sirkit diletakan pada sirkit reset list dan sebuah pesan *reset circuit* (RSC) dikirim setiap TRSC detik sampai sebuah pesan RLC yang dikenal diterima. Sampai pesan RLC diterima, saat ini tidak dapat digunakan untuk sebuah panggilan ulang baru oleh sentral asal. Aksi *recovery* ini diambil ketika pesan REL/RSC atau pesan RLC yang dikenal dibuang (*discard*) oleh STP.

2.11 *Visual Basic.NET*

Microsoft Visual Basic.NET adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak di atas sistem *.NET Framework*, dengan menggunakan bahasa *BASIC*. Dengan menggunakan alat ini, para programmer dapat membangun aplikasi *Windows Forms*, Aplikasi web berbasis ASP.NET, dan juga aplikasi *command-line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti *Microsoft Visual C++, Visual C#, atau Visual J#*), atau juga dapat diperoleh secara terpadu dalam *Microsoft Visual Studio .NET*. Bahasa *Visual Basic .NET* sendiri menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari *Microsoft Visual Basic* versi sebelumnya yang diimplementasikan di atas *.NET Framework*. Peluncurannya mengundang kontroversi, mengingat banyak sekali perubahan yang dilakukan oleh *Microsoft*, dan versi baru ini tidak kompatibel dengan versi terdahulu.

2.11.1. Versi

Terdapat empat buah versi *Visual Basic* yang dirilis hingga bulan November 2007, yakni:

2.11.1.1. *Visual Basic .NET* 2002 (VB 7.0)

Versi pertama dari *Visual Basic .NET* adalah *Visual Basic .NET* 2002 yang dirilis pertama kali pada bulan Februari 2002. *Visual Basic .NET* 2002 merupakan sebuah bahasa pemrograman visual yang berbasis bahasa *BASIC* (sama seperti halnya *Visual Basic* 6.0, tetapi lebih disempurnakan dan lebih berorientasi objek), dan didesain untuk berjalan di atas *Microsoft .NET Framework* versi 1.0.

Versi 7.0 ini dirilis bersamaan dengan *Visual C#* dan *ASP.NET*. Bahasa C# yang dianggap sebagai jawaban terhadap *Java*, mendapatkan perhatian yang lebih banyak dibandingkan dengan *VB.NET* yang kurang begitu banyak diulas. Hasilnya, sedikit orang di luar komunitas *Visual Basic* yang memperhatikan *VB.NET*. Versi pertama ini kurang mendapat sambutan yang bagus dari para *programmer*, dan pada saat itu, program berbasis *Visual Basic* 6.0 sedang marak-maraknya dibuat. Para *programmer* yang mencoba *Visual Basic .NET* untuk pertama kali akan merasakan bahwa *Visual Basic .NET* sangatlah berbeda dibandingkan dengan *Visual Basic*

sebelumnya. Contoh yang paling mudah adalah *runtime engine* yang lebih besar 10 kali lipat dibandingkan *Visual Basic* 6.0, dan juga meningkatkan beban di memori.

2.11.1.2. *Visual Basic .NET* 2003 (VB 7.1)

Selanjutnya, pada bulan Maret 2003, *Microsoft* pun merilis lagi versi yang lebih baru dari *Visual Basic .NET, Visual Basic .NET* 2003. Versi ini berisi beberapa perbaikan dibandingkan dengan versi sebelumnya, dan aplikasi yang dibuatnya dapat berjalan di atas *.NET Framework* versi 1.1. Fitur yang ditambahkan adalah dukungan terhadap *.NET Compact Framework* dan mesin *wizard upgrade* VB 6 ke *VB.NET* yang telah ditingkatkan. Peningkatan yang lainnya adalah peningkatan pada performa dan keandalan dari *Integrated Development Environment* (IDE) *Visual Basic* itu sendiri, dan juga *runtime engine*.

Visual Basic .NET 2003 tersedia dalam beberapa jenis cita rasa: *Professional, Enterprise Architect* dan *Academic Edition*. Khusus untuk *Visual Basic .NET* 2003 *Academic Edition*, versi tersebut didistribusikan secara gratis untuk beberapa sekolah di dalam setiap negara; versi *Professional* dan *Enterprise Architect* merupakan produk komersial.

2.11.1.3. *Visual Basic* 2005 (VB 8.0)

Setelah itu, *Microsoft* pun berkonsentrasi dalam mengembangkan *Microsoft .NET Framework* 2.0, dan tentunya alat bantu untuk membangun program di atasnya. Hingga pada tahun 2005, mereka pun merilis versi terbaru dari *Visual Basic .NET*, yang kali ini disebut dengan *Visual Basic* 2005 (dengan membuang kata "*.NET*"), bersama-sama dengan beberapa aplikasi pengembangan lainnya.

Untuk rilis 2005 ini, Microsoft menambahkan beberapa fitur baru, di antaranya adalah:

1. *Edit and Continue*

Fitur ini sebelumnya terdapat di dalam *Visual Basic*, akan tetapi dihapus di dalam *Visual Basic .NET*. Dengan keberadaan fitur ini, para *programmer* dapat memodifikasi kode pada saat program dieksekusi dan melanjutkan proses eksekusi dengan kode yang telah dimodifikasi tersebut.

2. Evaluasi ekspresi pada saat waktu desain

3. Munculnya *Pseudo-Namespace* "*My*", yang menyediakan:
    1. Akses yang mudah terhadap beberapa area tertentu dari dalam *.NET*
    2. *Framework* yang tanpanya membutuhkan kode yang sangat signifikan.
    3. Kelas-kelas yang dibuat secara dinamis (khususnya *My.Forms*).
4. Peningkatan yang dilakukan terhadap konverter kode sumber dari *Visual Basic* ke *Visual Basic .NET*.
5. Penggunaan kata kunci (*keyword*) *Using*, yang menyederhanakan penggunaan objek-objek yang membutuhkan pola *Dispose* untuk membebaskan sumber daya yang sudah tidak terpakai.
6. *Just My Code*, yang menyembunyikan kode *reusable* yang ditulis oleh alat bantu *integrated development environment* (IDE) *Visual Studio .NET*.
7. Pengikatan sumber data (*Data Source Binding*), yang mampu mempermudah pengembangan aplikasi basis data berbasis klien/ server.

Fungsi-fungsi yang tersebut di atas (khususnya *My*) ditujukan untuk memfokuskan *Visual Basic .NET* sebagai sebuah *platform* pengembangan aplikasi secara cepat dan "menjauhkannya" dari bahasa C#.

Bahasa Visual Basic 2005 memperkenalkan fitur-fitur baru, yakni:

1. Bawaan *.NET Framework* 2.0:
    1. *Generics*
    2. *Partial class*, sebuah metode yang dapat digunakan untuk mendefinisikan beberapa bagian dari sebuah kelas di dalam sebuah berkas, lalu menambahkan definisinya di lain waktu, sangat berguna khususnya ketika mengintegrasikan kode pengguna dengan kode yang dibuat secara otomatis.
    3. *Nullable Type*
2. Komentar XML yang dapat diproses dengan menggunakan beberapa alat bantu seperti NDoc untuk membuat dokumentasi secara otomatis.
3. Operator *overloading*
4. Dukungan terhadap tipe data bilangan bulat tak bertanda (*unsigned integer*) yang umumnya digunakan di dalam bahasa lainnya.

2.11.1.4. *Visual Basic* 9.0 (*Visual Basic* 2008)

Versi ini merupakan versi terbaru yang dirilis oleh *Microsoft* pada tanggal 19 November 2007, bersamaan dengan dirilisnya *Microsoft Visual* C# 2008, *Microsoft Visual* C++ 2008, dan *Microsoft .NET Framework* 3.5.

Dalam versi ini, *Microsoft* menambahkan banyak fitur baru, termasuk di antaranya adalah:

1. *Operator If* sekarang merupakan *operator ternary* (membutuhkan tiga *operand*), dengan sintaksis *If* (*boolean*, nilai, nilai). Ini dimaksudkan untuk mengganti fungsi *IF*.
2. Dukungan *anonymous types*
3. Dukungan terhadap *Language Integrated Query* (LINQ)
4. Dukungan terhadap ekspresi Lambda
5. Dukungan terhadap literal XML
6. Dukungan terhadap inferensi tipe data.

2.11.2. Hubungan dengan *Visual Basic* klasik

Apakah *Visual Basic .NET* dianggap sebagai sebuah versi *Visual Basic* atau benar-benar bahasa yang berbeda merupakan sebuah topik perdebatan yang hangat. Hal ini dikarenakan sintaksis bahasa *Visual Basic .NET* tidak mengalami perubahan yang sangat drastis, dan hanya menambahkan beberapa dukungan fitur baru seperti penanganan eksepsi secara terstruktur dan ekspresi yang bisa di-*short-circuit*-kan. Dua perubahan tipe data pun terjadi saat berpindah ke *Visual Basic .NET*. Dibandingkan dengan *Visual Basic* 6.0, tipe data Integer yang dimiliki oleh *Visual Basic .NET* memiliki panjang dua kali lebih panjang, dari 16 bit menjadi 32 bit. Selain itu, tipe data *Long* juga sama-sama berubah menjadi dua kali lipat lebih panjang, dari 32 bit menjadi 64 bit. Bilangan bulat 16-bit dalam *Visual Basic .NET* dinamakan dengan *Short*. Lagi pula, desainer GUI *Windows Forms* yang terdapat di dalam *Visual Studio .NET* atau *Visual Basic .NET* memiliki gaya yang sangat mirip dengan editor *form Visual Basic* klasik.

Jika sintaksis tidak banyak yang berubah, lain halnya dengan semantik, yang berubah secara signifikan. *Visual Basic .NET* merupakan sebuah bahasa pemrograman yang mendukung fitur "Bahasa Pemrograman Berorientasi Objek"

secara penuh, karena memang didukung oleh arsitektur *Microsoft .NET Framework*, yang mengandung kombinasi dari *Common Language Runtime* dan *Base Class Library*. *Visual Basic* klasik, hanya merupakan sebuah bahasa pemrogaman berbasis objek, yang berjalan di atas arsitektur *Component Object Model* (COM).

Perubahan ini telah mengubah banyak asumsi tentang hal yang benar yang harus dilakukan dengan mempertimbangkan performa dan kemudahan untuk dipelihara. Beberapa fungsi dan pustaka perangkat lunak, yang ada di dalam *Visual Basic* klasik, kini tidak terdapat di dalam *Visual Basic .NET* mungkin masih banyak yang masih terdapat di dalam Visual Basic *.NET*, tapi tidak seefisien apa yang ditawarkan oleh *.NET Framework*. Bahkan jika program *Visual Basic* klasik bisa dikompilasi dengan benar, sebagian besar program *Visual Basic* klasik harus melalui beberapa proses *refactoring* untuk mengadopsi fitur bahasa baru secara keseluruhan.