

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Umum

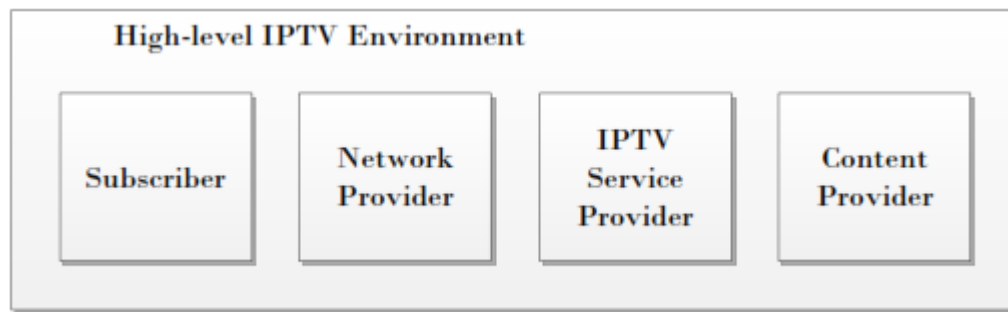
IPTV (*Internet Protocol Television*) diimplementasikan melalui teknologi *wireless Worldwide Interoperability for Microwave Access* (WiMAX). WiMAX merupakan salah satu teknologi nirkabel masa depan. Digelar sebagai generasi ke empat dan bersaing dengan *long term evolution* (4G LTE). Teknologi ini mampu membawa, data, *video* dan *voice* dengan kecepatan tinggi yaitu kecepatan data hingga 70 Mbps dan lebar kanal sebesar 20 MHz. Teknologi ini terus dikembangkan oleh *Institute of Electrical and Electronics Engineers* (IEEE) pada 802.16 - 2014. IEEE 802.16d yang dikenal sebagai *Fixed WiMAX* (WiMAX Tetap). Jaringan WiMAX 802.16d menyediakan kecepatan data tinggi hingga 70 Mbps, *bandwidth* yang lebar dengan rentang spektrum frekuensi 2 GHz sampai dengan 11 GHz.

#### 2.2. *Internet Protocol Television* (IPTV)

Perkembangan teknologi Internet memungkinkan transmisi data, suara, dan video format digital dapat dikirimkan melalui protokol internet. Dengan cara yang konvensional, untuk mendapatkan ketiga jenis data digital tersebut pengguna diharuskan mengakses ke alamat *website* tertentu dan menikmati data digital tersebut melalui antarmuka yang disediakan oleh *website* tersebut. Dalam pengembangan dan penggabungan beberapa jenis teknologi ke dalam satu medium transmisi mengakibatkan muncul sebuah istilah *Television over Internet Protocol - IPTV* (Donoso, 2009). Konsep IPTV yang berkembang saat ini adalah bagaimana sebuah konten TV dalam format digital ditransmisikan melalui medium IP pada *private network* yang bersifat *closed distribution network*, yang hanya bisa diakses oleh pelanggan yang terdaftar pada sebuah penyedia layanan yang sudah memiliki infrastruktur yang memadai untuk menyelenggarakan layanan IPTV tersebut, dan dijaga kualitas keamanan, keandalan, dan keandalannya secara profesional dan mengacu pada standar layanan yang berlaku. Konten TV digital ini dapat dimodifikasi sesuai kebutuhan dari pelanggan, sehingga tayangan yang tersedia bisa bervariasi dan berbeda antar setiap pelanggan.

### 2.2.1. Arsitektur

Arsitektur teknologi IPTV terdiri dari empat komponen, yaitu *Subscriber*, *Network Provider*, *IPTV Service Provider*, *Content Provider*. Masing-masing komponen memiliki fungsionalitas pelayanan yang saling terintegrasi untuk menjamin keberhasilan pengiriman paket-paket data digital melalui protokol Internet.



Gambar 2.1 *IPTV Environment*  
(Sumber: Uke Kurniawan, 2010)

#### 2.2.1.1. Content Provider

Komponen pertama dimulai dari Content Provider, bertugas dalam penyediaan konten yang akan dikirimkan secara eksklusif kepada pelanggan dan bersifat off-air. Content Provider bertanggung jawab dalam penyediaan konten yang akan diberikan kepada pelanggan melalui penyedia layanan IPTV untuk mendistribusikan dan mengontrol informasi dari konten tersebut.

#### 2.2.1.2. IPTV Service Provider

Komponen kedua dibentuk oleh penyedia layanan IPTV, yang bertanggung jawab dalam mengelola dan mengubah konten digital menjadi paket-paket IP dan mengirimkan paket tersebut kepada pelanggan melalui penyedia infrastruktur jaringan. Penyedia layanan IPTV melakukan perjanjian dengan pemilik konten (*content owners*), mengendalikan enkripsi konten yang dikirimkan untuk mencegah akses dari pihak yang tidak berkepentingan, menyetujui untuk menyertakan proteksi *Digital Rights Management* (DRM) untuk produk-produk premium dengan tujuan menghindari proses reproduksi, pemutaran kembali (replay), dan penyimpanan.

Komponen *Head End* menerima berbagai jenis format dan media sebagai sumber konten informasi, termasuk di dalamnya adalah transmisi video lokal, konten premium dari penyedia pihak ketiga/vendor, dan rekaman studio lokal. Sebagian konten dari sumber tersebut masih

dalam bentuk analog sehingga tidak bisa langsung ditransmisikan melalui jaringan protokol Internet. Konten analog tersebut bisa berasal dari format DVD, kaset video. Untuk konten analog seperti ini diperlukan proses encoding untuk mengubah format analog menjadi digital untuk dapat dijadikan ke dalam paket data agar dapat ditransmisikan melalui jaringan TCP/IP. Elemen tambahan yang ikut terlibat di dalam proses selanjutnya adalah aplikasi *Digital Rights Management* (DRM) dan *content management systems* (CMS) dimana perlu untuk menjamin keamanan konten yang disediakan sehingga kemungkinan pencurian konten dapat diminimalisir dengan metode manajemen konten dan aplikasi DRM.

### 2.2.2. Konten Input

Penyedia layanan dari arsitektur IPTV yang diterima oleh komponen *Head End* adalah sebagai berikut:

#### 1. *Premium and Direct-Feed Content*

Konten premium yang termasuk dalam kategori ini adalah konten berbayar dalam format analog dan diperlukan proses encoding oleh MPEG encoder sebelum dikirimkan melalui fungsi IP encapsulator. Media fisik yang digunakan sebagai penyedia konten juga termasuk ke dalam klasifikasi kategori direct feed.

#### 2. *Pre-encoded Content*

Konten yang disediakan oleh penyedia layanan adalah konten yang sudah siap didistribusikan kepada *subscribers* dengan format yang sudah didukung dengan codec yang sesuai, dan cocok dengan program yang ada di sisi *client*. Konten ini dikirim langsung menuju fungsi IP encapsulator.

#### 3. *Local off-the-air content*

Konten ini adalah hasil siaran oleh stasiun lokal dan diterima oleh *server* penerima sinyal PAL/NTSC sebelum konten tersebut dapat digunakan. Standar keluaran dari metode codec yang digunakan adalah sebagai berikut:

- MPEG-2, dengan kode H.262 yang digunakan untuk siaran video digital, sistem distribusi tv kabel, dan *encoding* untuk DVD.
- MPEG-4, part 10, merupakan penerus dari H.262 dengan tingkat kompresi yang lebih baik dibanding dengan pendahulunya tanpa mengurangi kualitas video yang dihasilkan.

### 2.2.3. Video Transcoder

*Video transcoder* menerima data dari konten lokal dengan berbagai macam format yang ada untuk dikonversi menjadi format standar yang digunakan dalam jaringan IPTV yang dibangun. Format video standar yang digunakan adalah MPEG-2, MPEG-4, atau format dengan codec tertentu yang ditentukan dan disepakati dalam menyediakan layanan IPTV. Perangkat yang digunakan untuk proses ini menggunakan spesifikasi sebagai berikut:

#### 1. MPEG Video Encoder

Bertugas dalam menyediakan sarana *recoding*, *recording manager* dan *capture/distribution server* berfungsi untuk mendapatkan data dan membuat konten dengan format yang tepat. Untuk mendapatkan hasil yang tepat, maka konten yang tersedia dari konten input diteruskan menuju MPEG Video encoder yang menghasilkan konten dalam bentuk video digital yang siap untuk dienkripsi, enkapsulasi, atau disimpan dan disebarluaskan menggunakan komponen jaringan IPTV lainnya. Penggunaan video encoder sebagai perangkat dalam jaringan, yang mengubah konten modul pembelajaran yang sudah terbentuk untuk dikonversi oleh fungsi *compressor/decompressor* (codecs). Hasil konversi tersebut menghasilkan format video digital MPEG-2 atau MPEG-4 untuk disesuaikan oleh penyedia layanan untuk dapat diterima dengan baik oleh perangkat yang digunakan di sisi subscriber.

#### 2. IP Encapsulator

Fungsi perangkat ini adalah untuk mengubah video dengan format H.264 (MPEG-4) yang sudah siap disiarkan menjadi paket IP untuk ditransmisikan ke dalam jaringan IP. Sehingga setiap paket IP tersebut siap didistribusikan ke dalam jaringan melalui koneksi Ethernet. Modul IP Encapsulator menerima input yang berasal *pre-encoded* video dan MPEG Video encoder yang mengkonversi video dari *direct feed* dan *video transcoder*. Hasil keluaran dari IP encapsulator dikirimkan secara langsung ke *video streaming server* untuk dilakukan proses pengiriman menuju *subscribers* jika ada permintaan untuk modul video. Hasil konversi dari IP encapsulator juga ditransmisikan menuju DRM dan *database* konten. Konten yang dikirimkan ke DRM akan dienkripsi untuk mengaplikasikan standar keamanan yang akan digunakan untuk mengurangi resiko pencurian konten yang mungkin terjadi. Proses selanjutnya modul video tersebut

dikirimkan ke *video repository server* untuk disimpan dan digunakan apabila ada permintaan dari *subscriber*.

#### 2.2.4. Video Repository

*Video repository* berfungsi untuk menyimpan konten yang akan digunakan berbagai aplikasi untuk proses *broadcasting*. Fasilitas penyimpanan yang tersedia termasuk *video library* dan *media library* yang dilayani oleh *library servers* untuk memastikan kecepatan akses dan akses handal terhadap konten yang diminta. Media yang tersimpan di dalam *Video repository* dapat berupa media siap pakai lengkap dengan proteksi DRM maupun media hasil konversi yang perlu proses enkripsi dan proteksi dengan DRM. Oleh karena itu ditetapkan prosedur standar untuk menjalankan proteksi DRM setiap kali ada video/media yang sudah selesai proses konversi.

#### 2.2.5. Video Streaming Server

*Video streaming server* menjalankan perintah yang diberikan dari *middleware* dan *video-on-demand server*, dan menerima input dari DRM dan *content management server* dengan format media MPEG-4, H.264 atau *codec* tersendiri yang ditentukan oleh penyedia jasa IPTV. Dalam prakteknya, *video streaming server* memiliki kapasitas untuk melayani video streaming ke sejumlah *subscribers* pada saat yang bersamaan. Maka dari itu perangkat ini harus ditambahkan apabila semakin banyak *subscriber* yang terdaftar agar kualitas dari layanan video *streaming* tidak terganggu. Seberapa banyak *subscribers* yang mampu dilayani oleh perangkat ini sangat ditentukan oleh *processing power*. Hasil keluaran dari perangkat video *streaming server* adalah *encoded media* dalam bentuk paket IP yang dikirimkan melalui jaringan TCP/IP menuju modul *set-top box*. Metode otentikasi dasar dapat diletakkan pada perangkat ini dimana akan berinteraksi dengan aplikasi DRM servers apabila konten yang dimaksud belum dienkripsi atau belum siap untuk diedarkan.

### 2.3. Worldwide Interoperability for Microwave Access (WiMAX)

WiMAX adalah singkatan dari *Worldwide Interoperability for Microwave Acces*. Definisi tersebut dibuat oleh forum WiMAX pada bulan Juni 2001 untuk menunjukkan kesesuaian dan interoperabilitas antara IEEE 802.16 yang dikenal secara resmi sebagai *WirelessMAN*. WiMAX

memiliki tujuan sebagai penyedia koneksi data nirkabel sama halnya seperti WiFi, namun dengan skala yang lebih besar, sangat besar bahkan bisa digunakan di satu negara. Forum WiMAX mendefinisikan WiMAX sebagai teknologi berbasis standar yang memungkinkan koneksi akses broadband nirkabel dalam jangkauan yang luas sebagai alternatif untuk koneksi kabel dan DSL (*Digital Subscriber Line*). Bandwith dan jangkauan WiMAX cocok untuk:

- Menghubungkan hotsop-hotspot WiFi
- Menyediakan alternatif koneksi bagi kabel atau DSL untuk daerah-daerah yang sulit dijangkau kabel
- Menyediakan koneksi data kecepatan tinggi dan layanan telekomunikasi
- Menyediakan konektivitas internet

### 2.3.1. Standarisasi WiMAX

Teknologi WiMAX adalah teknologi yang sedang hangat dibicarakan dan diklaim sebagai teknologi 4G masa depan. Standar WiMAX menggunakan standar 802.16 yang ditetapkan oleh IEEE. Standar 802.16 yang ditetapkan oleh IEEE termasuk dalam kategori WMAN (*Wireless Metropolitan Area Network*) memiliki data rate mencapai 70 Mbps dan dapat terjangkau sampai 5 km. IEEE telah menetapkan beberapa standar yang berhubungan dengan komunikasi mobile dan dibagi menjadi empat kategori, yaitu WPAN (*Wireless Personal Area Network*), WLAN (*Wireless Local Area Network*), WMAN (*Wireless Metropolitan Area Network*), dan WWAN (*Wireless Wide Area Network*). Standar 802.16 yang digunakan untuk standar WiMAX mengalami perkembangan dan penyempurnaan, seperti diagram historis berikut ini:

Tabel 2.1 Standarisasi WiMAX

Standar WiMAX	Keterangan
802.16 (Desember 2001)	<i>Original Fixed Wireless Broadband</i> . Bekerja pada frekuensi radio 10-66 GHz. Digunakan pada aplikasi point to point. Standar ini bekerja pada daerah yang LOS ( <i>Line of Sight</i> ). Bandwith mencapai 32 – 124 Mbps. Standar ini cocok untuk jaringan backhaul

802.16a (Januari 2003)	<i>Original Fixed Wireless Broadband</i> . Bekerja pada frekuensi 2 – 11 GHz. Digunakan pada aplikasi point to multipoint
802.16REVd/802.16-2004 (Oktober 2004)	Penambahan <i>WiMAX System Profile</i> . Bekerja pada frekuensi radio 2- 11 GHz. Bandwith mencapai 70 Mbps didesain untuk model penggunaan tetap ( <i>fixed</i> ). Standar ini berfungsi sebagai <i>fixed wireless broadband acces</i> dngan pemasangan antena pada lokasi pelanggan yang dipasangkann pada atap rumah atau dapat menggunakan tiang, sama seperti penampang antena televisi satelit. Teknologi ini menyediakan jaringan tanpa kabel jarak jauh sebagai alternatif pengganti dari modem kabel, <i>Digital Subscriber Lines</i> dengan beberapa tipe ( <i>xDSL</i> ), <i>transmit/exchange</i> (TX/EX), dan jaringan <i>optical carrier level</i> (OC-x). Standar ini digunakan untuk standar <i>WiMAX fixed</i> . Pemakaian <i>WiMAX fixed</i> dapat dikombinasikan dengan jaringan WiFi, sementara <i>WiMAX</i> digunakan sebagai penghubung ke <i>fixed outdoor</i> atau <i>fixed indoor</i> antena.
802.16e/802.16-2005 (Desember 2005)	Standar IEEE 802.16e merupakan amandemen 802.16a untuk perangkat mobile dengan menambahkan portabilitas dan kemampuan roaming pada perangkat CPE-nya. Sistem ini telah memiliki prosedur untuk <i>handover</i> dan roaming. Standar inilah yang digunakan <i>WiMAX mobile</i> .

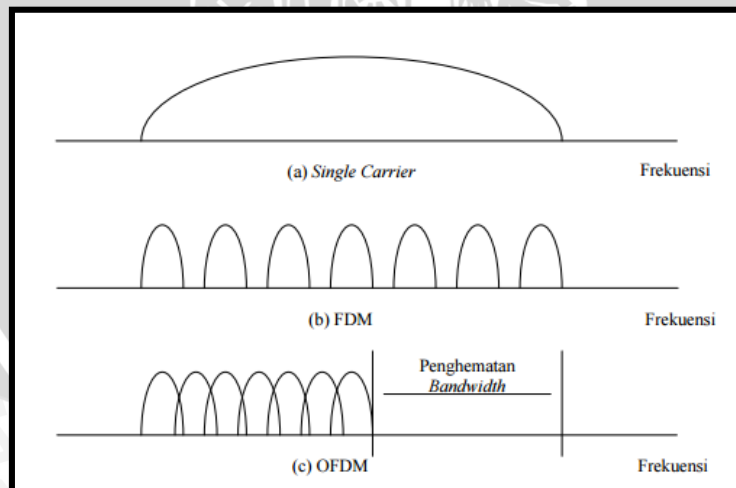
(Sumber: Uke Kurniawan, 2010)

### 2.3.2. Teknologi WiMAX

Implementasi teknologi wireless memerlukan tersedianya jalur *line of sight* (LOS) antara pengirim dan penerima. Apabila terdapat kondisi NLOS, maka dapat menimbulkan redaman propagasi yang dapat menurunkan kualitas sinyal. Teknologi WiMAX didesain bukan hanya untuk kondisi LOS, tetapi juga NLOS. Teknologi WiMAX mampu mengatasi atau mengurangi masalah pada kondisi NLOS.

#### 2.3.2.1. Teknologi OFDM

*Orthogonal Frequency Division Multiplexing* (OFDM) adalah metode modulasi *multicarrier* dengan ide awal untuk mengatasi efek dari *multipath fading* dalam lingkungan *wireless*. *Multipath effect* menyebabkan suatu simbol dapat diterima dalam *multiple copy* dengan waktu yang berbeda, yang menyebabkan terjadinya *intersymbol interference* (ISI) antar simbol penerima. Sistem wireless dengan data rate yang tinggi menggunakan rate simbol yang tinggi pula, dimana simbol yang dikirim pendek-pendek dalam domain waktu. Tetapi, simbol yang pendek tersebut akan mengalami ISI akibat *multipath effect*. Untuk meminimalisasi ISI, maka simbol yang dikirim lebih panjang. Simbol yang terkena ISI pada simbol yang lebih pendek, akan mengalami efek kerusakan yang lebih besar dibandingkan dengan simbol panjang. OFDM menggunakan simbol yang lebih panjang untuk meminimalisasi ISI yang berakibat pada simbol rate dan kapasitas kanal yang rendah.



Gambar 2.2 Teknologi OFDM

(Source: repository.usu.ac.id, 2016)



Prinsip kerja OFDM adalah deretan data informasi yang akan dikirim dikonversikan ke dalam bentuk paralel, sehingga bila bit rate semula adalah  $R$ , maka bit rate di tiap-tiap jalur paralel adalah  $R/M$  dimana  $M$  adalah jumlah jalur paralel (sama dengan jumlah sub-carrier). Setelah itu, modulasi dilakukan pada tiap-tiap sub-carrier. Kemudian sinyal yang telah termodulasi tersebut diaplikasikan ke dalam Inverse Discrete Fourier Transform (IDFT), untuk pembuatan simbol OFDM. Penggunaan IDFT ini memungkinkan pengalokasian frekuensi yang saling tegak lurus. Setelah itu simbol-simbol OFDM dikonversikan lagi ke dalam bentuk serial, dan kemudian sinyal dikirim.

Pada OFDM, frekuensi-frekuensi multicarrier tersebut saling tegak lurus, yang berarti bahwa crosstalk di antara sub-channels dihilangkan dan inter-carrier guard bands tidak diperlukan. Pada OFDM, sinyal didesain sedemikian rupa agar orthogonal, sehingga jika tidak ada distorsi pada jalur komunikasi yang menyebabkan ISI (intersymbol interference) dan ICI (intercarrier interference), maka setiap subchannel akan bisa dipisahkan stasiun penerima dengan menggunakan DFT. Solusi yang termudah adalah dengan menambah jumlah subchannel sehingga periode simbol menjadi lebih panjang, dan distorsi bisa diabaikan jika dibandingkan dengan periode simbol. Keunggulan OFDM adalah tingginya tingkat efisiensi dalam pemakaian frekuensi.

#### **2.3.2.2. Kanalisasi**

Tanpa kanalisasi pengaturan WiMAX menjadi terbatas dan memerlukan biaya tambahan pada CPE (*Customer Premise Equipment*), karena link budget menjadi tidak simetris, yang disebabkan oleh adanya keterbatasan pada komponen CPE. Kanalisasi pada uplink bukanlah merupakan fitur yang utama, tetapi bersifat optional. Kanalisasi dikenal dengan sebutan OFDMA. Teknik ini melakukan pemilihan atau pengelompokan *carrier* dari sejumlah *carrier* OFDM yang diperuntukan pada penerima tertentu.

#### **2.3.2.3. Modulasi Adaptif**

Modulasi adaptif memungkinkan WiMAX dapat mengatur pola sinyal modulasi bergantung pada kondisi signal to noise ratio (SNR) radio link, dimana pada kondisi radio link dengan kualitas yang baik menggunakan pola modulasi yang baik pula, sehingga memberikan sistem dengan kapasitas yang lebih besar. Akibat adanya sinyal fade, modulasi pada WiMAX

dapat beralih ke pola modulasi dengan kualitas yang lebih rendah untuk menjaga kestabilan kualitas hubungan. Fitur modulasi adaptif ini menyediakan sistem untuk melawan time-selective fading, dimana kunci dari modulasi adaptif adalah meningkatkan rentang pola modulasi untuk dapat digunakan pola modulasi dengan kualitas yang terbaik ini dikarenakan sistem dapat mengalihkan kondisi fading. Sebagai sarana untuk mendapatkan pola modulasi yang tetap, maka dalam perhitungannya digunakan parameter pada kondisi yang paling buruk.

### 2.3.3. Alokasi Spektrum Frekuensi WiMAX

WiMAX forum menetapkan 2 band frekuensi utama pada *certification profile* untuk *fixed* WiMAX (band 3.5 GHz dan 5.8 GHz), sementara untuk mobile WiMAX ditetapkan 4 band frekuensi pada sistem profile release-1, yaitu band 2.3 GHz, 2.5 GHz, 3.3 GHz, dan 3.5 GHz.

Secara umum, terdapat beberapa alternatif frekuensi untuk teknologi WiMAX sesuai dengan peta frekuensi dunia. Dari alternatif tersebut band 3.5 GHz menjadi frekuensi mayoritas *Fixed* WiMAX di beberapa negara, terutama di EROPA, Kanada, timur-tengah, Australia, dan Asia, sementara frekuensi mayoritas yang digunakan mobile WiMAX adalah 2.5 GHz.

### 2.4. Kelas Layanan (*Service Class*)

Pada mekanisme MAC layer memungkinkan WiMAX dapat menjalankan QoS dengan berbagai kebutuhan bandwidth dan aplikasi. Sebagai contoh aplikasi suara dan video memerlukan latency yang rendah tetapi masih bisa mentolerir beberapa error rate. Sebaliknya aplikasi-aplikasi data pada umumnya sangat sensitif terhadap error rate, sedangkan faktor latency bukan menjadi pertimbangan kritis. Perubahan parameter QoS bisa diminta oleh SS ke BS dengan sambungan masih tetap terjaga. Pada layer MAC menggunakan scheduling service class untuk mengirimkan dan menangani SDU dan MAC PDU berdasarkan QoS yang diinginkan. Berikut jenis dari scheduling service pada WiMAX.

#### 1. *Unsolicited Grant Service* (UGS)

UGS digunakan untuk layanan yang membutuhkan jaminan pengiriman data dengan prioritas paling utama. Karakteristik UGS meliputi:

- a. Seperti halnya layanan *Constant Bit Rate* (CBR) pada ATM, yang dapat memberikan pengiriman data secara periodik dengan ukuran yang sama.
- b. Untuk layanan yang membutuhkan jaminan *real-time*.

- c. Efektif untuk layanan yang sensitif terhadap *throughput*, *latency* dan *jitter*.
- d. Maximum dan minimum *bandwidth* yang ditawarkan sama.

## 2. *Real-time Polling Service* (rtPS)

Karakteristik dari rtPS meliputi :

- a. efektif untuk layanan yang sensitif terhadap *throughput* dan *latency*.
- b. Garansi rate dan syarat delay telah ditentukan.

## 3. *Non-real-time Polling Service* (nrtPS)

Karakteristik dari nrtPS meliputi :

- a. efektif untuk aplikasi yang membutuhkan *throughput* yang intensif.
- b. Layanan mungkin dapat di perluas sampai *full bandwidth* namun dibatasi pada kecepatan maksimum yang telah ditentukan.
- c. Garansi *rate* diperlukan namun delay tidak digaransi.

## 4. *Best Effort* (BE)

Karakteristik dari BE meliputi :

- a. Untuk trafik yang tidak membutuhkan jaminan kecepatan data.
- b. Tidak ada jaminan pada rate atau delaynya.

## 2.5. Parameter *Quality Of Services* (QoS)

Parameter QoS menggolongkan kualitas transfer yang diberikan oleh suatu koneksi yang diperoleh dengan membandingkan unit data pada sisi masukan dan keluaran *interface*. Tiga diantara Parameter QoS adalah:

### 2.5.1. Bandwith Aplikasi IPTV

Jumlah *bandwidth* yang diperlukan dalam aplikasi IPTV sangat bergantung pada ukuran *frame*, kecepatan *frame* dan jenis pengkodean yang digunakan. Untuk menghitung besar *payload video* dan *payload audio* pada IPTV ditentukan dengan persamaan 2-1 dan 2-2 (Wes Simpson, 2008):

$$P_v = B_{CODEC-v} \times \text{frame size video} \quad (2-1)$$

$$P_A = B_{CODEC-A} \times \text{frame size audio} \quad (2-2)$$

dengan:

$$P_v = \text{Besarnya paket data video pada file (bit)}$$

$P_A$  = Besarnya paket data *audio* pada *file* (*bit*)

$B_{CODEC-V}$  = Besarnya kecepatan *codec video* (*bps*)

$B_{CODEC-A}$  = Besarnya kecepatan *codec audio* (*bps*)

*Frame Size Video* = Lama waktu *video* 1 *frame video* (*s*)

*Frame Size Audio* = Lama waktu *audio* 1 *frame video* (*s*)

Jumlah *payload* yang disegmentasi berdasarkan *payload* maksimum dari ketentuan *codec*. Sehingga, jumlah paket data *video* dan *audio* sesuai dengan persamaan 2-3 dan 2-4 (Wes Simpson, 2008).

$$N_v = \frac{P_v}{P_{v\text{maksimum}}} \quad (2-3)$$

$$N_a = \frac{P_a}{P_{a\text{maksimum}}} \quad (2-4)$$

dengan:

$N_v$  = Jumlah paket *video*

$N_a$  = Jumlah paket *audio*

$P_v \text{ maksimum}$  = *Payload video* maksimum berdasarkan *codec* (*bit*)

$P_a \text{ maksimum}$  = *Payload audio* maksimum berdasarkan *codec* (*bit*)

*Payload* IPTV (*video* dan *audio*) masing-masing ditambahkan *header* NALU, RTP, UDP dan IPv4. Besarnya paket *video* dan *audio* setelah diencodekan dan ditambah *header* sesuai dengan persamaan 2-5 dan 2-6 (Wes Simpson, 2008).

$$P_v (\text{encoded}) = P_v + N_v \times (H_{NALU} + H_{RTP} + H_{UDP} + H_{IPv4}) \quad (2-5)$$

Dan untuk *audio*,

$$P_a (\text{encoded}) = P_a + N_a \times (H_{NALU} + H_{RTP} + H_{UDP} + H_{IPv4}) \quad (2-6)$$

dengan:

$P_v (\text{encoded})$  = *Payload video* dengan *header* NALU, RTP, UDP, IP (*byte*)

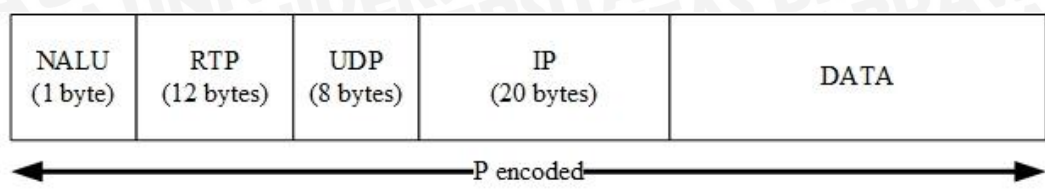
$P_a (\text{encoded})$  = *Payload audio* dengan *header* NALU, RTP, UDP, IP (*byte*)

$H_{NALU}$  = *Header* NALU (*byte*)

$H_{RTP}$  = *Header* RTP (*byte*)

$H_{UDP}$  = *Header* UDP (*byte*)

$$H_{IPv4} = \text{Header IPv4 (byte)}$$



Gambar 2.3 Format *Datagram* UDP RTP *Payload*  
(Sumber: Nuaymi, 2007)

Besarnya paket data aplikasi IPTV yang ditransmisikan pada sistem ditentukan dengan persamaan 2-7 (Wes Simpson, 2008).

$$P_{IPTV\ size} = Pv(encoded) + Pa(encoded) \quad (2-7)$$

dengan:

$$P_{iptv\ size} = \text{Panjang datagram IPTV (byte)}$$

$Pv(encoded)$  = *Payload video* setelah penambahan *header* NALU, RTP, UDP, IP (byte)

$Pa(encoded)$  = *Payload audio* setelah penambahan *header* NALU, RTP, UDP, IP (byte)

### 2.5.2. Delay

*Delay* atau keterlambatan adalah sebuah kondisi dimana terjadi selisih waktu antara waktu paket diterima dan waktu pengirimannya. *Delay* bisa disebabkan oleh berbagai hal baik karena proses pemaketan sebuah data, proses propagasi dan *delay* yang disebabkan banyaknya komponen yang mengakses. *Delay* dihitung dengan melihat total rata rata waktu terima dikurangi pengiriman dalam satu kali pengamatan dibagi jumlah usaha pengiriman. Kategori kinerja jaringan berdasarkan nilai *delay* dapat dilihat pada Tabel 2.2. berikut:

**Tabel 2.2.** Kategori kualitas *delay*

Kategori	Delay
Bagus	0 - 150 ms
Cukup, Dapat ditolerir	150 ms – 400 ms
Buruk	>400 ms

(Sumber: ITU-T G.114, 2000)

Berdasarkan tabel diatas, persamaan umum menghitung delay adalah sebagai berikut:

$$\text{Delay} = \frac{\text{Between First and Last Packet (sec)}}{\text{pakets}} \quad (2.8)$$

dengan:

*Between First and Last Packet* : Durasi IPTV (ms)

*Packet* : Total paket data (paket)

### 2.5.2.1. Delay End to End

*Delay end to end* pada WiMAX merupakan waktu yang dibutuhkan untuk mengirimkan paket data dari BS menuju SS. Total *delay end to end* pada WiMAX adalah: (Mischa Schwartz, 1994)

$$t_{\text{end to end}} = t_{\text{codec}} + t_{\text{MAN}} \quad (2.9)$$

dengan :

$$t_{\text{end to end}} = \text{delay end to end (ms)}$$

$$t_{\text{codec}} = \text{delay codec (ms)}$$

$$t_{\text{MAN}} = \text{delay MAN (ms)}$$

### 2.5.2.2. Delay Codec

*Delay codec* merupakan *delay* yang terjadi pada sisi encoder. *Delay codec* adalah waktu yang dibutuhkan untuk mengakumulasi *sample* suara ke dalam *frame* suara, waktu untuk mengkompresi paket suara, waktu untuk memuat *frame* suara ke dalam paket dan mentransfer paket tersebut ke jaringan *transport* dan *delay hardware* yang bersifat tetap. *Delay codec* pada aplikasi IPTV dapat dihitung dengan menggunakan persamaan berikut:

$$t_{\text{codec}} = t_a + t_v \quad (2.10)$$

dengan:

$t_a$  : *delay codec* audio (ms)

$t_v$  : *delay codec* video (ms)

### 2.5.2.3. Delay MAN

*Delay* MAN merupakan *delay* total keseluruhan yang diperlukan untuk mengirimkan data dari sisi pelanggan ke pelanggan lain pada jaringan *fixed* WIMAX. *Delay* MAN dapat dihitung dengan menggunakan persamaan berikut:

$$t_{MAN} = t_{proses} + t_{prop} + t_{trans} + t_w \quad (2.11)$$

dengan :

$t_{MAN}$  = *delay* total pada MAN (ms)

$t_{proses}$  = *delay* proses (ms)

$t_{prop}$  = *delay* propagasi (ms)

$t_{trans}$  = *delay* transmisi (ms)

$t_w$  = *delay* antrian (ms)

- *Delay* proses

*Delay* proses merupakan waktu yang dibutuhkan untuk memproses paket data dan menentukan kemana data tersebut akan diteruskan. *Delay* proses pada jaringan *fixed* WIMAX terdiri dari *delay* enkapsulasi dan *delay* dekapsulasi. *Delay* enkapsulasi adalah waktu yang dibutuhkan untuk menambahkan *header* pada paket data, sehingga paket data tersebut dapat tepat sampai ke *host* tujuan. Sedangkan *delay* dekapsulasi adalah waktu yang dibutuhkan untuk melepaskan keseluruhan *header* dari sebuah paket data.

*Delay* enkapsulasi dapat dihitung menggunakan persamaan :

$$t_{enc} = \frac{W_{frame\ total}}{c} \times 8 \quad (2.12)$$

dengan:

$t_{enc}$  = *delay* enkapsulasi (ms)

$W_{frame\ total}$  = panjang *frame* (byte)

$c$  = kecepatan transmisi kanal (bps)

Sedangkan *delay* dekapsulasi dirumuskan sebagai berikut:

$$t_{dec} = \frac{W_{frame\ total}}{c} \times 8 \quad (2.13)$$

sehingga *delay* proses dapat dirumuskan sebagai berikut:

$$t_{proc} = t_{enc} + t_{dec} \quad (2.14)$$

- *Delay* propagasi

*Delay* propagasi adalah waktu yang dibutuhkan untuk merambatkan paket data melalui media transmisi. *Delay* propagasi dapat ditulis dengan persamaan:

$$t_{prop} = \frac{d_{max}}{v} \quad (2.15)$$

dengan:

$d_{max}$  = jarak jangkauan *base station* dan *subscriber station* (m)

$v$  = kecepatan sinyal pada media *wireless* ( $3 \times 10^8$  m/s)

- *Delay* transmisi

*Delay* transmisi adalah waktu yang dibutuhkan untuk meletakkan semua data pada media transmisi dan dipengaruhi oleh ukuran paket serta kapasitas media transmisi. *Delay* transmisi dapat ditentukan dengan persamaan (Mischa Schwartz, 1987):

$$t_{trans} = \frac{w}{c} \times 8 \quad (2.16)$$

dengan:

$w$  = panjang total *frame* yang dikirimkan (byte)

$c$  = kecepatan (bps)

- *Delay* antrian

*Delay* antrian adalah waktu yang dibutuhkan data selama berada dalam antrian untuk ditransmisikan. Selama waktu ini, paket data menunggu hingga paket yang lain selesai ditransmisikan. *Delay* antrian dapat dihitung dengan persamaan berikut :

$$\mu = \frac{c}{w} \quad (2.17)$$

$$\lambda_w = \mu \rho \quad (2.18)$$

$$t_w = \frac{\lambda_w}{\mu(\mu - \lambda_w)} + \frac{1}{\mu} \quad (2.19)$$

dengan :



- $t_w$  = delay antrian (ms)  
 $\lambda_w$  = kecepatan kedatangan (paket/s)  
 $\mu$  = waktu pelayanan (paket/s)  
 $c$  = kapasitas kanal (bps)  
 $\rho$  = faktor utilitasi sistem nirkabe

### 2.5.3. Throughput

*Throughput* didefinisikan sebagai ukuran yang menyatakan berapa banyak bit yang dapat ditransmisikan dan sukses diterima di tujuan per detik untuk lebar pita yang dialokasikan.

*Throughput* ditunjukkan oleh persamaan (Scwartz, Mischa. 1987) :

$$\lambda = \frac{(1-\rho)}{t_1[1+(\alpha-1)\rho]} \quad (2.20)$$

Simbol  $\alpha$  merupakan konstanta propagasi dengan persamaan sebagai berikut :

$$\alpha = \left(3 + \frac{2t_{prop}}{t_t}\right) \quad (2.21)$$

dengan :

$\alpha$  : konstanta propagasi

$t_{prop}$ : delay propagasi(s)

$t_t$  : delay transmisi (s)

### 2.5.4. Packet loss

Probabilitas *Packet loss* menunjukkan banyaknya jumlah *packet* yang hilang setelah terjadi proses pentransmision sesudah melewati sistem jaringan. Sedangkan untuk probabilitas *packet loss* sendiri adalah probabilitas yang menunjukkan kemungkinan adanya data yang hilang ketika proses pentransmision selama berada dalam sistem.

Maka bisa diartikan *packet loss* menunjukkan juga banyak *packet* yang hilang dalam satu kali pengamatan atau simulasi. Rumusan penghitungan dengan melihat *packet loss* yang ada. *Packet loss* dilihat berdasarkan persentase nilai paket yang hilang/loss dalam satu jaringan. Secara umum terdapat empat kategori penurunan performansi jaringan berdasarkan nilai *packet loss* sesuai dengan versi *TIPHON-Telecommunications and Internet Protocol Harmonization Over Networks*.

Persamaan umum untuk mendapatkan nilai probabilitas *packet loss* dapat dihitung dengan persamaan sebagai berikut:

- Probabilitas *Packet loss* IPTV

Probabilitas *packet loss* pada IPTV dapat dihitung sesuai persamaan :

$$\rho_{\text{live IPTV}} = P_{\text{va IPTV}} \times \rho_b$$

- Probabilitas *packet loss* jaringan

Perhitungan *packet loss* jaringan WIMAX untuk tipe modulasi 64 QAM sesuai dengan persamaan berikut:

$$\rho_{\text{net}} = \frac{e^{\frac{Eb}{No}}}{2\sqrt{3,14}(\frac{Eb}{No})} \quad (2.22)$$

Maka probabilitas *packet loss* total untuk modulasi 64 QAM adalah

$$\rho_{\text{total}} = 1 - [(1 - \rho_{\text{net}})(1 - \rho_{\text{vid}})] \quad (2.23)$$

Tabel 2.3. Kategori *Packet loss*

Kategori	<i>Packet loss</i>
Sangat Bagus	0 %
Bagus	0 % - 3 %
Sedang	3 % - 15 %
Buruk	15 % - 25 %

(Sumber: ITU-T G.1010, 2002)

## 2.6. Aplikasi Packet Analyzer Wireshark

WireShark adalah sebuah Network Packet Analyzer. Network Packet Analyzer akan mencoba menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi di paket tersebut secara detail.



Gambar 2.4 Logo WireShark

(Sumber: [www.wireshark.org](http://www.wireshark.org))

Perumpamaan sebuah Network Packet Analyzer sebagai alat untuk memeriksa apa yang sebenarnya sedang terjadi di dalam kabel jaringan, seperti halnya voltmeter atau tespen yang digunakan untuk memeriksa apa yang sebenarnya sedang terjadi di dalam sebuah kabel listrik. Dulunya *tool* semacam ini sangatlah mahal harganya, dan biasanya dengan alasan hak cipta. Namun, dengan adanya WireShark, kita akan sangat dimudahkan. Mekanik tidak sedikit yang bilang bahwa WireShark adalah salah satu tool gratis (open source) terbaik untuk menganalisa paket jaringan. Berikut beberapa contoh penggunaan WireShark:

1. Troubleshooting masalah-masalah di jaringan
2. Analisa performansi jaringan
3. Pemeriksaan keamanan jaringan
4. Pengembangan software untuk men-*debug* implementasi protokol jaringan dalam software
5. Analisa protokol jaringan secara detail

### 2.6.1. Fitur WireShark

Untuk mendapatkan versi terbaru dari WireShark, kita bisa memeriksa di [www.wireshark.org/download.html](http://www.wireshark.org/download.html). Disini nantinya akan ada banyak mirror yang menyediakan link download untuk WireShark. Untuk instalasi WireShark tidak memerlukan tambahan aplikasi, namun akan diminta penginstalan WinPcap, apabila tidak mempunyai WinPcap, nanti kita tidak akan bisa meng-capture penggunaan WireShark. Fitur-fitur WireShark diantaranya:

1. Tersedia untuk Linux dan Windows
2. Menangkap / capture paket data secara langsung dari sebuah network interface
3. Mampu menampilkan informasi yang sangat detail mengenai hasil capture tersebut
4. Import dan Export hasil Capture dari atau ke komputer lain
5. Pencarian paket dengan berbagai macam kriteria filter
6. Dapat menampilkan berbagai macam tampilan statistika