

PENGANTAR SISTEM INFORMASI GEOGRAFIS

PENGANTAR

SISTEM INFORMASI

GEOGRAFIS

Sejarah, Definisi dan Konsep Dasar

Rolly Maulana Awangga



Kreatif Industri Nusantara

Pengantar Sistem Informasi Geografis

Penulis:

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

Editor:

M. Yusril Helmi Setyawan

Penyunting:

Syafrial Fachrie Pane

Khaera Tunnisia

Diana Asri Wijayanti

Desain sampul dan Tata letak:

Deza Martha Akbar

Penerbit:

Kreatif Industri Nusantara

Redaksi:

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

Distributor:

Informatics Research Center

Jl. Sarasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, Februari 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara
apapun tanpa ijin tertulis dari penerbit

*For my family Bunda, Khafayah dan
Khanza*

KATA PENGANTAR

Saya ucapkan alhamdulillah atas selesainya buku ini. Berkat semua dukungan rekan dan mahasiswa alhamdulillah buku ini bisa terselesaikan. Terima kasih untuk keluarga kecil saya bunda Mila, Khafa dan Khanza yang senantiasa menemani dan mendukung seluruh aktifitas penulis. Tidak lupa pula terima kasih kepada rekan-rekan mahasiswa bimbingan yang membantu layouting dan editing agar buku ini bisa terbit.

Buku ini merupakan buku hasil pemikiran dan uji coba serta praktik lapangan selama mendalami Sistem Informasi Geografis. Diawali dengan kebutuhan akan aplikasi open source pada Kementerian Sekretariat Negara yang menolak menggunakan Google Map untuk dijadikan base map di Aplikasinya. Hal ini dikarenakan untuk mengendalikan kedaulatan data rahasia milik negara yang tidak ingin dikirimkan ke pihak manapun apalagi pihak luar. Yang akhirnya saya mencoba berbagai macam alternatif untuk membangun sendiri server peta dan kelengkapan penggunaannya. Seiring perkembangan waktu dan kerja sama dengan berbagai pihak. Ternyata memang banyak kebutuhan GIS dengan aplikasi open source ini dengan alasan yang sama yaitu kedaulatan data. Selain itu open source yang membawa jargon freedom sesuai dengan filosofi keamanan data tanpa ketergantungan dengan berbagai aplikasi Proprietary. Dengan aplikasi open source juga bisa menekan biaya pengeluaran instansi dalam hal pengeluaran pembelian piranti lunak GIS. Termasuk dalam hal pembelajaran di kampus. Kampus tidak perlu lagi membeli piranti lunak mahal, dan mahasiswa tidak perlu untuk berlangganan kepada Google Maps yang cukup menguras uang ketika akan mengimplementasikannya di tempat kerja. Mungkin kita sudah mengetahui betapa mudahnya menggunakan Google Maps API pada saat kuliah. Akan tetapi ketika diterapkan pada sebuah proyek GIS, nilainya akan tinggi sekali untuk pembayaran penggunaan API tersebut. Belum lagi masalah keamanan data yang terus di pantau oleh Google.

Semoga kedepannya buku ini akan terus dikembangkan dan akan semakin dilengkapi dengan pengalaman-pengalaman baru penulis di edisi selanjutnya. Go Open Source Go Freedom.

Bandung, Februari 2019

Penulis

ROLLY MAULANA AWANGGA

DAFTAR ISI

Kata Pengantar	vii
Daftar Gambar	xvii
Daftar Tabel	xxi

BAGIAN I PENDAHULUAN

1 Definisi	3
1.1 Definisi GIS (GEOGRAPHICS INFORMATION SYSTEM)	3
1.1.1 Pemahaman pada Geographic Information System GIS	3
1.1.2 Definisi GIS (Geography Information and System)	4
1.2 Geographic Information System (GIS): Introduction to the computer perspective	5
1.2.1 Pengenalan GIS atau Geography Information System	5
1.2.2 Komponen GIS atau Geography Information System	5
1.2.3 Kaidah GIS atau Geography Information System	5
1.2.4 Kesimpulan GIS atau Geography Information System	6
1.2.5 Saran GIS atau Geography Information System	7
1.2.6 Tipe Data Geospasial	7
2 Sejarah Ptolemy	9
2.1 Peta	9
2.1.1 Peta Menurut Claudius Ptolemaeus Ptolemy	9
2.1.2 Peta Dunia Ptolemy	10
2.1.3 Sejarah Ptolemy	10
2.1.4 The Geography	12
3 Sejarah eratosthenes	15
3.1 Eratosthenes	15
3.1.1 Bapak Geografi	15
3.1.2 Mempelajari Bumi	15
3.1.3 Mengukur Bumi	17
3.1.4 Prestasi	18
3.1.5 Pekerjaan	18
4 Sejarah aliddrissi	21
4.1 Peta	21

4.1.1	Peta Menurut Abu Abdullah Muhammad al-Idrisi al-Qurtubi al-Hasani as-Sabti ‘al-Idrisi’	21
4.1.2	Sejarah Peta dari pandangan Al-Qur'an 'al-idrisi'	22
5	Sejarah Peta Dinding	25
5.1	Biografi Willem Jansz Blaeu	25
5.1.1	Karya individual Willem Jansz Bleau	25
5.2	Peta dinding diterbitkan	25
5.2.1	Karya Willem Blaeu	27
6	Sejarah Bumi	29
6.1	Sejarah Bumi	29
6.1.1	Teori-teori terbentuknya Bumi	29
6.2	Pendapat Tentang Sejarah Bumi	32
7	Sejarah Benua	35
7.1	Sejarah Benua	35
7.1.1	Benua pertama	35
7.1.2	Benua raksasa pada masa Proterozoikum	35
7.1.3	Bukti Tersusunnya Benua Kuno	36
7.2	Sejarah Koordinat	36
7.3	Sistem Koordinat	37
7.3.1	Sistem Koordinat Dua Dimensi	37
7.4	Geometri Koordinat	39
7.4.1	Sketsa Grafik Garis	39
7.4.2	Persamaan Garis Lurus	39
7.4.3	Persamaan Lingkaran	39
7.4.4	Program Linear	39
7.4.5	Pembelajaran Geometri Koordinat	39
8	Sejarah Penentuan Waktu	41
8.1	Sejarah Waktu	41
8.2	Penentuan Waktu	42
8.2.1	Hari Matahari	42
8.2.2	Hari Bintang	42
8.2.3	Hari Matahari Menengah/Matahari Khayal dan Perata Waktu	42
8.2.4	Greenwich Mean Time(GMT)	43
8.3	Waktu Standar	43
9	Sejarah Penanggalan	45
9.1	Sejarah penanggalan	45
9.1.1	Solar calendar/Kalender Surya	45
9.1.2	lunar calendar/Kalender candra	46
9.1.3	lunisolar calendar/kalender surya candra	48
BAGIAN II DASAR PEMETAAN		
10	Mengenal Bangun Ruang	51
10.1	Bangun Ruang	51
10.1.1	Bola	53
11	Mengenal Diagram Kartesius	59
11.1	Pengertian Diagram Kartesius	59
11.2	Penghitungan Rumus Diagram Kartesius	60
11.2.1	menghitung rumus, mencari titik	60
11.3	Contoh Penerapan/Pemetaan Diagram Kartesius	61

11.3.1 Kuadran B	62
11.3.2 Kuadran D	63
11.4 Pengertian Bidang atau Diagram Kartesius	63
12 Garis Khatulistiwa	67
12.1 Definisi	67
12.1.1 pengertian garis khatulistiwa	67
12.1.2 pengertian prime meridian	67
12.2 Kordinat dan Wilayah	67
12.2.1 sistem koordinat bumi	67
12.2.2 pemanfaatan prime meridian	68
12.3 Penutup	70
12.3.1 Kesimpulan	70
12.3.2 Saran	71
13 Kordinat Indonesia	73
13.1 Koordinat Lintang Utara, Lintang Selatan, Bujur Timur, Bujur Barat	73
13.1.1 Sistem Koordinat	74
14 Kordinat Internasional	79
14.1 latitude longitude	79
14.1.1 Latitude	79
14.1.2 Longitude	79
14.2 LINTANG	79
14.3 GARIS BUJUR	80
14.4 Waktu Lokal (LT) dan Zona Waktu	80
14.4.1 Glosarium	81
14.5 Konversi antara koordinat geografis dan Kartesian koordinat	82
14.6 LINTANG/LATITUDE	82
14.7 BUJUR/LONGTITUDE	82
BAGIAN III DATA GEOSPASIAL	
15 Data Raster	87
15.1 Data Raster	87
15.1.1 Pengertian Data Raster	87
15.1.2 Pengertian Data Vektor	87
15.1.3 Kelebihan dan kekurangan Data Raster	87
15.1.4 perbedaan data raster dan data vektor	88
15.1.5 karakteristik data raster	88
15.1.6 Metode Penyimpanan Data Raster	88
15.1.7 Akses ikonik ke repositori format data raster data monokrom elektronik jarak jauh	89
15.1.8 Pengertian PostGIS	90
16 Data Vektor	91
16.1 Data Vektor	91
16.1.1 Data Raster	91
16.1.2 Model Data Vektor	91
16.1.3 Perbedaan Data Vektor dan Raster	92
16.1.4 Objek pada data vektor	92
16.1.5 Cara Menggunaan Data Vektor	92
16.1.6 Aplikasi Yang Memudahkan Pembuatan Data Vektor	93
16.1.7 Kekurangan dan Kelebihan Data Vektor	93
16.1.8 Model Data Vektor	94
16.1.9 Klasifikasi Model Data Spasial	95

17 Open Geospatial Consortium	97
17.1 Open Geospatial Consortium	97
17.1.1 Definisi	97
17.1.2 Dasar-dasar Ilmiah	98
17.1.3 Pengertian Geospasial	99
17.1.4 Definisi	99
17.1.5 GeospatialWebService	99
17.1.6 GeospatialSemanticWeb	99
17.1.7 OGC Standar	99
18 Web Map Tile Service	101
18.1 Web Map Tile Service	101
18.1.1 Skema Ubin	101
18.1.2 Metode Pemotongan Web Map Tiles Service	101
18.1.3 Peta Jalan	102
18.1.4 Desain Ekstraksi Data WMTS	103
18.1.5 Kerangka Tethys	103
18.1.6 Desain Antarmuka Pengguna Inspektur Salju	103
18.1.7 Salju Inspektur Data Application Programmer Interface Design	104
19 Web Map Service	105
19.1 Deskripsi WMS	105
19.1.1 Pengenalan peta map service	105
19.1.2 Interoperability antara WM dan grid data lingkungan	106
19.2 Mengimplementasikan NcWMS sebagai aplikasi Web Java	106
19.2.1 Menjelaskan tentang layanan WMS	107
19.3 Beberapa operasi wms pada tahap implementasi	107
19.3.1 WMS	107
20 Data Vektor Line	109
20.1 Geospasial	109
20.1.1 Pengertian Geospasial	109
20.2 Data Spasial	109
20.2.1 Definisi Data Spasial	109
20.3 Tipe Data Vektor	109
20.3.1 Definisi Tipe Data Vektor	109
20.4 Data Atribut	110
20.4.1 pengertian Data Atribut	110
20.5 Data Vektor Line	111
20.5.1 Pengertian	111
20.6 Raster	112
20.6.1 Pengertian Raster	112
21 Shapefile	115
21.1 Shapefile	115
21.1.1 Pengertian Shapefile	115
21.1.2 Struktur Data Shapefile	115
21.1.3 Daftar beberapa file extension	116
21.1.4 Komponen Teknis	117
21.1.5 Deskripsi Teknis Shapefile	117
21.1.6 Mengapa Harus menggunakan Shapefile	117
21.1.7 Shapefile dapat dibuat dengan 4 Metode Umum	117
21.1.8 Jenis Numerik pada Shapefile	117
21.1.9 Pengertian Shapefile	117
21.1.10 sistem informasi geografis shapefile	118

21.1.11 spesifikasi shapefile	118
21.1.12 format data shapefile	118
21.1.13 pembuatan shapefile	118
22 Shapefile Point	119
22.1 Definisi GIS(GEOGRAPHICS INFORMATION SYSTEM)	119
22.2 Definisi Data Spasial (GEOGRAPHICS INFORMATION SYSTEM)	119
22.2.1 ABSTRAK	119
22.2.2 Diagram Voronoi untuk Point, Line dan Poligon dalam fitur GIS	120
22.2.3 Model data Vektor pada Geographics Information System GIS	120
22.3 contoh perancangan sistem (GEOGRAPHICS INFORMATION SYSTEM) dengan menggunakan Point	121
22.3.1 Layer objek/point	121
22.3.2 Pola Titik	121
22.3.3 Point dalam MongoDB	122
23 Shapefile Poligon	125
23.1 Data Polygon	125
23.1.1 Data Geospasial	125
23.2 Pengertian Polygon	126
23.2.1 Jenis - jenis Polygon	126
23.2.2 Bentuk-bentuk Polygon	126
23.2.3 Karakteristik Polygon	127
BAGIAN IV PEMROGRAMAN SIG	
24 Python	131
24.1 Pengertian python	131
24.2 Sejarah python	131
24.3 Indentation	131
24.3.1 Fitur dan filosofi	132
24.4 instalasi python untuk windows	132
24.4.1 Pemrograman Dalam Bahasa Python	133
24.4.2 methods	134
24.4.3 typing	135
24.4.4 mathematics	135
24.4.5 Pernyataan dan Arus Kontrol	136
25 Variabel	139
25.1 Variabel	139
25.1.1 Membuat Perubah Variabel	142
25.1.2 Aturan Penulisan Variabel	144
25.1.3 Menghapus Variabel	144
25.1.4 Bilangan integer dan float	145
25.1.5 List	145
26 Looping	147
26.1 Perulangan Pada Python	147
26.1.1 While dan For	147
26.1.2 Perintah Break, Continue dan Pass Perintah Break	147
26.1.3 Nested Loop	149
26.1.4 While Loop	150
26.1.5 Perulangan do-while	150
26.1.6 Perulangan (for loop)	150
26.1.7 For Loop	150
26.1.8 While Loop	151

26.1.9 For looping with list	151
26.1.10 Infinite Loop	151
26.1.11 break and continue statements	152
26.1.12 Perintah break, continue dan else	152
26.1.13 Range	152
26.1.14 for loop with else	153
26.1.15 Penggunaan loop dengan else statement	154
26.1.16 Middle-test loop	154
26.1.17 Penjelasan Penggunaan For Loop	154
26.1.18 Pendukung kontrol dalam penggunaan looping python	154
27 Kelas dan Fungsi di Python	155
27.1 Phyton	155
27.1.1 Pengertian Phyton	155
27.1.2 Contoh Script Phyton	155
27.1.3 INTERPRETER PYTHON	155
27.2 Class	156
27.2.1 Pengertian Class	156
27.2.2 Area Pokok Class	156
27.2.3 Hubungan Antar Class	157
27.3 Fungsi	157
27.3.1 FUNGSI-FUNGSI UMUM	157
27.4 Contoh Class dan Fungsi pada Phyton	158
27.4.1 Contoh Class	158
27.4.2 Contoh Fungsi	158
27.4.3 Contoh Script Shapefile	158
28 OpenLayer	161
28.1 deskripsi openlayer	161
28.1.1 Base Layers	161
28.1.2 Non Base Layers	161
28.1.3 Raster Layers	161
28.1.4 Google	161
28.1.5 Overlay layer	162
28.1.6 Boxes	162
28.1.7 GML	162
28.1.8 GeoRSS	162
28.1.9 Markers	162
28.1.10 Text	162
28.1.11 Vector	162
28.1.12 WFS	162
28.2 Map Controls	163
28.2.1 ArgParser	163
29 LeafletJS	165
29.1 sejarah leafletjs	165
29.2 pengertian leafletjs	165
29.3 penggunaan leafletjs	166
29.4 fungsi leafletjs	166
29.5 permulaan leafletjs	166
29.6 Kelebihan dan Kekurangan Leaflets	166
29.7 TWEAK BASIS KODE Leaflets	166
BAGIAN V APLIKASI SIG	
30 Mapfile	169

30.1	Raster Layer dan Vektor Layer	169
30.1.1	Raster Layer	169
30.1.2	Vektor Layer	170
30.1.3	Menampilkan Map Bandung	171
30.1.4	Class	171
30.1.5	STYLE Objects	171
30.1.6	MapServer	171
30.1.7	About STYLE Objects	172
31	QGIS	173
31.1	QGIS	173
31.1.1	Getting QGIS	173
31.1.2	Installing QGIS	174
31.1.3	Pemberian Label	174
31.1.4	Toolbar	175
31.1.5	Status Bar	176
31.1.6	Menu Bar	176
31.1.7	Atribut	177
31.1.8	Label Tool	178
31.1.9	Fitur Dasar Quantum GIS Dalam Pengelolaan Data Vektor dan Raster	180
31.1.10	Menambahkan Data Vektor	181
31.1.11	Mode Algoritma	182
31.1.12	Klasifikasi Data Vektor	183
31.1.13	MS4W	183
32	Python Shapefile	185
32.1	Pyshp	185
32.1.1	ESRI Shapefile	185
32.1.2	Instalasi pyshp	186
32.1.3	Dasar Penggunaan	186
	Daftar Pustaka	187

DAFTAR GAMBAR

1.1	data spasial berikut berupa titik, garis, poligon (2-D), permukaan (3-D).	3
1.2	komponen GIS.	6
1.3	kaidah GIS	7
1.4	aplikasi GIS	8
2.1	Gambar Peta menurut Ptolemy	9
2.2	Foto Ptolemy	10
2.3	Gambar Ptolemy	11
2.4	Gambar Perincian Timur dan Asia	11
2.5	Konsep artis zaman pertengahan dari Claudius Ptolemaeus.	12
2.6	Geography by Ptolemy	13
3.1	ilustrasi yang tidak bertanggal dari para ilmuwan di Perpustakaan Alexandria Bettmann / CORBIS	16
3.2	Rekonstruksi Eratosthenes c. 194 SM peta dunia, dari E.H. Bunbury's 1883 Sejarah Geografi Kuno di antara orang Yunani dan Romawi dari Abad Pertengahan sampai Kejatuhan Kekaisaran Romawi, ranah publik	17
3.3	Sebuah diagram yang menunjukkan bagaimana Eratosthenes mengukur Bumi, diakses dari Simon Fraser University Online	18
4.1	Gambar Peta al-Idrisi tahun 1154 berdasarkan French National Library	21
4.2	Monumen Muhammad al-Idrisi di tempat kelahirannya Ceuta Spanyol	22
4.3	Tabula Rogeriana Salah Satu peta yang paling canggih pada masa itu	23
5.1	Willem Jansz Blaeu	26
5.2	Peta Dinding	26
6.1	Gambar Teori Nebula	30
6.2	Gambar Teori Planetesimal	30
6.3	Gambar Pasang Surut	31
6.4	Gambar Teori Bintang Kembar	32
6.5	teori big bang	33
7.1	Peta geologi Amerika Utara, kode warna menunjukkan usia. Warna merah dan pink menunjukkan batuan dari masa eon Arkean.	36

7.2	Rekonstruksi benua raksasa Pannotia (warna kuning) pada 550 juta tahun lalu.	37
7.3	Keempat kuadran sistem koordinat Kartesius	38
7.4	Nilai x dan y pada Kuadran I,II,III,IV	38
8.1	Gambar skala waktu yang ditetapkan.	41
8.2	Perata Waktu	43
8.3	Pembagian Daerah Waktu di Dunia.	44
8.4	Pembagian Waktu di Indonesia Tahun 1964	44
9.1	Kalender tahun 2015 Masehi / 1436 Hijriah.	47
10.1	beberapa kumpulan gambar yang termasuk dalam bangun ruang	52
10.2	contoh bola	53
10.3	contoh sisi lengkung	54
10.4	contoh usur bola	55
10.5	sebuah bola yang terdapat dalam tabung, untuk mengukur luar permukaan tabung	56
10.6	gambar bola	56
10.7	beberapa kumpulan gambar yang termasuk dalam bangun ruang	57
10.8	wadah dalam bola	57
10.9	pasir dalam wadah	58
11.1	hubungan antar titik pada diagram kartesius.	59
11.2	rumus mencari K faktor.	60
11.3	penentuan kuadran pada diagram kartesius.	60
11.4	penentuan titik pada kuadran kartesius.	61
11.5	penentuan garis pada kuadran kartesius.	62
11.6	Diagram Kartesius	63
11.7	Hasil Perhitungan Diagram Kartesius pada Kuadran A	64
11.8	Hasil Perhitungan Diagram Kartesius pada Kuadran B	65
11.9	Hasil Perhitungan Diagram Kartesius pada kuadran C	65
11.10	Hasil Perhitungan Diagram Kartesius pada Kuadran D	66
11.11	penentuan garis/titik dalam diagram kartesius	66
12.1	menjelaskan tentang sudut lintang dan bujur pada bumi.	68
12.2	menjelaskan tentang zona waktu pada tiap belahan dunia.	69
12.3	list negara yang dilalui garis khatulistiwa.	70
12.4	wilayah di dunia yang dilewati garis khatulistiwa.	70
13.1	Koordinat Lintang dan Bujur	73
13.2	Bentuk titik Koordinat	74
13.3	Titik Lintang dan Bujur	75
13.4	Titik koordinat Lintang pada sumbu Y	76
13.5	Titik koordinat Bujur pada sumbu X	77
14.1	Garis Lintang atau Latitude.	80
14.2	Garis Bujur atau Longitude.	81
14.3	gambar Latitude.	82

14.4	gambar longitude.	83
15.1	Gambar Run Length Encoding	89
15.2	Gambar Block Encoding	89
15.3	Gambar Chain Encoding	90
15.4	Gambar Quadtree Data Structure	90
16.1	tampilan dari data vektor dan data raster	92
16.2	tampilan skala data vektor	93
16.3	representasi data vektor dan atributnya	94
17.1	Logo OGC	97
17.2	OGC Geoprocessing Standards	100
18.1	Representasi piramida ubin	102
18.2	Snow Retrieval Function Loop for Point Location	104
19.1	Menjeleskan tentang Interoperability antara WMS dan grid data lingkungan	106
20.1	Perbedaan Vektor dengan Raster.	110
20.2	Sungai merupakan contoh vektor line pada peta.	110
21.1	Gambar Shapefile Poligon	116
22.1	voronoi.	120
22.2	point.	121
22.3	kelaspoint.	122
22.4	layerpoint.	123
23.1	Klasifikasi Model Data Geospasial.	125
24.1	Guido van Rossum pencipta Python	132
24.2	tampilan awal instalasi	133
24.3	pilih folder tempat instalasi	134
24.4	komponen yang akan di install	135
24.5	tampilan proses instalasi	136
24.6	tampilan akhir instalasi	137
25.1	gambar yang menggambarkan keadaan variabel pada python	141
25.2	Tampilan Contoh Input/Output Tipe Data String	144
25.3	Tampilan Contoh Input/ Output Tipe Data Bilangan	145
25.4	kode program	145
25.5	Tampilan Contoh Konversi Tipe Data String dan Integer	145
25.6	Tanda operasi pada python	146
26.1	Contoh Penggunaan Break	152
26.2	Hasil penggunaan range	153
26.3	Hasil setelah di jalankan	154
26.4	Nilai Akhir Setelah run	154
31.1	Alamat Situs Qgis	173
31.2	Tampilan Situs Qgis	174
31.3	Pilihan Paket Aplikasi	174

31.4	Pilihan Sistem Operasi	174
31.5	File Installer Qgis	175
31.6	Awal Mulai Instalasi	175
31.7	Form Persetujuan Instalasi	176
31.8	Lokasi Instalasi	176
31.9	Pilihan Paket Instalasi	177
31.10	Proses Instalasi	177
31.11	Selesai Instalasi	178
31.12	Tampilan Awal Qgis	178
31.13	Label yang menumpuk	178
31.14	gambar toolbar yang ada pada QGis	178
31.15	gambar toolbar yang ada pada QGis	179
31.16	gambar status bar yang ada pada QGis	179
31.17	Gambar project pada Menu Bar	179
31.18	Gambar edit pada Menu Bar	179
31.19	Gambar view pada Menu Bar	180
31.20	Gambar setting pada Menu Bar	180
31.21	Gambar plugin pada Menu Bar	180
31.22	Gambar vektor pada Menu Bar	180
31.23	Gambar raster pada Menu Bar	181
31.24	Gambar database pada Menu Bar	181
31.25	Gambar pengolahan pada Menu Bar	181
31.26	Gambar bantuan pada Menu Bar	181
31.27	Toolbar Label tampak seperti ini	181
31.28	tombol Layer Labelling Options	182
31.29	tombol Layer this Label With	182
31.30	Field Name	182
31.31	Gambar konversi sistem koordinat dan proyeksi peta pada QGIS	182
31.32	Gambar setting tampilan peta pada QGIS	183
31.33	Gambar vektor pada qgis	183
32.1	ESRI.	185

DAFTAR TABEL

31.1	Paket dasar M4SW	183
------	------------------	-----

Bagian I

DEFINISI, TEORI DAN SEJARAH GEOSPASIAL

BAB 1

PENDAHULUAN DEFINISI

1.1 Definisi GIS (GEOGRAPHICS INFORMATION SYSTEM)

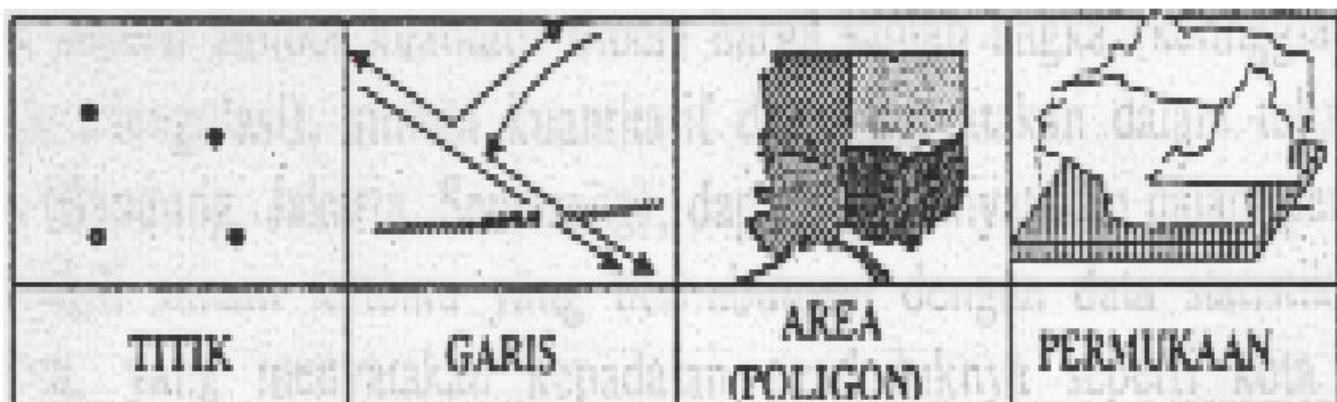
Geographical information system (GIS) adalah sebuah komputer yang berbasis sistem Informasi digunakan untuk memberikan informasi bentuk digital dan analisis terhadap Permukaan geografi bumi.

1.1.1 Pemahaman pada Geographic Information System GIS

Dimana GIS merupakan pemahaman dari, sebagai berikut:

1. Geography

Dimana GIS dibangun berdasarkan pada istilahgeografi atau spasial. Object mengacu pada spesifikasi lokasi dalam suatu tempat/ruang. Objek dapat berupa fisik, budaya ataupun ekonomi alamiah. Penampakan yang seperti ini ditampilkan pada suatu peta yang digunakan untuk memberikan gambaran yang lebih representatif dari spasial dari suatu objek. sesuai dengan kenyataannya yang terdapat di bumi. Dimana simbol, warna dan gaya garis digunakan sebagai perwakilan dari setiap spasial yang berbeda pada peta dua dimensi. Pada gambar 1.1 dijelaskan bahwa data spasial berikut berupa titik, garis, poligon (2-D) dan permukaan (3-D).



Gambar 1.1 data spasial berikut berupa titik, garis, poligon (2-D), permukaan (3-D).

Dan arti dari gambar di atas adalah:

- Format Titik
 - Memiliki koordinat tunggal

- Tanpa memiliki panjang
- Tanpa memiliki luasan
- Format Garis
- Memiliki koordinat titik awal dan akhir
- Memiliki panjang tanpa luasan
- Format Polygon
- Memiliki koordinat titik awal dan akhir
- Memiliki panjang dan luasan
- Format Permukaan
- Memiliki area koordinat vertikal
- Memiliki area dengan ketinggian

2. Information

Informasi berasal dari kata pengolahan sejumlah data. Di dalam GIS informasi mempunyai Volume terbesar. Dan setiap object geografi memiliki setting datanya tersendiri karena tidak sepenuhnya data yang ada dapat terwakili di dalam peta. Maka, semua data harus Diasosiasikan pada objek spasial yang mampu membuat peta menjadi intelligent.

3. System

Pengertian dari suatu sistem merupakan kumpulan elemen-elemen yang saling berintegrasi Dan berinterdependensi dalam sebuah lingkungan yang dinamis untuk mencapai tujuan tertentu.

1.1.2 Definisi GIS (Geography Information and System)

Dan definisi dari GIS dapat selalu berubah karena GIS adalah bidang kajian ilmu Dan teknologi yang masih baru. Beberapa definisi dari Geographical Information System yaitu:

1. Definisi GIS menurut (Rhind, 1988): Yaitu: GIS is a computer system for collecting, checking, integrating and analyzing Information related to the surface of the earth.
2. Definisi GIS menurut (Marble & Peuquet, 1983) and (Parker, 1988; Ozemoy et al., 1981; Burrough, 1986): Yaitu: GIS deals with space-time data and often but not necessarily, employs computer Hardware and Software.
3. Definisi GIS menurut (Purwadhi, 1994): - SIG adalah suatu sistem yang mampu mengorganisir perangkat keras (hardware), Perangkat lunak (software), dan data, serta dapat mendayai dan digunakan sistem Penyimpanan, pengolahan, maupun analisis data yang dilakukan secara simultan, sehingga dapat Diperoleh seluruh informasi yang berkaitan secara langsung dengan aspek keruangan. - SIG adalah manajemen data spasial dan data non-spasial yang berbasis komputer Dengan menggunakan tiga karakteristik dasar, yaitu:
 - Memiliki fenomena yang aktual (variabel data non-lokasi) dan berhubungan dengan topik permasalahan di lokasi bersangkutan.
 - Merupakan suatu kejadian di suatu lokasi tertentu.
 - Memiliki dimensi waktu. Alasan GIS dibutuhkan adalah karena untuk data spasial.

Penanganannya sangat sulit karena peta dan data statistik cepat mengalami kedaluwarsa Sehingga tidak ada pelayanan penyediaan data dan informasi yang diberikan menjadi tidak akurat.

Berikut merupakan keistimewaan analisis dengan Geographical Information System (GIS) yaitu:

1. Analisis Proximity Analisis Proximity adalah geografi yang berbasis pada jarak antar layer. Di dalam analisis proximity GIS menggunakan proses yang disebut dengan buffering Yaitu membangun lapisan pendukung sekitar layer dalam jarak tertentu agar dapat menentukan dekatnya hubungan antara sifat bagian yang ada.
2. Analisis Overlay Analisis Overlay adalah proses integrasi data dari lapisan-lapisan layer yang berbeda (overlay). Yang secara analisis membutuhkan lebih dari satu layer yang akan di tumpang susun secara fisik agar dapat di analisis secara visual.

Maka artikel: Dalam sebuah artikel dari husein yang menyebutkan bahwa GIS merupakan pemahaman dari Geography, Information dan System [1].

1.2 Geographic Information System (GIS): Introduction to the computer perspective

Sistem Informasi Geografi (GIS) diartikan sebagai sistem untuk menyimpan, memeriksa, mengintegrasikan, memanipulasi, menganalisis dan memaparkan data yang berkaitan dengan semua ruang yang berhubungan dengan keadaan bumi.

Maka artikel : Dalam sebuah artikel dari prahasta yang menyebutkan bahwa GIS merupakan menyimpan, memeriksa, mengintegrasikan, memanipulasi, menganalisis dan memaparkan data yang berkaitan dengan semua ruang yang berhubungan dengan keadaan bumi[2].

1.2.1 Pengenalan GIS atau Geography Information System

1. GIS atau dikenal dengan Sistem Informasi Geografi ditujukan sebagai sistem yang mampu menyimpan, memeriksa, mengintegrasikan, memanipulasi, menganalisis dan memaparkan data-data yang terkait dengan spasial yang merujuk terhadap bagian bumi. (Jabatan Alam Sekitar, 1987).
2. GIS merupakan satu set lat untuk mengumpulkan, menyimpan, mendapatkan, mengubah dan memaparkan data ruang dari keadaan bumi yang sebenarnya untuk keperluan tertentu (Burrough, 1986).
3. GIS adalah setiap set manual atau prosedur komputer yang digunakan untuk menyimpan dan memanipulasi data geografis yang tersedia (Arronoff, 1989).
4. GIS merangkum keadaan bumi dengan peranti atau perangkat tertentu yang digunakan untuk peta input atau peta produk, bersama-sama dengan dengan sistem komunikasi yang diperlukan untuk dijadikan sebagai penghubung berbagai unsur. (Star & Ester, 1990).
5. GIS adalah suatu sistem untuk membantu dalam membangunkan model tertentu yang mustahil untuk dijadikan sintesi data yang banyak. (Martin, 1996).

1.2.2 Komponen GIS atau Geography Information System

Komponen GIS sendiri dibagikan menjadi 3 komponen, yaitu: Sistem Komputer (perkakas dan sistem operasi), Software GIS (ArcGIS), database GIS, metode GIS (Prosedur analisis), People (Orang-orang yang menggunakan GIS/User). Pada gambar 1.2 dijelaskan bahwa komponen GIS sebagai berikut.

1.2.2.1 Komponen GIS atau Geography Information System Sesuai dengan gambar diatas komponen GIS dibagi menjadi 3 bagian, yaitu :

- Sistem Komputer (perkakas dan sistem operasi), merupakan hardware dari sebuah sistem GIS. Perkakas terdiri dari monitor, unit sistem atau CPU, keyboard dan mouse (Heywood et al., 2002). Teknologi komputer harus memiliki kemampuan kuasa yang tinggi untuk menjalankan perisian GIS.
- Software GIS, merupakan ArcGIS untuk tujuan perancangan, pengurusan ataupun pemodelan pada kebutuhan tertentu.
- Database GIS, merupakan tempat yang melibatkan data GIS baik data spatial dan pengurusan datanya. Memori untuk menyimpan jumlah data yang besar dan mempunyai kualitas yang baik dengan resolusi tinggi pada skrin grafik warna (untuk membantu dalam menentukan maklumat yang dihasilkan atau diberikan melalui penggunaan warna yang berbeda).
- Metode GIS, merupakan prosedur dari analisis sistem GIS yang melibatkan proses input, proses menyimpan, proses mengurus, proses menukar, proses menganalisis, dan proses output yang hanya melibatkan perisian GIS untuk mengatur sistem dan data-data tersebut (Heywood et al., 2002)
- People, merupakan orang-orang yang menggunakan sistem GIS, atau orang yang mengendalikan proses input-output sistem GIS.

1.2.3 Kaidah GIS atau Geography Information System

Data spatial, pengurusan data atribut, paparan data, penerokaan data, analisis dan pemodelan data GIS; Yang dijelaskan oleh gambar sebagai berikut:

1. Input data spatial Merupakan langkah awal agar terciptanya data baru, dengan cara menginputkan data dan sistem GIS akan menyuntingnya dalam bentuk transformasi geometri yang nantinya akan menghasilkannya kedalam bentuk hard copy. (Chang, 2008) (Heywood et al., 2002).
2. Pengurusan data artibut Merupakan langkah selanjutnya agar sumber peta dapat dipindahkan kepada peta digital yang dapat dibaca oleh GIS. (Chang, 2008) (Worboy & Duckham, 2003) (Heywood et al., 2002)
3. Pengumpulan data Merupakan aktivitas untuk proses melakukan eksplorasi lebih jauh dalam meneliti ciri kesamaan dalam suatu graf peta yang berbeda. (Worboy & Duckham, 2003).
4. Analisis data Merupakan cara untuk memaparkan dan memanipulasi data yang didapat. Dengan menggunakan 2 jenis format, yaitu :



Gambar 1.2 komponen GIS.

- data vektor : melibatkan beberapa kaidah seperti penimbalan / buffering, penindihan/overlay, pengukuran jarak, statik ruang, dan manipulasi peta.
- data raster : menganalisis pengumpulan data tempatan, kaidah kejiranian, kaidah berzon, dan kaidah operasi global. (Chang, 2008) (Worboy & Duckham, 2003) (Heywood et al., 2002)

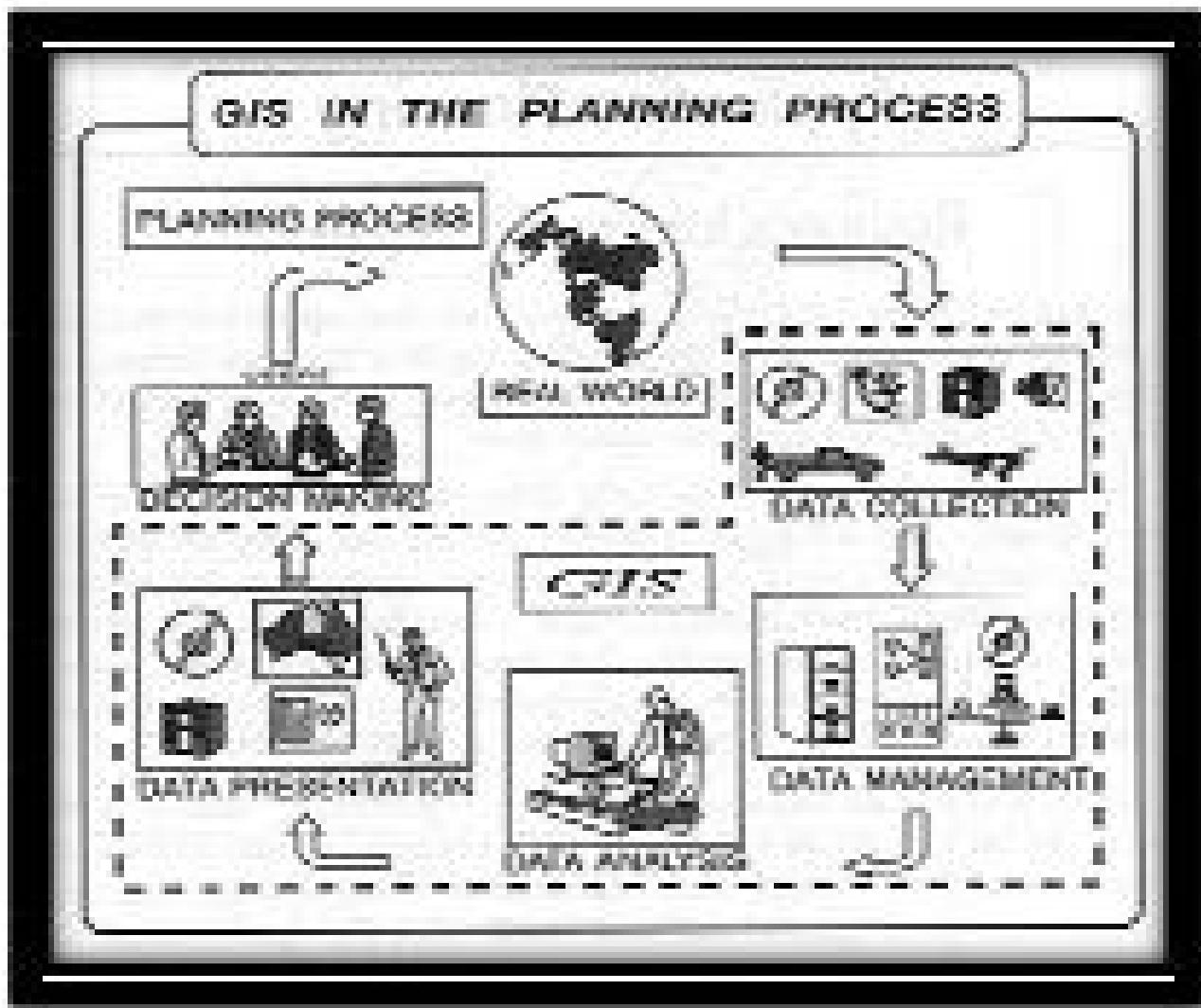
5. Paparan data dan output data Dasarnya disediakan untuk tujuan pemaparan hasil dari analisis data yang fungsinya ditujukan untuk pengguna.
6. Aplikasi GIS Digunakan untuk keperluan tertentu dan bersifat umum bagi masyarakat tergantung keperluan penggunanya. (Heywood et al., 2002).

Pada gambar 1.4 dijelaskan bahwa aplikasi GIS sesuai keperluan penggunaan sebagai berikut.

Maka artikel: Dalam sebuah artikel dari hua yang menyebutkan bahwa GIS memiliki kaidah dan komponen, Information dan System [3].

1.2.4 Kesimpulan GIS atau Geography Information System

Kesimpulannya, GIS merupakan alat yang penting dalam perspektif komputer pada masa kini dikarenakan GIS mempunyai kemampuan aplikasi dalam berbagai bidang, misalnya dalam proses perancangan bandar dan kartografi, penilaian kesan alam sekitar dan pengurusan sumber asli. GIS juga memainkan peranan dalam perspektif perniagaan, dimana alat ini sangat bermanfaat dalam pengiklanan dan pemasaran, jualan, dan logistik mampu digunakan untuk mencari dan meningkatkan perniagaan seperti tapak perniagaan yang strategi. Sebagai umum, pengguna GIS dapat dilibatkan dengan agensi-agensi penguat kuasaan undang-undang, strategi perancangan, perhutanan, industri, pemberdayaan alam, perencanaan kota, profesional telekomunikasi, kesehatan, pengangkutan, geografi, dan pembangunan pemasaran. Penjelasan ini menyediakan platform untuk memahami lebih lanjut tentang komponen, kaidah, dan aplikasi GIS, untuk mempelajari tentang alat GIS.



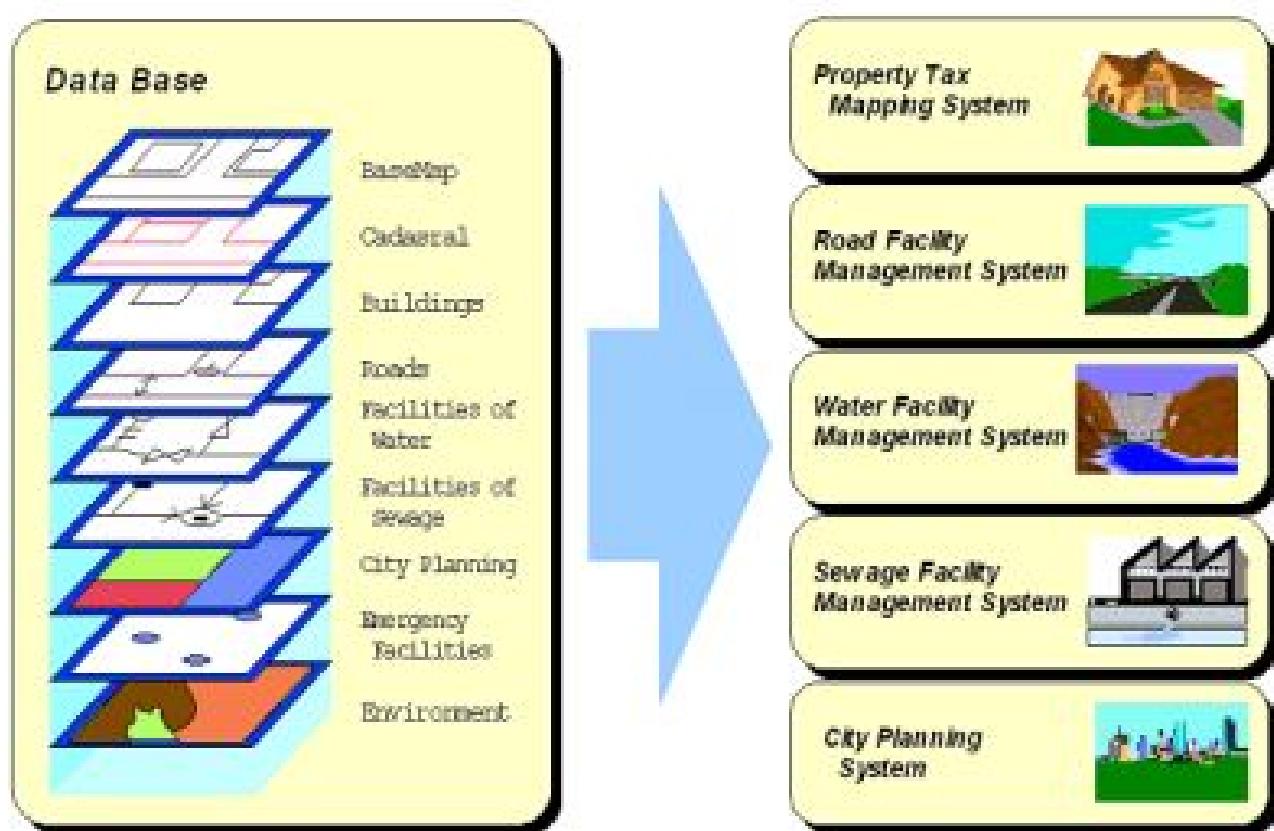
Gambar 1.3 kaidah GIS

1.2.5 Saran GIS atau Geography Information System

GIS dapat diaplikasikan di dalam kehidupan sehari-hari untuk memenuhi kebutuhan dan dapat membantu kebutuhan setiap masyarakat menjadi lebih baik dan lebih bermanfaat. Karena dengan memanfaatkan kemajuan teknologi maka teknologi yang digunakan akan ikut turut serta terus berkembang untuk menyesuaikan pemenuhan kebutuhan setiap pengguna yaitu masyarakat. Demikian kesimpulan dan saran yang dapat disampaikan kurang lebihnya mohon maaf dan terimakasih.

1.2.6 Tipe Data Geospasial

Ada dua tipe data geospasial. Data vektor dan data raster. Model data vektor merupakan model data yang paling banyak digunakan, model ini berbasiskan pada titik dengan koordinat (x,y) untuk membangun objek spasialnya. Objek yang dibangun dibagi menjadi tiga bagian, yaitu: titik, garis, dan area (polygon). Keuntungan dari data vector, yaitu: ketepatan dalam merepresentasikan fitur titik, batasan dan garis lurus. Data raster adalah data yang dihasilkan dari sistem pengindraan yang jauh. Pada data raster, objek geografis di representasi kan sebagai struktur sel grid yang disebut pixel. Resolusi pada data raster tergantung pada ukuran pixel-nya. Maka, resolusi pixel menggambarkan ukuran sebenarnya dari permukaan bumi yang diwakili oleh setiap pixel pada citra. Semakin tinggi resolusi nya, semakin kecil permukaan bumi yang di representasi kan oleh suatu sel. Data raster cocok untuk me representasi kan batas-batas yang berubah secara gradual, seperti jenis tanah, vegetasi, suhu tanah, dan kelembaban tanah.



Gambar 1.4 aplikasi GIS

BAB 2

PENDAHULUAN SEJARAH PTOLEMY

2.1 Peta

Peta adalah/merupakan penggambaran secara grafis atau bentuk skala (perbandingan) pada konsep mengenai bumi. Dalam hal ini peta merupakan alat untuk menyampaikan atau menginformasikan mengenai ilmu kebumian. Bagaimana peta dahulu ditemukan ? pengetahuan mengenai dasar pembentukan peta sama seperti filosofi, yang mana sering terdapat perbedaan.

2.1.1 Peta Menurut Claudius Ptolemaeus Ptolemy



Gambar 2.1 Gambar Peta menurut Ptolemy

Gambar 2.1 Berikut adalah gambar dari Peta yang dibuat oleh Claudius Ptolemy. Claudius Ptolemaeus yang dikenal dengan nama Ptolemy, hidup antara tahun 100 M dan 168 M, beliau merupakan salah satu sarjana sains pada masanya. Dia tinggal dan bekerja di Alexandria, kota Mesir yang merupakan pusat Intelektual dunia barat dengan perpustakaan paling luas yang pernah diciptakan. Ptolemy membawa semua pengetahuan dan keterampilan matematika dan astronomi dan menerapkannya pada pembuatan peta. Dia memiliki daya tarik matematikawan dengan presisi untuk menunjukkan hubungan satu tempat ke tempat lain. Berdasarkan perhitungan lingkaran dunia 18.000 mil, ia juga mengembangkan sistem grid latitude dan longitude yang dirancang oleh Marinus of Tire. Sementara beberapa rincian peta mungkin sedikit aneh dengan garis lintang sejajar dengan garis khatulistiwa dengan garis bujur yang membentang ke utara-selatan dengan busur anggun, sudah tidak asing lagi bagi siapa saja yang pernah memiliki atlas. Dalam kerangka ini, ptolemy mampu membangun koordinat dan mendaftarkan lebih dari 8000 tempat dengan koordinat masing-masing. Bagi ptolemy, ini adalah latihan matematik dan kita tidak akan pernah tahu apakah dia benar-benar menggambar peta dari sini. Data-data tentang Pembuatan peta sempat hilang ketika perpustakaan Alexandria yang terkenal dibakar oleh orang-orang kristen fanatik pada tahun 390 Masehi - sebuah contoh awal konflik antara iman dan sains. Tapi setidaknya satu salinan yang telah dibuat dari karya Ptolemy terselamatkan dan ini bertahan di Byzantium. 1000 tahun berlalu dan

kemudian tulisannya digunakan untuk dikembangkan oleh ilmuwan Arab, sementara di bagian Eropa tetap dalam ketidaktahanan akan warisannya. Baru pada saat renaisans muncul di Italia dan daya tarik di dunia, Geografi Ptolemy diterjemahkan dalam bahasa Latin dan gagasannya terhadap PETA Dunia dapat diakses oleh para ilmuwan. Namun tidak ada peta dalam keadaan masih utuh, hanya petunjuk dan saran untuk pembuatan map dan daftar koordinat [4]. gambar 2.2 Foto Ptolemy seorang ahli astronomi,



Gambar 2.2 Foto Ptolemy

geografi, matematikawan.

2.1.2 Peta Dunia Ptolemy

Peta dunia Ptolemy adalah peta gambaran dunia yang diketahui masyarakat barat pada kurun kedua masihi. Peta tersebut berdasarkan penerangan yang terkandung di dalam buku *Geographia*, ditulis kira-kira pada 150 masehi. Walaupun peta autentik tidak dijumpai, buku *Geographia* yg mengandungi beribu-ribu rujukan pelbagai tempat di dunia lama, beserta koordinat, yang membolehkan para pelukis peta menyusun semula peta dunia Ptolemy apabila manu skriptnya telah ditemui sekitar 1300 masehi. gambar 2.3 Peta Dunia Ptolemy, disusun semula dari *Geographia Ptolemy* (kira-kira 150) pada kurun ke 15, menunjukan Sinae (China) di sebelah kanan, di bawah pulau Taprobane (Sri Lanka, diperbesarkan) dan Aurea Chersonesus (Semenanjung Asia Tenggara).

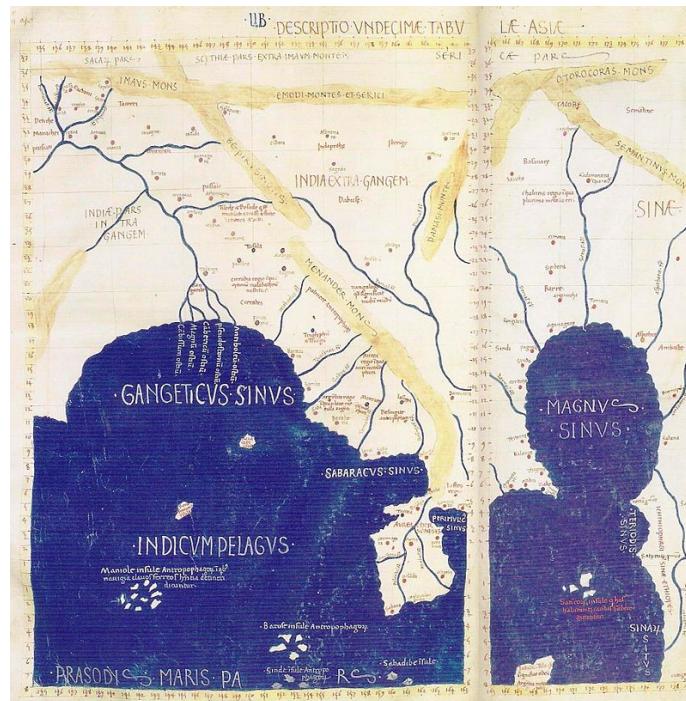
Gambar 2.4 adalah perinician Timur dan Asia Tenggara dalam peta dunia Ptolemy. Teluk Ganges (Teluk Bengali) kiri, Semenanjung Asia Tenggara di tengah, Laut China selatan kanan, bersama Sinae (China).

2.1.3 Sejarah Ptolemy

Gambar 2.5 adalah Claudius Ptolemaeus (90 - 168), adalah seorang ahli geografi, astronom, dan astrolog yang hidup pada zaman Helenistik di provinsi Romawi, Aegyptus. Ptolemaeus adalah pengarang beberapa risalah ilmiah, tiga di antaranya kemudian memainkan peranan penting dalam keilmuan Islam dan Eropa. Yang pertama adalah risalah astronomi yang dikenal sebagai *Almagest* (dalam bahasa Yunani Η νεγάληΣύνταξις c , Risalah Besar). Yang kedua adalah *Geographia*, yang merupakan diskusi teliti mengenai pengetahuan geografi Helenistik. Yang ketiga adalah risalah astrologi dikenal sebagai *Tetrabiblos* (Empat buku) di mana dia berusaha mengadaptasi astrologi horoskop ke filosofi alam Aristotelian. Ia juga melestarikan daftar raja-raja kuno, disebut *Kanon Ptolemaeus*, yang penting bagi penelitian sejarah Timur Tengah. Claudius adalah nomen (nama keluarga) seorang Roma; Ptolemaeus menyandang nama itu, sehingga menjadi bukti bahwa dia adalah seorang warga negara Roma. Sesuai kebiasaan, Keluarga Ptolemy pertama yang menjadi warga negara (entah itu dia atau nenek moyangnya) mengambil nama itu dari seorang Roma yang bernama Claudius, sehingga membuatnya diberi status kewarganegaraan. Jika orang Roma ini adalah kaisar,



Gambar 2.3 Gambar Ptolemy



Gambar 2.4 Gambar Perincian Timur dan Asia

kewarganegaraan sudah akan diberi di antara tahun 41 dan 68 M. (waktu Claudius, lalu Nero, menjabat kaisar). Astronom itu juga mempunyai praenomen (nama pertama), yang tetap tak diketahui. Tetapi, kemungkinan Tiberius, karena praenomen itu sangat umum di antara yang keluarga-keluarga yang diberi kewarganegaraan oleh kaisar ini. Ptolemaeus (Ptolemy) adalah sebuah nama Yunani. Muncul satu kali di mitologi Yunani, dalam bentuk Homeric. Cukup biasa di antara golongan sol bagian atas Makedonia pada saat Alexander Agung, dan ada beberapa di antara tentara Alexander, satu di antaranya pada tahun 323 S.M. menjadikan dirinya sendiri Raja Mesir: Ptolemy I Soter; semua raja setelah dia, sampai Mesir menjadi provinsi Roma pada tahun 30 S.M., adalah juga dari dinasti Ptolemaic. Hanya ada sedikit bukti tentang subyek asal usul Ptolemy (meskipun melihat di atas kewarganegaraan Roma keluarganya), tetapi kebanyakan sarjana dan sejarawan mempertimbangkannya tak mungkin bahwa Ptolemeus berhubungan dengan dinasti kerajaan Ptolemies. Selain dianggap sebagai seorang anggota masyarakat Yunani Alexandria, hanya sedikit rincian hidup Ptolemaeus yang diketahui. Dia menulis dalam bahasa Yunani Kuno dan diketahui sudah menggunakan data astronomis Babilonia. Seorang warganegara Roma, beberapa sarjana menyimpulkan bahwa secara etnik, Ptolemeus adalah orang Yunani, dan sarjana lainnya berpendapat bahwa dia secara etnik orang Mesir, meskipun Hellenize. Dia banyak dikenal dalam sumber bahasa Arab yang muncul kemudian sebagai Upper Egyptian, diperkirakan dia mungkin berasal dari Mesir selatan. Astronom, ahli ilmu bumi, dan pakar fisika Arab selanjutnya merujuk padanya menggunakan nama Arabnya Batlamus. Karya utama Ptolemy lainnya adalah Geografi nya (juga disebut Geography), kompilasi koordinat geografis dari bagian dunia yang dikenal oleh Kekaisaran Romawi pada masanya. Dia agak bergantung pada karya seorang ahli geografi sebelumnya, Marinus of Tire, dan pada kano Romawi dan Kekaisaran Persia kuno. [Rujukan] Dia juga mengakui astronom Hipparchus kuno karena telah menyediakan ketinggian kutub utara untuk beberapa kota.[19]



Gambar 2.5 Konsep artis zaman pertengahan dari Claudius Ptolemaeus.

2.1.4 The Geography

gambar 2.6 Bagian pertama dari Geografi adalah diskusi tentang data dan metode yang dia gunakan. Seperti model tata surya di Almagest, Ptolemy memasukkan semua informasi ini ke dalam skema besar. Setelah Marinus, dia memberikan koordinat ke semua tempat dan fitur geografis yang dia ketahui, dalam kotak yang membentang di seluruh dunia. Lintang diukur dari khatulistiwa, seperti sekarang, tapi Ptolemy lebih suka [20] untuk mengekspresikannya sebagai climata, panjang hari terpanjang daripada deretan busur: panjang hari Senin mulai meningkat dari 12h menjadi 24 jam saat seseorang pergi. dari khatulistiwa ke lingkaran kutub. Dalam buku 2 sampai 7, dia menggunakan gelar dan meletakkan garis meridian 0 bujur di tanah paling barat yang dia kenal, Kepulauan Terberkati, yang sering di identifikasi sebagai Kepulauan Canary, seperti yang disarankan oleh lokasi dari enam titik yang diberi label FORTUNATA pulau-pulau di dekat ekstrem kiri laut biru peta Ptolemeus di sini di reproduksi. Ptolemeus juga merancang dan memberikan petunjuk bagaimana membuat peta di seluruh dunia yang berpenghuni (oikoumen) dan provinsi Romawi. Di bagian kedua Geografi, dia memberikan daftar topografi yang diperlukan, dan teks untuk peta. Oikoumennya membentang 180 derajat bujur dari Kepulauan Terberkati di Samudra Atlantik sampai ke Cina tengah, dan sekitar 80 derajat garis lintang dari Shetland menjadi anti-Meroe (pantai timur Afrika); Ptolemy sangat sadar bahwa dia tahu tentang hanya seperempat dunia, dan perpanjangan yang salah dari Cina ke selatan menunjukkan sumbernya tidak sampai ke Samudra Pasifik. Peta di manuskrip yang masih ada di Ptolemy's Geography, bagaimanapun, hanya berasal dari sekitar tahun 1300, setelah teks tersebut ditemukan kembali oleh Maximus Planudes. Tampaknya tabel topografi dalam buku 2-7 adalah teks kumulatif - teks yang diubah dan ditambahkan sebagai pengetahuan baru yang tersedia di abad setelah Ptolemy. [21] Ini berarti bahwa informasi yang terdapat di berbagai bagian Geografi kemungkinan berasal dari tanggal yang berbeda. Peta berdasarkan prinsip ilmiah telah dibuat sejak zaman Eratosthenes, pada abad ke-3 SM, namun Ptolemy memperbaiki proyeksi peta. Diketahui dari sebuah pidato oleh Eumenius bahwa peta dunia, sebuah orbis pictus, yang tidak diragukan lagi berdasarkan Geografi, di pajang di sebuah sekolah di Augustodunum, Gaul pada abad ketiga. [22] Pada abad ke-15, Geografi Ptolemy mulai dicetak dengan peta terukir; edisi cetak paling awal dengan peta terukir di produksi di Bologna pada 1477, diikuti dengan cepat oleh edisi Romawi tahun 1478 (Campbell, 1987). Sebuah edisi yang dicetak di Ulm pada tahun 1482, termasuk peta tebing kayu, adalah yang pertama dicetak di utara Pegunungan Alpen. Peta terlihat terdistorsi bila dibandingkan dengan peta modern, karena data Ptolemy tidak akurat. Salah satu alasannya adalah Ptolemy memperkirakan ukuran Bumi terlalu kecil: sementara Eratosthenes menemukan 700 stadion untuk sebuah lingkaran besar di dunia, Ptolemy menggunakan 500 stadion di Geografi. Sangat mungkin bahwa ini adalah stadion yang sama, karena Ptolemy beralih dari skala sebelumnya ke yang terakhir antara Syntaxis dan Geography, dan menyesuaikan derajat bujur yang sesuai. Lihat juga unit pengukuran dan Sejarah Yunani Kuno geodesi. Karena Ptolemy berasal dari garis lintang utamanya dari nilai terpanjang minyak mentah, garis lintang nya rata-rata keliru kira-kira satu derajat (2 derajat untuk Bizantium, 4 derajat untuk Kartago), meskipun para astronom kuno mampu mengetahui garis lintang mereka lebih lama. Dia setuju (Geografi 1.4) bahwa bujur paling baik ditentukan oleh observasi simultan gerhana bulan, namun dia sangat tidak berhubungan dengan ilmuwan pada masanya bahwa dia tidak mengetahui data semacam itu. lebih baru dari 500 tahun sebelumnya (Arbela gerhana). Ketika beralih dari 700 stadia per derajat ke 500, dia (atau Marinus) memperluas perbedaan bujur antara kota-kota yang sesuai (sebuah titik yang pertama kali di realisasi kan oleh P.Gosselin pada tahun 1790), yang mengakibatkan peregangan skala bumi timur-barat yang serius dalam derajat, meski tidak jauh. Mencapai garis bujur yang sangat tepat tetap menjadi masalah



Gambar 2.6 Geography by Ptolemy

dalam geografi sampai penerapan metode bulan Jovian Galileo di abad ke-18. Harus ditambahkan bahwa daftar topografi nya yang asli tidak dapat direkonstruksi: tabel panjang dengan angka dikirim ke anak cucu melalui salinan yang mengandung banyak kesalahan juru tulis, dan orang selalu menambahkan atau memperbaiki data topografi: ini adalah kesaksian akan popularitas yang terus-menerus dari Karya ini berpengaruh dalam sejarah kartografi.

BAB 3

PENDAHULUAN SEJARAH ERATOSTHENES

3.1 Eratosthenes

Lebih dari 2000 tahun yang lalu Eratosthenes membandingkan posisi Matahari di dua lokasi untuk menentukan ukuran bumi dengan alasan yang akurat. Berdasarkan [5] Eratosthenes lahir di Yunani pada koloni yang bernama Cyrene, sekarang disebut kota Shahhat, Libya. Sebagai pemuda, dia pergi ke Athena demi melanjutkan studinya. Lalu ia kembali ke Cyrene, dan membuat nama untuk dirinya sendiri untuk kebutuhan ilmiah sehingga penguasa Yunani di Mesir membawanya ke Aleksandria. Seorang pria yang mempunyai banyak talenta, Eratosthenes adalah seorang pustakawan, geografer, matematikawan, astronom, sejarawan dan penyair. Teman-temannya di perpustakaan menjulukinya sebagai Pentathlos atau atlit yang berkompetisi dalam lima acara yang berbeda. Julukan itu sepertinya cocok ditujukan kepada untuk seorang penerima beasiswa dari banyak bidang studi. Banyak dari tulisan karya Eratosthenes telah hilang tetapi ada beberapa orang yang melaporkan pekerjaannya telah ditemukan[6].

3.1.1 Bapak Geografi

Eratosthenes melanjutkan pengetahuannya tentang Bumi. Dengan menggunakan penemuan dan pengetahuan tentang ukuran dan bentuknya, dia mulai membuat sketsa. Di Perpustakaan Alexandria ia memiliki akses ke berbagai buku perjalanan, yang berisi berbagai bentuk informasi dan representasi dunia. [7] Dalam karya jilid tiga nya Geografi (bahasa Yunani: Geographika), dia menggambarkan dan memetakan seluruh dunia yang dikenalnya, bahkan membagi Bumi menjadi lima zona iklim: [8] dua zona pembekuan di sekitar kutub, dua zona beriklim sedang, dan sebuah zona yang meliputi khatulistiwa dan daerah tropis. Ia telah menemukan geografi. Dia menciptakan terminologi yang masih digunakan sampai sekarang. Dia meletakkan grid garis tumpang tindih di atas permukaan Bumi. Dia menggunakan kesejarahan dan garis meridian untuk menghubungkan setiap tempat di dunia. Sekarang mungkin untuk memperkirakan jarak seseorang dari lokasi dasar jaringan ini di atas permukaan Bumi. Dalam Geografi nama-nama lebih dari 400 kota dan lokasi mereka ditunjukkan: Sayangnya, Geografi-nya telah hilang dari sejarah, Pliny, Polybius, Strabo, dan Marcianus. [9] Buku pertama adalah pengantar dan memberikan ulasan tentang pendahulunya, mengakui kontribusi mereka yang dia susun di perpustakaan. Dalam buku ini Eratosthenes mengelam Homer karena tidak memberikan wawasan tentang apa yang sekarang digambarkannya sebagai geografi. Tidak setuju terhadap topografi Homer membuat marah banyak orang yang percaya bahwa dunia digambarkan dalam Odyssey sebagai hal yang sah. [10] Bumi adalah dunia yang tak tergoyahkan; Sementara di permukaannya ada tempat yang sedang berubah. Dia telah berhipotesis bahwa pada suatu waktu Mediterania adalah danau besar yang menutupi negara-negara yang mengelilinginya; dan hanya terhubung ke barat saat sebuah lorong telah dibuka dalam sejarahnya. Dalam buku kedua adalah penemuannya tentang keliling bumi. Di sinilah, menurut Pliny, "Dunia digenggam." Eratosthenes menggambarkan kisah terkenalnya tentang sumur di Syene, yang dijelaskan di atas. Buku ini akan dianggap sebagai teks tentang geografi matematika. Buku ketiganya tentang Geografi berisi geografi politik. Dia mengutip negara-negara dan menggunakan garis sejajar untuk membagi peta menjadi beberapa bagian, untuk memberikan deskripsi alam yang akurat. Ini adalah terobosan, dan bisa dianggap sebagai awal geografi. [7]

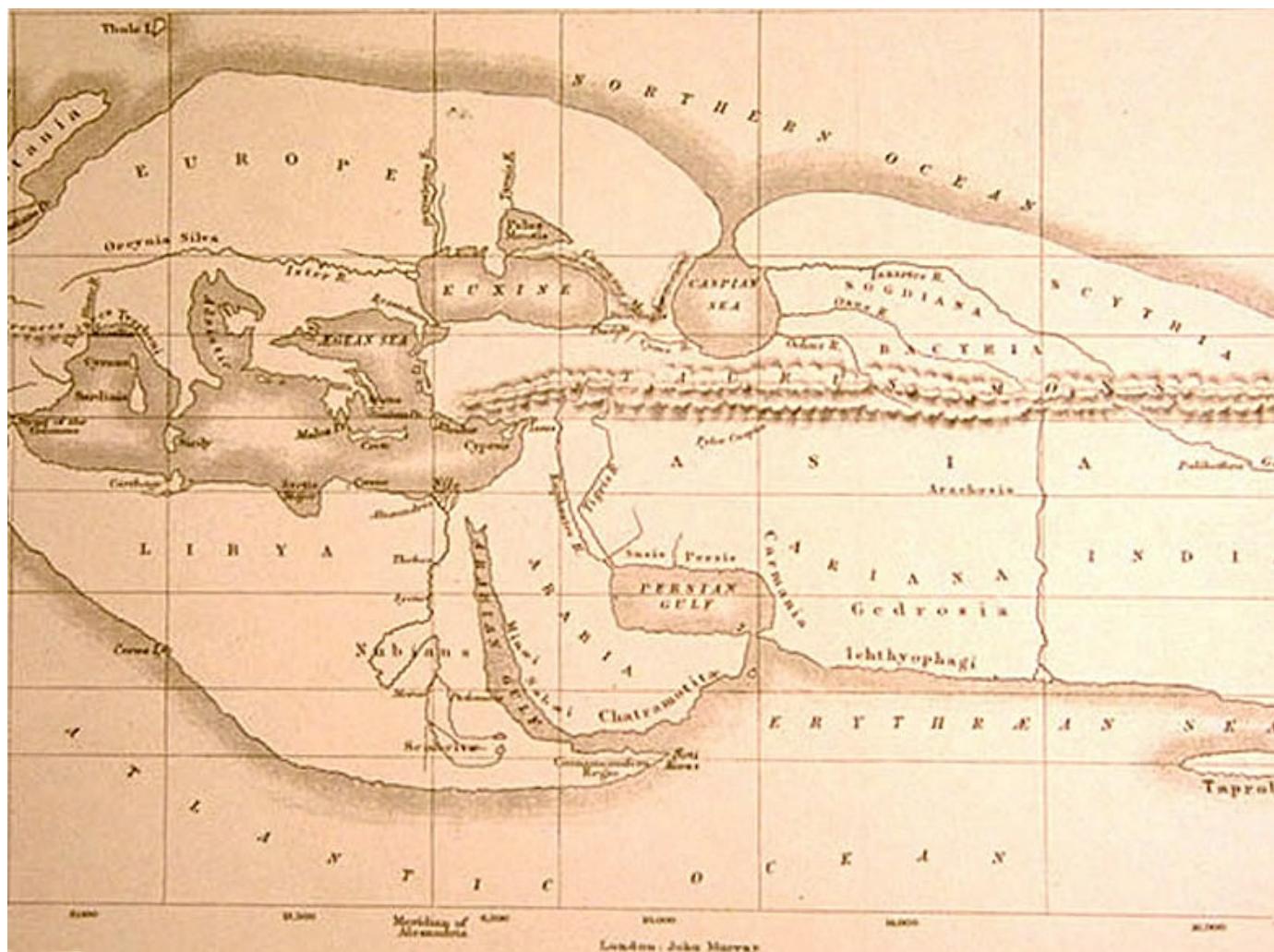
3.1.2 Mempelajari Bumi

Eratosthenes mungkin orang pertama yang menggunakan kata geografi. Dia menciptakan sistem garis bujur dan garis lintang dan membuat peta dari dunia yang sekarang dikenal. Dia juga merancang sistem untuk menemukan penomoran utama bilangan bulat



Gambar 3.1 ilustrasi yang tidak bertanggal dari para ilmuwan di Perpustakaan Alexandria Bettmann / CORBIS

yang hanya dapat dibagi sendiri atau dengan angka 1. Metode ini, masih digunakan sampai hari ini. Pada saat itu, Eratosthenes mengusulkan sebuah algoritma sederhana untuk menemukan bilangan prima. Algoritma ini dikenal dalam matematika sebagai "Sieve of Eratosthenes." Dalam matematika, Sieve of Eratosthenes, salah satu dari sejumlah saringan bilangan prima, adalah algoritma kuno yang sederhana untuk menemukan semua bilangan prima sampai batas tertentu. Hal itu terjadi karena secara iteratif menandai sebagai komposit, yaitu, tidak prima, kelipatan masing-masing prima, dimulai dengan kelipatan 2. Kelipatan bilangan prima yang diberikan dihasilkan mulai dari yang prima, sebagai urutan angka dengan perbedaan yang sama, sama dengan yang prima, antara angka berurutan. Ini adalah perbedaan kunci saringan dari penggunaan divisi percobaan untuk menguji secara berurutan setiap nomor kandidat untuk dibagi masing-masing prima. Eratosthenes juga yang pertama menghitung sumbu kemiringan bumi, yang dia pikir dengan tingkat akurasi yang tinggi; temuan dilaporkan oleh Ptolemy (85-165 M). Eratosthenes juga menghitung jarak dari bumi ke bulan dan ke matahari, tetapi dengan tingkat akurasi yang rendah. Ia membuat katalog 675 bintang. Ia membuat kalender dengan tahun kabisat dan meletakkan dasar kronologi di dunia barat dengan penyelenggaraan tanggal sastra dan politik kegiatan dari pengepungan Troy (sekitar 1194 - 1184 SM) ke dalam waktu ia sendiri. Namun, pencapaianannya yang paling abadi adalah perhitungan lingkar bumi yang sangat akurat (jarak di sekitar lingkaran atau bola). Dia menghitung ini dengan menggunakan geometri dan trigonometri sederhana dan dengan mengenali Bumi sebagai bola di ruang angkasa. Sebagian besar ilmuwan Yunani pada masa Aristoteles (384-322 SM) sepakat bahwa Bumi adalah sebuah bola, tapi tidak ada yang tahu seberapa besar itu. Bagaimana para ilmuwan Yunani mengetahui bahwa bumi adalah sebuah lingkungan? Mereka mengamati bahwa kapal menghilang di atas cakrawala sementara tiang-tiang mereka masih terlihat. Mereka melihat bayangan melengkung Bumi di Bulan selama gerhana bulan. Dan mereka melihat perubahan posisi bintang di langit.

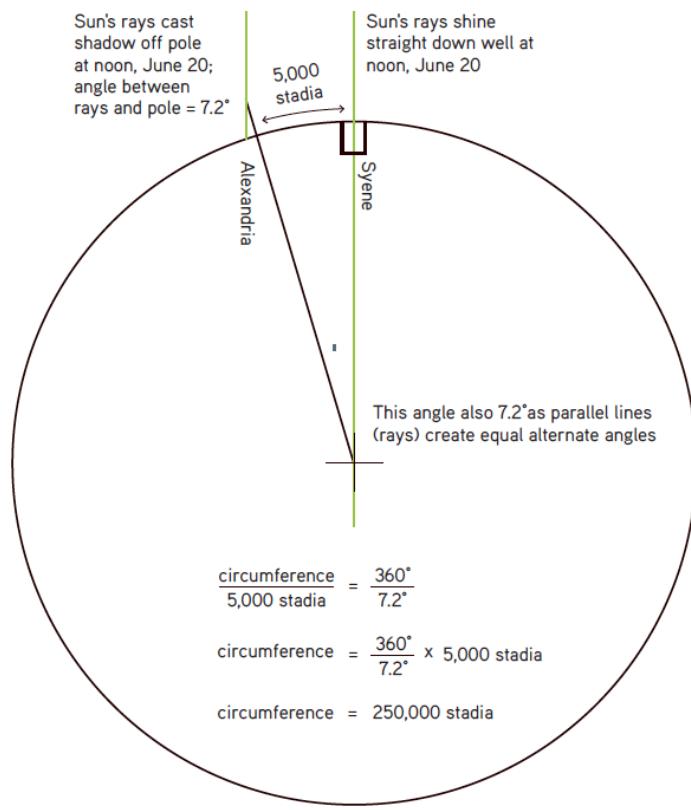


Gambar 3.2 Rekonstruksi Eratosthenes c. 194 SM peta dunia, dari E.H. Bunbury's 1883 Sejarah Geografi Kuno di antara orang Yunani dan Romawi dari Abad Pertengahan sampai Kejatuhan Kekaisaran Romawi, ranah publik

3.1.3 Mengukur Bumi

Eratosthenes mendengar tentang sumur terkenal di kota Swenet Mesir (Syene dalam bahasa Yunani, dan sekarang dikenal sebagai Aswan), di Sungai Nil. Siang hari satu hari setiap tahun - titik balik matahari musim panas (antara 20 Juni dan 22 Juni) - sinar matahari bersinar langsung ke dalam lubang dalam. Mereka hanya menyinari air di bagian bawah, bukan sisi sumur seperti pada hari-hari lain, membuktikan bahwa Matahari langsung di atas kepala. (Syene terletak sangat dekat dengan apa yang kita sebut Tropic of Cancer, 23,5 derajat ke utara, garis lintang paling utara di mana Matahari selalu berada di atas kepala pada siang hari.) Eratosthenes mendirikan sebuah tiang di Alexandria, dan pada titik balik matahari musim panas ia mengamati bahwa ia melemparkan bayangan, membuktikan bahwa Matahari tidak berada di atas kepala tetapi sedikit ke selatan. Mengakui kelengkungan Bumi dan mengetahui jarak antara kedua kota memungkinkan Eratosthenes untuk menghitung lingkar planet. [11] Eratosthenes bisa mengukur sudut sinar matahari dari vertikal dengan membagi panjang kaki di seberang sudut (panjang bayangan) dengan kaki yang bersebelahan dengan sudut (tinggi tiang). Ini memberinya sudut 7,16 derajat. Dia tahu bahwa lingkar bumi membentuk lingkaran 360 derajat, jadi 7,12 (atau 7,2, untuk membagi 360 secara merata sampai 50) derajatnya kira-kira seper lima puluh keliling. Dia juga tahu perkiraan jarak antara Alexandria dan Syene, jadi dia bisa mengatur persamaan ini:

Eratosthenes memperkirakan jarak dari Alexandria ke Syene sebanyak 5.000 stadion, atau sekitar 500 mil (800 kilometer). Dia membuat estimasi ini dari saat dibutuhkan pejalan kaki, yang dilatih untuk mengukur jarak dengan mengambil langkah reguler, untuk berjalan-jalan di antara kota-kota. Dengan memecahkan persamaan tersebut, dia menghitung keliling 250.000 stadia, atau 25.000 mil (40.000 kilometer) seperti pada gambar 3.3. Beberapa sumber kesalahan meremehkan perhitungan Eratosthenes dan interpretasi kita terhadapnya. Untuk satu hal, dia menggunakan unitnya untuk mengukur satuan stadion Yunani atau stadion atletik. Tapi tidak semua stadion dibangun dengan panjang yang sama. Di Yunani sebuah stadion setara dengan 185 meter (607 kaki), sedangkan di Mesir stadion berjarak sekitar 157,5 meter (517 kaki). Kami tidak tahu unit mana yang digunakan Eratosthenes. Jika dia menggunakan ukuran Yunani, perhitungannya akan turun sekitar 16 persen. Jika dia menggunakan orang Mesir, kesalahannya akan berada di bawah 2 persen dari lingkar bumi sebenarnya dari 24.860 mil (40.008 kilometer). Satu abad setelah Eratosthenes, astronom Yunani Posidonius dari Rhodes (sekitar 135-51 SM) menghitung lingkar bumi. Posidonius menggunakan bintang Canopus sebagai kerangka acuan: ketika bintang tersebut terlihat di cakrawala di Rhodes, itu adalah 7,5 derajat di atas cakrawala di Alexandria. Perhitungan pertamanya hampir benar, tapi dia merevisi jarak antara Rhodes dan Alexandria, yang menghasilkan angka yang sebanding dengan sekitar 18.000 mil (sekitar 29.000 kilometer), sekitar 28 persen lebih kecil dari lingkar sebenarnya. Ptolemy melaporkan perhitungan Posidonius dan bukan kata-kata Eratosthenes, dan tulisan Ptolemy inilah



Gambar 3.3 Sebuah diagram yang menunjukkan bagaimana Eratosthenes mengukur Bumi, diakses dari Simon Fraser University Online

yang menemukan jalan mereka menuju Christopher Columbus. Jika Ptolemy telah menggunakan sosok Eratosthenes yang lebih besar dan lebih akurat untuk keliling bumi, Columbus mungkin tidak akan pernah berlayar ke barat. Eratosthenes hidup sampai usia 82 tahun, saat dia kelaparan sampai mati karena dia takut pada awitan kebutaan.

3.1.4 Prestasi

Eratosthenes digambarkan oleh Leksikon Suda sebagai (Pentathlos) yang dapat diterjemahkan sebagai “All-Rounder”, karena ia ahli dalam berbagai hal: Dia adalah seorang polymath sejati. Dia dijuluki Beta karena dia hebat dalam banyak hal dan mencoba mendapatkan setiap informasi, namun tidak pernah mencapai peringkat tertinggi dalam segala hal; Straboaccounts Eratosthenes sebagai matematikawan di antara ahli geografi dan ahli geografi di kalangan matematikawan.

- Eusebius dari Kaisarea dalam bukunya *Preparatio Evangelica* mencakup sebuah bab singkat dari tiga kalimat pada jarak selesstial (Kitab XV, Bab 53). Dia menyatakan dengan sederhana bahwa Eratosthenes menemukan jarak ke Matahari untuk menjadi stadion miriad 400 dan 80.000 dan jarak ke Bulan menjadi 780.000 stadion. Ungkapan jarak ke Matahari telah diterjemahkan sebagai 4.080.000 stadia (1903 terjemahan oleh E. H. Gifford), atau sebagai 804.000.000 stadia (edisi Edouard des Places, bertanggal 1974-1991). Artinya tergantung pada apakah Eusebius berarti 400 segudang ditambah 80.000 atau “400 dan 80.000” segudang. Dengan stadion 185 m, 804.000.000 stadion adalah 149.000.000 Km, kira-kira jaraknya dari Bumi ke Matahari.
- Menurut [7] Eratosthenes juga menghitung diameter Matahari. Menurut Macrobius, Eratosthenes membuat diameter Matahari menjadi sekitar 27 kali lipat dari Bumi. Angka sebenarnya kira-kira 109 kali.
- Selama berada di Perpustakaan Alexandria, Eratosthenes merancang sebuah kalender dengan menggunakan ramalannya tentang ekliptika Bumi. Dia menghitung bahwa ada 365 hari dalam setahun dan setiap tahun keempat akan ada 366 hari.
- Dia juga sangat bangga dengan solusinya untuk Menggandakan Cube. Motivasi adalah bahwa ia ingin menghasilkan ketapel. Eratosthenes membangun perangkat garis mekanis untuk menghitung kubus, yang disebut mesolabio. Dia mendedikasikan solusinya untuk Raja Ptolemy, menyajikan sebuah model di perunggu dengan sebuah surat dan sebuah epigram. Archimedes adalah teman Eratosthenes dan dia juga mengerjakan alat perang dengan matematika. Archimedes mendedikasikan bukunya *The Method to Eratosthenes*, mengetahui cintanya untuk belajar dan matematika. [12]

3.1.5 Pekerjaan

Eratosthenes adalah salah satu tokoh ilmiah paling terkemuka di masanya, dan menghasilkan karya-karya yang mencakup pengetahuan luas sebelum dan selama waktunya di Perpustakaan. Dia menulis banyak topik - geografi, matematika, filsafat, kronologi, kritik sastra, tata bahasa, puisi, dan bahkan komedi lama. Sayangnya, hanya ada sisa fragmen karya-karyanya setelah Penghancuran Perpustakaan Alexandria. Judul karya yang dihasilkan oleh Eratosthenes [13] diantaranya :

1. Platonikos
2. Hermes
3. Erigone
4. Chronographies
5. On the Measurement of the Earth (hilang)
6. Geographika (hilang, dikritik oleh Strabo)
7. Arsinoes (sebuah memoar Ratu Arsinoe; hilang;)
8. Ariston (tentang Aristo dari Chios kecanduan kemewahan); kalah; dikutip oleh Athenaeus di *Deipnosophistae*)

PENDAHULUAN ALIDDRSSI

4.1 Peta

Peta merupakan penggambaran secara grafis atau bentuk skala (perbandingan) pada konsep mengenai bumi. Dalam hal ini peta merupakan alat untuk menyampaikan atau menginformasikan mengenai ilmu kebumian. bagaimana peta dahulu ditemukan ? pengetahuan mengenai dasar pembentukan peta sama seperti filosofi, yang mana sering terdapat perbedaan.

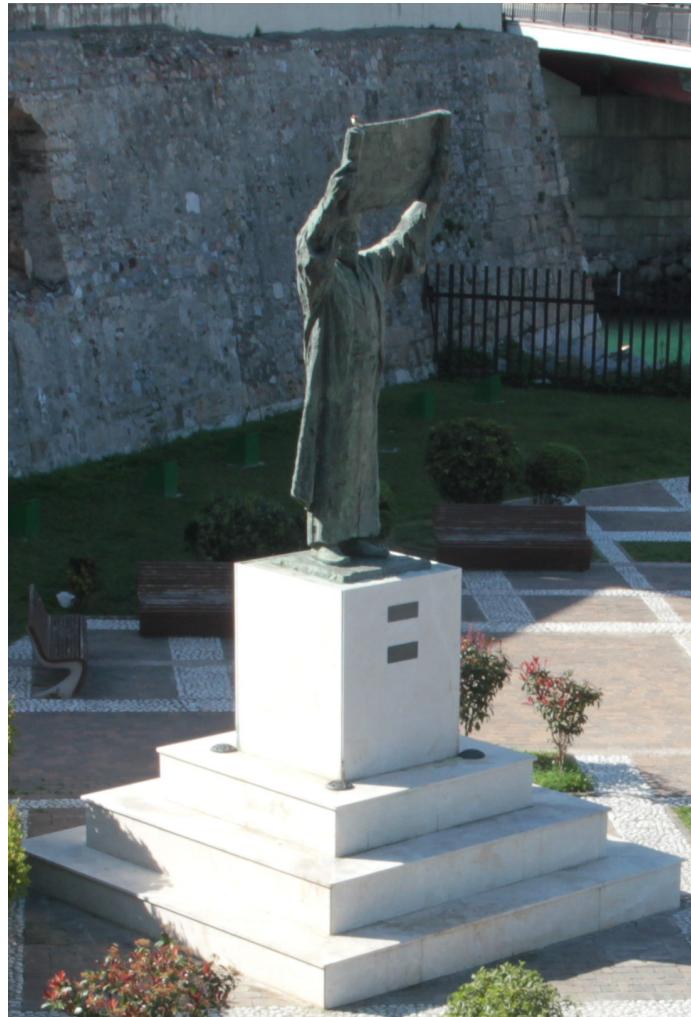
4.1.1 Peta Menurut Abu Abdullah Muhammad al-Idrisi al-Qurtubi al-Hasani as-Sabti ‘al-Idrisi’



Gambar 4.1 Gambar Peta al-Idrisi tahun 1154 berdasarkan French National Library

Peta yang dibuat oleh Muhammad al - Idrisi terlihat di Gambar 4.1. Abu Abdulla Muhammad al-Idrisi al-Qurtubi al-Hasani as-Sabti yang dikenal dengan nama al-Idrisi, hidup antara tahun 1100 - 1165. Al-Idrisi lahir di keluarga Hammudid besar Afrika Utara dan Al-Andalus, yang mengklaim turun dari Idrisids of Morocco dan akhirnya nabi Muhammad. Al-Idrisi lahir di kota Ceuta, di mana kakek buyutnya terpaksa menetap setelah jatuhnya Hammudid Mlaga ke Zirids di Granada. Dia menghabiskan sebagian besar masa mudanya untuk bepergian melalui Afrika Utara dan Al-Andalus (Spanyol Muslim saat ini) dan tampaknya telah

mendapatkan informasi terperinci mengenai kedua wilayah tersebut. Dia mengunjungi Anatolia saat dia baru berusia 16 tahun. Dia belajar di Crdoba. Al-Idrisi memasukkan pengetahuan tentang Afrika, Samudra Hindia dan Timur Jauh yang dikumpulkan oleh pedagang dan penjelajah Islam dan dicatat di peta Islam dengan informasi yang dibawa oleh pelayaran Norman untuk membuat peta dunia yang paling akurat pada zaman pra-modern, yang berfungsi sebagai ilustrasi konkret Kitab nuzhat al-mushtaq-nya, yang dapat diterjemahkan sebagai Pengalihan Manusia untuk Berkelana ke Tempat yang Jauh. 4.3 adalah Tabula Rogeriana digambar oleh Al-Idrisi pada tahun 1154 untuk Raja Norman Roger II dari Sisilia. Setelah tinggal delapan belas tahun di istananya, di mana dia mengerjakan komentar dan ilustrasi peta. Peta tersebut, dengan legenda yang ditulis dalam bahasa Arab, sekaligus menunjukkan benua Eurasia secara keseluruhan, hanya menunjukkan bagian utara benua Afrika dan tidak memiliki rincian Tanduk Afrika dan Asia Tenggara. Foto 4.2 adalah monumen al iddrise yang terletak di Ceuta yang sekarang wilayah Spanyol.



Gambar 4.2 Monumen Muhammad al-Idrisi di tempat kelahirannya Ceuta Spanyol

4.1.2 Sejarah Peta dari pandangan Al-Qur'an 'al-idrisi'

Pandangan Islam mempercayai adanya dua kegaiban, yaitu kegaiban yang eksistensi nya tidak harus terukur atau tidak harus dipahami melalui metodologi sains, keghaiban yang difahami melalui sistem keimanan, selain itu juga kegaiban yang dikenal sebagai sunattullah. Manusia tak menyaksikan secara langsung tentang proses-proses pembentukan alam semesta, bahkan kehidupannya di planet Bumi ini hanya merupakan sebagian kecil episode dalam alam semesta. Metodologi sains, Islam memperkenalkan sebuah metodologi lain yaitu ilmu pengetahuan (tentang eksistensi kebenaran) bagi manusia bisa bersumber dari sumber terpercaya (kitab Allah) dan disampaikan dengan cara terpercaya. Al Quran merupakan sebuah kebenaran yang melengkapi pandangan yang dibangun melalui rasionalitas manusia semata, memperkenalkan Allah sebagai Tuhan alam semesta, ada alam ghaib, ada sesuatu yang ghaib dan manusia tak banyak mengetahuinya, ada cara atau upaya-upaya mencapai atau berkomunikasi dengan Yang MahaPencipta dan MahaPenguasa alam semesta. Pemahaman sunatullah dipergunakan untuk penyempurnaan dan membuat kemudahan-kemudahan dalam hidup dan beribadah agar bisa senantiasa lebih mendekat lagi kepada Allah SWT, Maha Pencipta dan Maha Penguasa alam semesta. Al Quran sebagai wahyu Allah yang telah diturunkan kepada umat manusia, perlu dipandang sebagai 'resources' bagi kehidupan umat manusia di planet Bumi yaitu sebagai petunjuk Allah swt agar hati manusia bisa tertutun dan terdidik sehingga berahlak mulia. Gambaran wahyu Allah dalam al Qur'an mengingatkan walaupun ada kepastian dari tangan-tangan ghaib berupa sunatullah yang sebagian telah bisa di ungkap melalui metodologi sains, namun masih terdapat pengetahuan dan kehendakNya yang ghaib dalam penciptaan Alam Semesta. Pemikiran atas fenomena alam semesta itu sangat diharapkan bisa membangun kesadaran beragama, mempertemukan kerja tangan-tangan ghaib Allah yang mengatur alam semesta dengan wahyuNya dalam al Quran. Keduanya adalah kebenaran yang berasal dari Allah zat yang Maha Esa. Mempertemukan



Gambar 4.3 Tabula Rogeriana Salah Satu peta yang paling canggih pada masa itu

kebenaran wahyu Allah dan ayat kauniyah merupakan proses pemahaman manusia tentang lingkungan kehidupannya yang lebih luas, menjangkau dunia dan akhirat, memadukan akal dan keyakinan dalam perspektif Islam. Abad sains dan teknologi telah dijalani manusia, makin tinggi pengetahuan manusia makin diperlukan kesadaran beragama yang lebih tinggi, perlu hidayah yang lebih banyak, agar mendapatkan tuntunanNya sehingga dijauhkan dari bencana sains dan teknologi. Berbagai bentuk upaya-upaya dakwah umat Islam perlu dikembangkan, upaya dakwah hendaknya juga menggerakkan kaum muda masyarakat kampus maupun non kampus dengan wawasan membangun kualitas lingkungan. Agar peran umat Islam dalam hal amar maruf nahi munkar dapat lebih dirasakan dalam masyarakat, perlu dirancang kegiatan yang bersinambung dalam pengembangan, pembelajaran maupun sosialisasi IPTEK. Di abad ke-11 M, seorang geografer termasyhur dari Spanyol, Abu Ubaid Al Bakri berhasil menulis kitab di bidang geografi, yakni Mujam Al Istajam (Ensiklopedi Geografi) dan Al Masalik wa Al Mamalik (Jalan dan Kerajaan). Buku pertama berisi nama-nama tempat di Jazirah Arab. Sedangkan yang kedua berisi pemetaan geografis dunia Arab zaman dahulu. Pada abad ke-12, geografer Muslim Al Idrisi berhasil membuat peta dunia. Al Idrisi yang lahir pada tahun 1100 di Ceuta Spanyol itu juga menulis kitab geografi berjudul Kitab Nazhah Al Muslak fi Ikhtira Al Falak (Tempat Orang yang Rindu Menembus Cakrawala). Kitab ini begitu berpengaruh sehingga diterjemahkan ke dalam bahasa Latin, Geographia Nubiensis. Seabad kemudian, dua geografer Muslim yakni Qutubuddin Asy Syirazi (1236 M-1311 M) dan Yaqut Ar Rumi (1179 M-1229 M) berhasil melakukan terobosan baru. Qutubuddin mampu membuat peta Laut Putih atau Laut Tengah yang dihadiahkan kepada Raja Persia. Sedangkan, Yaqut berhasil menulis enam jilid ensiklopedia bertajuk Mujam Al Buldan (Ensiklopedi Negeri-negeri). Penjelajah Muslim asal Maroko, Ibnu Battutah di abad ke-14 M memberi sumbangan dalam menemukan rute perjalanan baru. Hampir selama 30 tahun, Ibnu Battutah menjelajahi daratan dan mengarungi lautan untuk berkeliling dunia. Penjelajah Muslim lainnya yang mampu mengubah rute perjalanan laut adalah Laksamana Cheng Ho dari Tiongkok. Dia melakukan ekspedisi sebanyak tujuh kali mulai dari tahun 1405 hingga 1433 M. Sederet geografer Muslim telah banyak memberi kontribusi bagi pengembangan ilmu bumi. Al Kindi diakui begitu berjasa sebagai geografer pertama yang memperkenalkan percobaan ke dalam ilmu bumi. Sedangkan, Al Biruni didapat sebagai bapak geodesi yang banyak memberi kontribusi terhadap geografi dan juga geologi. John J OConnor dan Edmund F Robertson menuliskan pengakuannya terhadap kontribusi Al Biruni dalam MacTutor History of Mathematics. Menurut mereka, Al Biruni telah menyumbangkan kontribusi penting bagi pengembangan geografi dan geodesi. Dialah yang memperkenalkan teknik pengukuran bumi dan jaraknya dengan menggunakan triangulation. Al Birunilah yang menemukan radius bumi mencapai 6.339,6 km. Hingga abad ke-16 M, Barat belum mampu mengukur radius bumi seperti yang dilakukan Al Biruni. Bapak sejarah sains, George Sarton juga mengakui kontribusi sarjana Muslim dalam pengembangan geografi dan geologi. Kita menemukan dalam tulisannya metode penelitian kimia, sebuah teori tentang pembentukan besi. Salah satu kekhasan yang dikembangkan geografer Muslim adalah munculnya bio-geografi. Hal itu didorong oleh banyaknya orang Arab di era kekhilafahan yang tertarik untuk mendistribusi dan mengklasifikasi tanaman, binatang dan evolusi kehidupan. Para sarjana Muslim mencoba menganalisis beragam jenis tanaman.

BAB 5

PENDAHULUAN SEJARAH PETA DINDING

5.1 Biografi Willem Jansz Blaeu

Dalam biografi nya tentang willwm jansz Blaeu 1932, tapi tidak mungkin F. C. Wieder sudah merujuk pada ada nya lembaran tangan bawah dari peta ini di Museum Nasional Germanisches di Nurnberg. Sampai saat ini, investigasi di dasarkan pada lembar satu ini di Nurnberg dan deskripsi singkat dari Wieder yang menyebut peta ini sebagai monumen ukiran dan karya seni Belanda yang di ciptakan oleh seorang pria hebat dan berbudaya, sebuah monumen ilmiah yang luar biasa dari awal masa kolonial. Sementara penulis saat ini sedang belajar di Rijksmuseum Nederlands Scheepvaart-Museum di Amsterdam, dia membolak-balik dua kotak foto peta yang diperoleh dalam sebuah lelang yang di adakan pada tahun 1958. 'Betapa terkejut nya dia menemukan di antara foto-foto ini yang terpaku pada kardus gambar kecil (9 x 13cm) peta dinding dunia pada proyeksi Mercator oleh Willem Jansz Bleau dari tahun 1606-1607. Ini memberi kesan visual tentang keindahan asli bintang yang hilang ini. Gambar 1) dan memungkinkan deskripsi detail dari peta itu sendiri dan pengaruhnya terhadap peta dunia lainnya. Penulis buku ini mencoba membuat foto kecil yang di perbesar oleh Layanan Topografi di Delft sampai ukuran seperti itu sehingga garis pantai bisa menjadi lebih jelas dan huruf iegen terbaca. Pembesaran ini membuat kontur pantai lebih jernih namun teks tetap tidak jelas.

5.1.1 Karya individual Willem Jansz Bleau

Willem Jansz Bleau memiliki karya kartografi individual yang di terbitkan paling awal di Blaeu mencakup tiga peta dinding dunia, masing - masing di gambar dalam proyeksi yang berbeda. Dengan menyediakan berbagai macam bahan, yang telah disediakan untuk proyeksi tersebut. willem jansz bleau mencoba memenuhi semua permintaan pelanggan nya sambil bersaing dengan penerbit peta lain nya di Amsterdam.

pada paten dua belas tahun yang di berikan kepada cornelis claeszon untuk dapat mempublikasi peta dunia 1592 oleh plancius, peta datar silinder 20 lembar, yang telah kadaluarsa pada tahun 1604. Peta oleh plancius ini merupakan tengara penting dalam pengembangan kartografi dan membangkitkan keagungan terhadap orang - orang di zaman pembuatan peta, dimana peta dibuat untuk alasan komersial, willem jansz bleau Buru - buru memproyeksikan agar peta dunia ini di ukir kembali oleh josua van den dan di ubah dengan penemuan terbaru.

5.2 Peta dinding diterbitkan

Pada periode yang sama, willem jansz bleau. Dianggap menerbitkan peta dinding dunia dengan proyeksi stereografik. peta dinding di dua belahan otak ini di terbitkan pada tahun 1605 oleh willem jansz bleau. Dan pada akhirnya, untuk memenuhi semua kebutuhan pelanggannya, willem jansz bleau juga memutuskan untuk menerbitkan peta dunia mengenai proyeksi mercator. peta dinding ini di proyeksikan akan berpengaruh pada peta dunia lainnya yang akan dibahas di bawah ini. Tidak ada salinan lengkap peta ini yang bertahan. sementara itu j.keuning sedang melakukan katalogisasi f.c. Wieder's tentang warisan museum maritiem prins Hendrik di Rotterdam, ia menemukan di antara dokumen - dokumen deskripsi dari peta dunia ini pada proyeksi mercator. Deskripsi ini kemudian di tulis oleh f. c. wieder dan di masukkan secara lengkap dalam j. Keuning biografi bleau.

Wieder Menjual Petanya dan Berhasil ditemukanya peta dinding oleh WillemJansz Bleau. Namun, artikel tersebut tidak menyebutkan lokasi peta dinding berada. Selama perang dunia kedua wieder menjual koleksi peta itu sendiri di berlin dimana ia hilang



Gambar 5.1 Willem Jansz Blaeu



Gambar 5.2 Peta Dinding

di dalam kekacauan. Sehingga sangat mungkin koleksi ini termasuk peta dinding dunia. tentang proyeksi mercator 1606 - 1607, karena dalam salinan pribadi kartografi monumenla ini (sekarang di pelihara di perpustakaan negara bagian barat berlin) dr. Wieder di tandai dalam manuskrip - dengan catatan : ‘sekarang peta ini selesai’.

Di dalam biografi nya tentang willem jansz Blaeu 1932, tapi tidak mungkin F. C. Wieder sudah merujuk pada adanya lembaran tangan dari peta ini di Museum Nasional Germanisches di Nurnberg. Sampai saat ini, investigasi di dasarkan pada lembar yang satu ini di Nurnberg dan deskripsi singkat dari Wieder yang menyebut peta ini sebagai monumen ukiran dan karya seni Belanda yang di ciptakan oleh seorang pria hebat dan berbudaya, sebuah monumen ilmiah yang luar biasa dari awal masa colonial. Sementara penulis saat ini sedang belajar di Rijksmuseum Nederlands Scheepvaart-Museum di Amsterdam, dia membolak-balik dua kotak foto yang di peroleh dalam sebuah lelang yang di adakan pada tahun 1958. Betapa terkejut nya dia menemukan di antara foto-foto ini yang terpaku pada kardus gambar kecil (9 x 13 cm) peta dinding dunia pada proyeksi Mercator oleh Willem Jansz Bleau dari tahun 1606 - 1607. Ini memberi kesan visual tentang keindahan asli bintang yang hilang ini. Gambar 1) dan memungkinkan deskripsi detail dari peta itu sendiri dan pengaruh nya terhadap peta dunia lain nya. Penulis buku ini mencoba membuat foto-foto kecil yang di perbesar oleh Layanan Topografi di Delft sampai ukuran seperti itu sehingga garis pantai bisa menjadi lebih jelas dan huruf iegen terbaca. Pembesaran ini membuat kontur pantai lebih jernih namun teks tetap tidak jelas.

Untung nya, dengan mencari jalan ke berbagai peta dunia yang berasal dari Willem Jansz Bleau. 'peta dinding dia berhasil di patenkan [14].

5.2.1 Karya Willem Blaeu

Selama hampir 70 tahun, biografi Baudet Willem Jansz Bleau Maior sudah menjadi sumber utama untuk pengetahuan kita tentang kehidupan dalam karya willem jansz Blaeu's. Karena, bagaimana pun kurang nya sumbersumber yang tersedia di tahun-tahun sebelum nya sudah di tangani sangat ringkas dalam karya tersebut. Ini khusus nya adalah kebenaran hubungan Brahe's dengan Tycho Brahe dan pelatihan yang diterima oleh Joll Ander muda di bawah bimbingan Brahe's pafa Hven. Pelatihan ini diketahui, dalam banyak cara untuk menentukan penting nya pekerjaan nanti nya dia di tanah sendiri. Pada tahun 1914 kehidupan dan pekerjaan willem jansz Baleu's akan diteliti dari awal oleh Stevenson, kali ini dengan referensi khusus untuk karya nya, peta penting dunia 1605. Pekerjaan tidak berisi informasi lain yang bener-bener baru tentang hubungan willem jansz Blaeu's dengan Brabe. Sejak hari Baudet's bagaimanapun, pengetahuan kita tentang kehidupan dan penelitian Brahe's sudah sangat meningkat. Dan telah memiliki sedikit bagian dalam hal ini. Pada tahun 1980 dreyer di terbitkan sejauh biografi paling rinci dan terbaik dari Brabe. Dreyer, salah satu sejarawan pembimbing astronomi di waktu kita dan dane, pada khusus nya di lengkapi untuk kumpulan karya Brahe's dan pada 1913 ia terbit di edisi jilid pertama dari Opera Omnia. Dua tahun setelah kematian nya, pada tahun 1928 pekerjaan selesai. Artinya dari koleksi surat ini dapat di tetapkan beberapa fakta lebih lanjut untuk tetap di Blaeu Hven dan pandangan sebelum nya di koreksi. Untuk jenis pengetahuan pelatihan dari Blaeu di berikan pada Hven, bagaimanapun penelitian masih harus mencari sumber utama dalam karya ilmiah umum dan praktis yang di lakukan disana [15].

Hampir tidak memiliki brahe menetap dirinya di pulau kecil hven (kemudian yang dimiliki Denmark) Suara, yang ia telah diberikan pada 1576 oleh Fredrik II, raja Denmark dari laporan view baru astronom telah di rayakan dua ini telah menyebar ke luar negeri. Atau apakah itu lama sebelum orang - orang muda yang tertarik dari dekat dan jauh membuat mereka berada di sana sebagai murid-muridnya. Beberapa menjadi para pembantu nya. Dalam dua puluh tahun brahe tinggal di hven ada sebagai aturan enam sampai delapan. Kadang-kadang bahkan sebanyak sepuluh sampai dengan dua puluh, pria muda yang bekerja terus . Brahe membuat permintaan besar pada muridmurid nya. Beberapa yang ia di undang untuk datang; server lain ia pasti hanya mengambil rekomendasi khusus. Semua dianggap sebagai sebuah kehormatan yang besar untuk di izinkan untuk datang ke hven. Untuk pertanyaan apakah brahe dan Maior, mengingat pembentuk kematian awal sebagai 1601, bisa membentuk setiap persahabatan bertahun-tahun berdiri seperti yang sering di katakan di biografi data mengenai willwm jansz bleu. dan itu ada, tidak di ragukan lagi bahwa blaen, karena kemampuan ini, sekaligus memenangkan brahe's hal khusus. Ini mau menjadi bahwa brahe punya pengalaman baik orang Belanda muda dengan siapa ia datang ke dalam kontak pada hven, karena Maior tidak satu - satunya. Groningenrudolphus tertentu ia disebutkan dalam ini 1586 dan 1588 Meteorologi hari buku yang telah di pelihara dari tahun 1582 untuk 1597 pada hven. Pada 1590 tinggal pendek dibuat tidak oleh Arnold Van langren, putra terkenal dunia penanda florentius Jacobus Van langren. Jacobus buruk mengirimnya ke brahe untuk meminta untuk akan di izinkan supaya bisa menyalin Katalog bintang brahe's, yang Jacobus berkeinginan untuk membuat penggunaan untuk bola langit.

Ini mau menjadi bahwa brahe punya pengalaman baik orang Belanda muda dengan siapa ia datang ke dalam kontak pada hven, karena Maior tidak satu - satunya. Groningenrudolphus tertentu ia disebutkan dalam ini 1586 dan 1588 Meteorologi hari buku yang telah di pelihara dari tahun 1582 untuk 1597 pada hven. Pada 1590 tinggal pendek dibuat tidak oleh Arnold Van langren, putra terkenal dunia penanda florentius Jacobus Van langren. Jacobus buruk mengirimnya ke brahe untuk meminta untuk akan di izinkan supaya bisa menyalin Katalog bintang brahe's, yang Jacobus berkeinginan untuk membuat penggunaan untuk bola langit [15].

PENGANTAR SEJARAH BUMI

6.1 Sejarah Bumi

Bumi merupakan planet atau rumah kita dalam kedudukan di tata surya. Peradaban kuno percaya bahwa bumi itu datar, dengan langit berputar-putar sekali sehari. Secara umum yang diyakini bahwa kehidupan di Bumi dimulai di Bumi itu sendiri, beberapa waktu setelah terbentuknya planet antara 4000-5000 juta tahun yang lalu. Namun ada yang berpendapat bahwa kehidupan di luar bumi itu ada, tetapi kita tidak memiliki bukti pasti tentang kehidupan di tempat lain. Yang perlu kita ketahui bumi berada pada galaksi bimasakti dimana terdapat matahari sebagai sistem bintang.

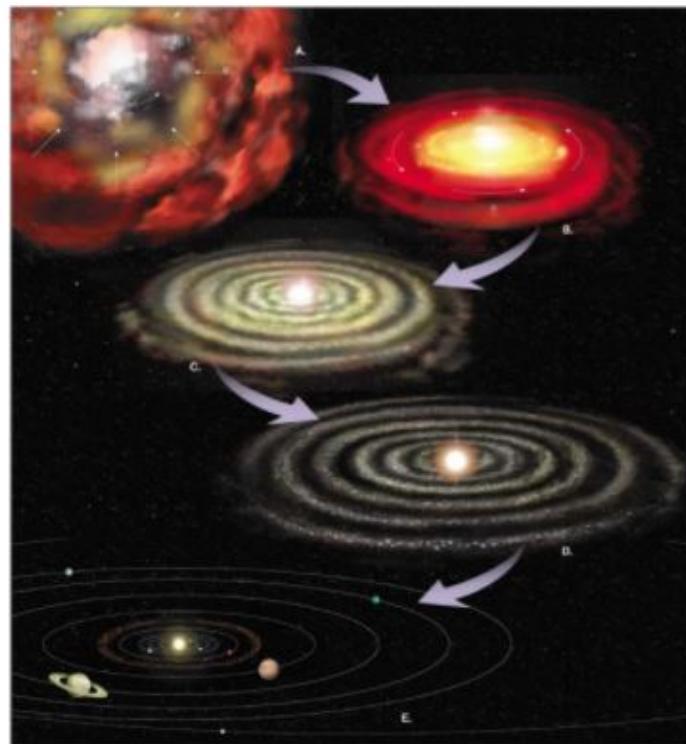
Dalam Geologi sendiri atau biasa disebut sebagai ilmu pengetahuan tentang Kebumian yang mempelajari segala sesuatu mengenai planet Bumi beserta isinya yang pernah ada. Dalam Geologi juga akan dibahas tentang sifat-sifat dan bahan-bahan yang membentuk bumi itu apa, serta struktur dan proses-proses yang bekerja baik di dalam maupun dibagian teratas permukaan bumi, kedudukannya di Alam Semesta hingga sekarang. Geologi merupakan ilmu pengetahuan yang kompleks, mempunyai pembahasan materi yang beraneka ragam namun juga merupakan ilmu pengetahuan yang bagus untuk dipelajari. Sebagai landasan prinsip untuk dapat mempelajari ilmu geologi adalah bahwasanya kita harus menganggap bumi ini sebagai suatu benda yang secara dinamis berubah sepanjang masa, setiap saat dan setiap detik. Pemikiran geologi modern dikenal kan oleh Huttonian revolution mengemukakan pemikiran-pemikirannya sebagai berikut:

1. Bahwasanya proses-proses alam yang sekarang terjadi, menyebabkan perubahan pada permukaan bumi, juga bekerja sepanjang umur dari bumi ini.
2. Ia juga mengamati bahwa proses-proses tersebut yang walaupun bekerja sangat lambat, tetapi pada akhirnya mampu menyebabkan terjadi nya perubahan-perubahan yang sangat besar pada bumi.
3. Bahwa bumi ini sangat dinamis, yang berarti mengalami perubahan-perubahan secara terus-menerus mengikuti suatu pola daur (siklus) yang berulang-ulang.

Bumi sendiri berada di kawasan di mana terjadi nya tumpang tindih antara litosfer (daratan) bagian padat dari bumi, hidrosfer (perairan), dan atmosfer (udara) yang menyelubungi bumi dengan zarah-zarah dan benda-benda yang mengisinya. Dalam sejarah terbentuknya bumi sewaktu SMA kita pernah mempelajari teori-teori terbentuknya bumi dalam pelajaran geografi [16].

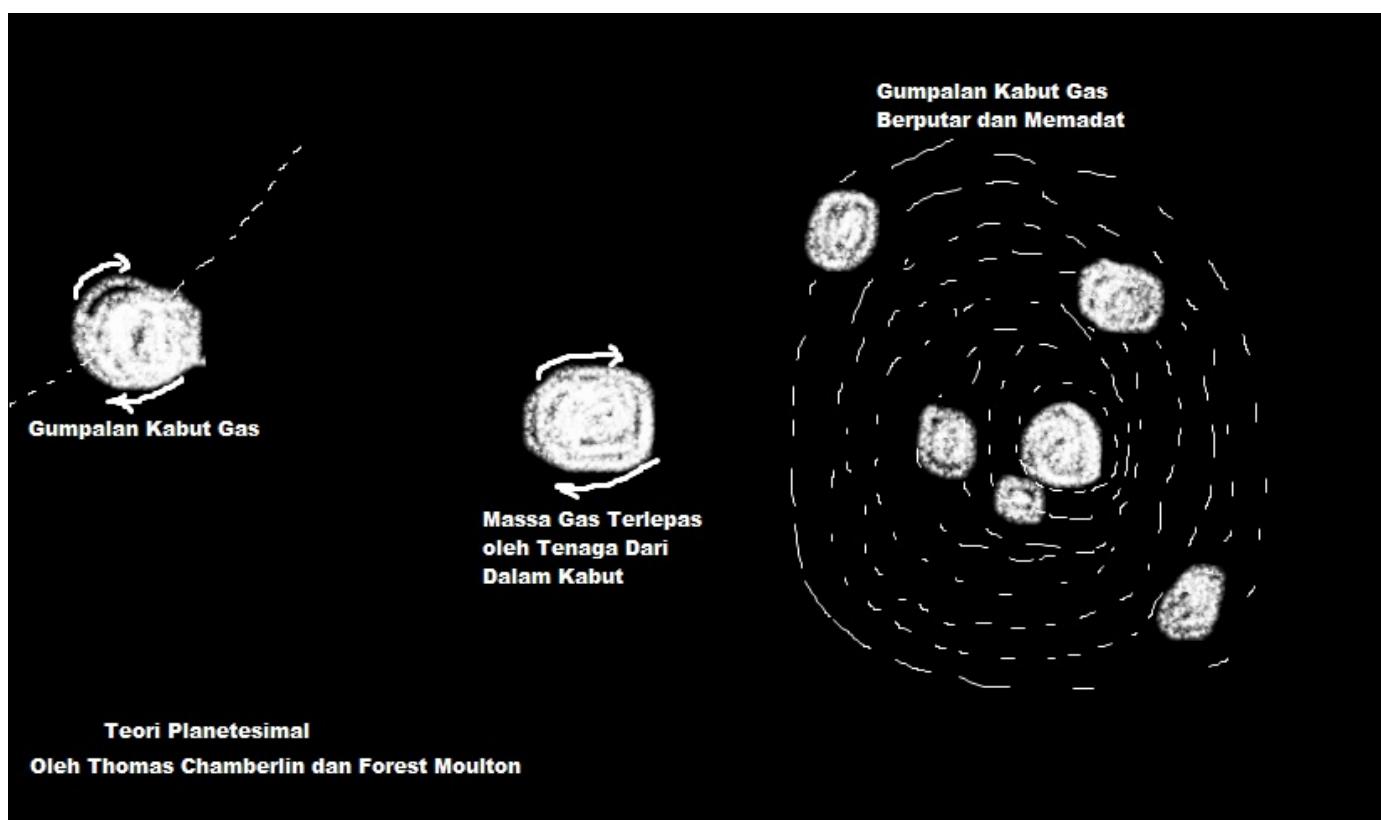
6.1.1 Teori-teori terbentuknya Bumi

6.1.1.1 Teori Kabut Kant-Laplace Gambar 6.1 adalah gambar dari Teori Kabut Nebula. Teori ini dikenal dengan teori kabut (nebula) yang dikemukakan oleh Immanuel Kant (1755) dan Pierre de Laplace (1796). dalam teori ini dikemukakan bahwa di jagat raya terdapat gas yang kemudian berkumpul menjadi kabut(nebula). gaya tarik-menarik antar gas ini membentuk kumpulan kabut yang sangat besar dan berpusat semakin cepat sehingga materi kabut bagian khatulistiwa ter lempar memisah dan memadat(karena pendinginan), bagian yang ter lempar inilah yang kemudian menjadi sebuah planet dalam tata surya. Bumi baru terus bertumbuh sampai suhu interior nya cukup panas untuk melelehkan logam siderofil. Dengan massa jenis yang lebih tinggi dari silikat, membuat logam ini menjadi tenggelam. Proses ini terjadi 10juta tahun setelah Bumi mulai terbentuk, dan menghasilkan struktur Bumi yang berlapis-lapis dan mengakibatkan terbentuknya medan magnet. J. A. Jacobs merupakan orang pertama yang menunjukkan bahwa inti dalam bagian dalam yang padat berbeda dari inti luar yang padatmembeku dan mengembang keluar inti



Gambar 6.1 Gambar Teori Nebula

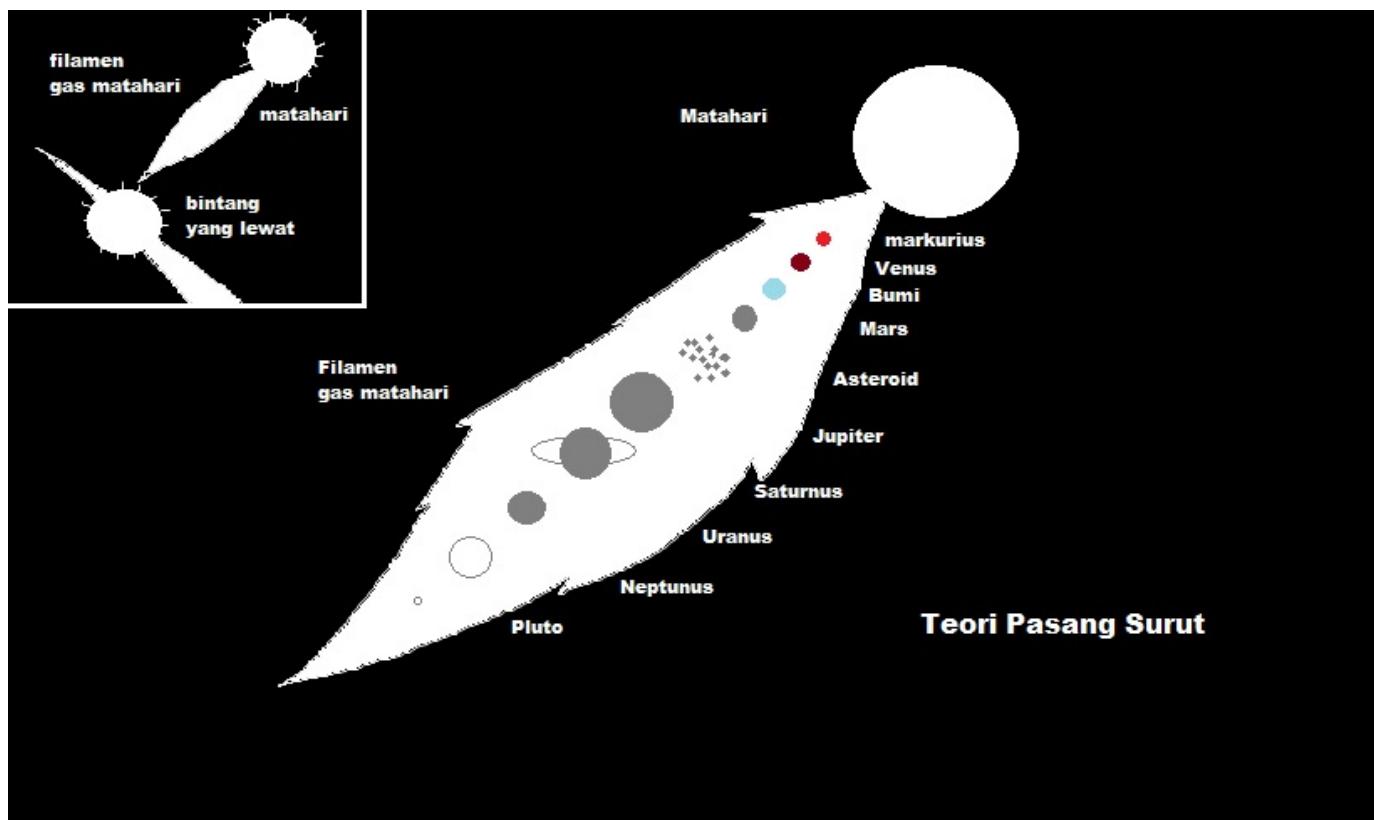
luar yang cair dikarbagus untuk bagian dalam bumi yang makin mendingin (sekitar 100 C per miliar tahun). Ekstrapolasi dari pengamatan ini memperkirakan bahwa inti terbentuk pada masa 24 miliar tahun yang lalu. Jika ini benar maka berarti bahwa inti bumi bukanlah fitur primordial yang berasal selama pembentukan planet.



Gambar 6.2 Gambar Teori Planetesimal

6.1.1.2 Teori Planetesimal Pada Gambar berikut 6.2 adalah gambar dari Teori Planetesimal. seabad kemudian sesudah teori kabut tersebut muncul teori Planetesimal yang dikemukakan oleh Chamberlin dan Moulton. Teori ini mengungkapkan bahwa pada mulanya telah terdapat Matahari asal. Pada suatu ketika, matahari asal ini di dekati sebuah bintang besar yang menyebabkan terjadi nya penarikan pada bagian matahari. Akibat tenaga tarik menarik tadi, terjadilah ledakan yang dahsyat. Gas yang meledak ini keluar dari atmosfer matahari, kemudian mengembun dan membeku sebagai benda-benda yang padat(disebut planetesimal). Planetesimal ini dalam perkembangannya menjadi planet-planet, dan salah satunya planet bumi kita.

Pada Gambar 6.2 adalah gambar dari Teori Planetesimal. Seabad kemudian sesudah teori kabut tersebut muncul teori Planetesimal yang dikemukakan oleh Chamberlin dan Moulton. Teori ini mengungkapkan bahwa pada mulanya telah terdapat Matahari asal, pada suatu ketika, matahari asal ini didekati sebuah bintang besar yang menyebabkan terjadinya penarikan pada bagian matahari. Akibat tenaga tarik menarik tadi, terjadilah ledakan yang dahsyat. Gas yang meledak ini keluar dari atmosfer matahari, kemudian mengembun dan membeku sebagai benda-benda yang padat (disebut planetesimal). Planetesimal ini dalam perkembangannya menjadi planet-planet, dan salah satunya planet bumi kita.



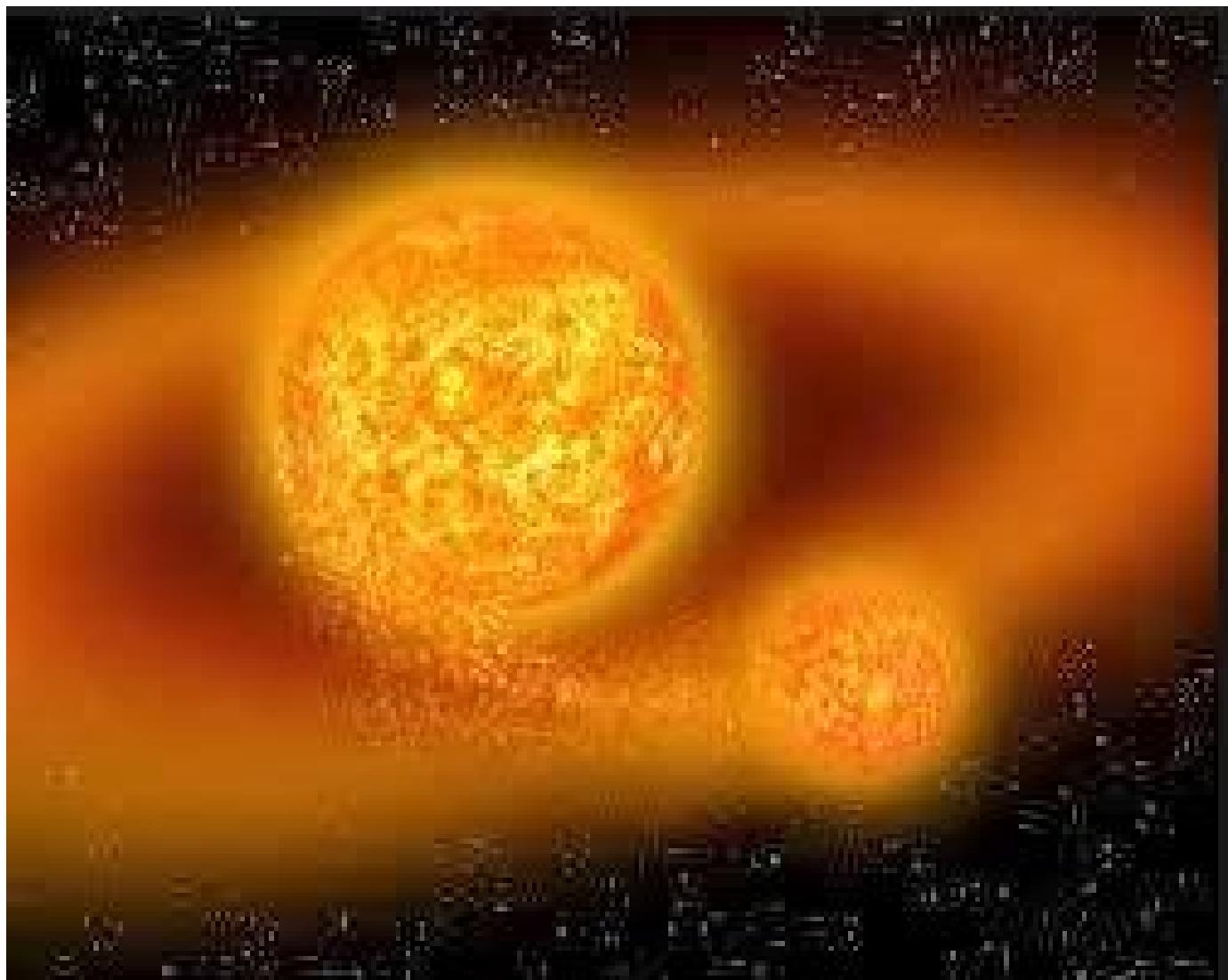
Gambar 6.3 Gambar Pasang Surut

6.1.1.3 Teori Pasang Surut Gas Pada Gambar berikut 6.3 adalah gambar dari Teori Pasang Surut. Teori ini dikemukakan oleh Jeans dan Jeffreys, yakni bahwa sebuah bintang besar mendekati matahari dalam jarak pendek, sehingga menyebabkan terjadi pasang surut pada tubuh matahari. Dalam lidah yang panas ini terjadi perapatan gas-gas dan akhirnya kolom-kolom ini akan pecah, lalu berpisah menjadi benda-benda tersendiri yaitu planet-planet. Bintang besar yang menyebabkan penarikan pada bagian-bagian tubuh matahari tadi melanjutkan perjalanan di jagat raya, sehingga lambat laun akan hilang pengaruhnya terhadap planet-planet yang terbentuk tadi, lalu planet-planet itu akan mengelilingi matahari dan mengalami proses pendinginan, proses pendinginan berjalan lambat pada planet besar seperti Jupiter dan Saturnus, sedangkan planet kecil seperti bumi mengalami proses pendinginan yang relatif lebih cepat.

Pada Gambar berikut 6.3 adalah gambar dari Teori Pasang Surut. Teori ini dikemukakan oleh Jeans dan Jeffreys, yakni bahwa sebuah bintang besar mendekati matahari dalam jarak pendek, sehingga menyebabkan terjadi pasang surut pada tubuh matahari. Dalam lidah yang panas ini terjadi perapatan gas-gas dan akhirnya kolom-kolom ini akan pecah, lalu berpisah menjadi benda-benda tersendiri yaitu planet-planet. Bintang besar yang menyebabkan penarikan pada bagian-bagian tubuh matahari tadi melanjutkan perjalanan di jagat raya, sehingga lambat laun akan hilang pengaruhnya terhadap planet-planet yang terbentuk tadi, lalu planet-planet itu akan mengelilingi matahari dan mengalami proses pendinginan, proses pendinginan berjalan lambat pada planet besar seperti Jupiter dan Saturnus, sedangkan planet kecil seperti bumi mengalami proses pendinginan yang relatif lebih cepat.

6.1.1.4 Teori Bintang Kembar Pada Gambar 6.4 adalah gambar dari Teori Bintang Kembar. Teori ini dikemukakan oleh seorang ahli astronomi R. A. Lyttleton. Menurut teori ini, galaksi berasal dari kombinasi bintang kembar. Salah satu bintang meledak sehingga banyak materi yang ter lempar. Karena bintang yang tidak meledak mempunyai gaya gravitasi yang masih kuat, maka sebaran pecahan bintang tersebut mengelilingi bintang yang tidak meledak. Bintang yang tidak meledak itu adalah matahari, sedangkan pecahan bintang yang lain itu adalah planet-planet yang mengelilinginya.

6.1.1.5 Teori Dentuman Besar (Big Bang Teori) Pada Gambar 6.5 adalah gambar dari Teori Dentuman Besar(Big Bang). Pada Teori ini berdasarkan dari asumsi adanya massa yang sangat besar dan mempunyai massa jenis sangat besar. Adanya reaksi inti menyebabkan massa tersebut meledak hebat. Massa tersebut kemudian mengembang dengan sifat sangat cepat menjauhi pusat ledakan, karena adanya gravitasi, maka bintang yang paling kuat gravitasi nya akan menjadi pusatnya. Dari berbagai teori, teori ini yang paling banyak didukung oleh para ilmuwan.



Gambar 6.4 Gambar Teori Bintang Kembar

6.2 Pendapat Tentang Sejarah Bumi

Bumi terbentuk sekitar 4,54 miliar (4,54109) tahun yang lalu melalui akresi dari nebulosa matahari. Pelepasan gas vulkanik diduga menciptakan atmosfer tua yang nyaris tidak beroksigen dan beracun bagi manusia dan sebagian besar makhluk hidup masa kini. Sebagian besar permukaan Bumi meleleh karena vulkanisme ekstrem dan sering bertabrakan dengan benda angkasa lain. Sebuah tabrakan besar diduga menyebabkan kemiringan sumbu Bumi dan menghasilkan Bulan. Seiring waktu, Bumi mendingin dan membentuk kerak padat dan memungkinkan cairan tercipta di permukaannya. Bentuk kehidupan pertama muncul antara 2,8 dan 2,5 miliar tahun yang lalu. Kehidupan fotosintesis muncul sekitar 2 miliar tahun yang lalu, nam memperkaya oksigen di atmosfer. Sebagian besar makhluk hidup masih berukuran kecil dan mikroskopis, sampai akhirnya makhluk hidup multiseluler kompleks mulai lahir sekitar 580 juta tahun yang lalu. Pada periode Kambrium, Bumi mengalami diversifikasi filum besar-besaran yang sangat cepat. Perubahan biologis dan geologis terus terjadi di planet ini sejak terbentuk. Organisme terus berevolusi, berubah menjadi bentuk baru atau punah seiring perubahan Bumi. Proses tektonik lempeng memainkan peran penting dalam pembentukan lautan dan benua di Bumi, termasuk kehidupan di dalamnya. Biosfer memiliki dampak besar terhadap atmosfer dan kondisi abiotik lainnya di planet ini, seperti pembentukan lapisan ozon, proliferasi oksigen, dan penciptaan tanah. Dalam sebuah artikel dari zuhdi2012sistem yang menyebutkan bahwa Bumi merupakan salah satu planet yang berada dalam tata surya yang diduga terbentuk dari pecahan-pecahan bintang pada jutaan tahun yang lalu, dan kemudian terperangkap pada gravitasi matahari sehingga akan selalu mengelilingi matahari. Menurut Hukum Newton kenapa planet dapat bertahan dalam pergerakan keliling atau biasa disebut revolusi dikarbagus untukkan planet melakukan gerak melingkar yang menimbulkan gaya sentrifugal yang besarnya dengan gaya gravitasi namun berlawanan arah. Gaya gravitasi ini sendiri akan berkurang sesuai dengan semakin jauhnya jarak planet dari matahari, sedangkan gaya sentrifugal akan tergantung pada kecepatan gerak melingkar planet. Semakin cepat gerakan tersebut maka akan semakin besar daya sentrifugal. Bila secara kebetulan kedua gaya ini memiliki kecepatan yang sama besar, maka planet akan terjebak mengelilingi matahari. Pada saat tata surya terbentuk diperkirakan terdapat jutaan planet. Akan tetapi sebagian terjatuh ke matahari atau terlempar lepas dari pengaruh matahari. Selain berkeliling, planet juga akan bergerak memutari porosnya (rotasi). Gerak rotasi ini sendiri berlangsung dalam waktu lama sehingga membuat planet berbentuk seperti bola. Pada masa lalu, planet planet bukanlah sebuah benda padat, melainkan berupa magma atau berupa cairan batu. Bagian padat pada planet terbentuk selama proses pendinginan dan terjadi pada bagian kulit terluar dari planet tersebut. Bentuk Bumi sendiri yang



Gambar 6.5 teori big bang

dikatakan berbentuk bola tidaklah sempurna. Gerak Rotasi telah mengubah bentuk bumi menjadi agak cepat terhadap kedua kutub nya.[17]

Sejarah pembentukan Bumi yang dipelajari dalam materi pelajaran Geografi cenderung memiliki sifat abstrak yang akan lebih mudah dimengerti, jika memakai media yang cocok. Salah satu Inovasi pembelajaran yang tepat untuk dilakukan adalah menggunakan kartu indeks dan media film. Media seperti kartu indeks yang dipergunakan sebagai salah satu upaya yang memudahkan peserta didik agar mengingat konsep-konsep materi yang sedang dipelajari sedangkan media film sendiri merupakan media visual yang akan menjelaskan dengan lebih konkret tentang fenomena bumi. Dalam sebuah artikel dari @articlewidiyati2011 meningkatkan menyebutkan bahwa Pembentukan Bumi dengan kategori Continental Drift Teori atau biasa disebut dengan teori pengepungan benua yang di temukan oleh Alfred Wegener pada tahun 1912 mengemukakan bahwa sampai sekitar 255 juta tahun lalu, di bumi baru ada satu benua dan samudra yang sangat luas. Benua raksasa ini sendiri dinamakan pangea, sedangkan kawasan samudra yang mengapit nya itu mengalami retakan-retakan dan pecah. Sekitar 135 juta tahun lalu, benua raksasa tersebut pecah menjadi dua, yaitu pecahan benua di sebelah utara yang dinamakan Lauransia dan di bagian selatan dinamakan gondwana. [18]

PENDAHULUAN SEJARAH BENUA

7.1 Sejarah Benua

7.1.1 Benua pertama

Mantel konveksi, proses yang mendorong lempeng tektonik adalah hasil dari aliran panas dari dalam bumi ke permukaan bumi [20]. Termasuk juga penciptaan lempeng tektonik di pegunungan bawah laut. Lempeng ini dihancurkan oleh subduksi di zona subduksi. Pada awal eon Arkean 7.1 (sekitar 3 miliar tahun yang lalu) mantel itu jauh lebih panas mungkin sekitar 1600 C, sehingga proses konveksi terjadi lebih cepat.

Kerak bumi mulai terbentuk saat permukaan bumi mulai memadat, menghilangkan bekas-bekas pergeseran lempeng tektonik Hadean. Namun, diperkirakan kerak bumi memiliki komposisi Basalt seperti Kerak Samudera. Potongan kerak benua besar yang pertama, muncul saat akhir masa Hadean, sekitar 4 miliar tahun yang lalu. Kraton adalah bagian kecil yang tersisa dari benua pertama. Potongan-potongan yang terjadi pada akhir Hadean sampai awal Arkean membentuk inti lempengan yang tumbuh menjadi benua seperti sekarang.

Batuan tertua ditemukan di Laurentia, Kanada, yang berupa tonalit yang berumur sekitar 4 miliar tahun. Bebatuan ini menunjukkan jejak metamorfosis oleh suhu tinggi, dan biji-bijian sedimen yang terkena erosi selama terbawa oleh air, yang menunjukkan terdapat sungai dan laut pada 4 miliar tahun yang lalu.

7.1.2 Benua raksasa pada masa Proterozoikum

Rekonstruksi pergerakan lempeng tektonik pada 250 juta tahun terakhir (pada era Kenozoikum dan mesozoikum) dapat dilakukan dengan melihat kecocokan benua, anomali magnetik dasar laut, dan kutub paleomagnetik [20]. Para ahli tidak menemukan kerak samudera yang terbentuk sebelum waktu tersebut, sehingga rekonstruksi sebelum waktu tersebut sulit untuk dilakukan. Kutub paleomagnetik dilengkapi dengan bukti geologi seperti sabuk orogenik, yang menandai tepi lempeng kuno, dan distribusi flora dan fauna pada masa itu.

Sepanjang sejarah bumi, ada saat dimana benua bertabrakan dan membentuk benua raksasa, yang kemudian pecah menjadi benua baru. Sekitar 1000830 juta tahun yang lalu, benua yang paling luas bersatu membentuk sebuah benua raksasa Rodinia. Sebelum Rodinia terbentuk, diperkirakan telah terbentuk terlebih dahulu Columbia atau Nuna pada awal sampai pertengahan masa Proterozoikum.

Setelah Rodinia pecah sekitar 800 juta tahun lalu, benua-benua tersebut kemungkinan telah membentuk benua raksasa lain yang berumur pendek yaitu, Pannotia 7.2 pada 550 juta tahun lalu. Hipotesis benua raksasa mengacu pada Pannotia atau Vendia. Bukti yang memperkuat hipotesis tersebut adalah fase tabrakan benua yang diketahui sebagai orogeni Pan-Afrika, yang bergabung dengan benua Afrika, Amerika Selatan, Antartika dan Australia. Keberadaan Pannotia ditentukan oleh terjadinya retakan antara Gondwana (sebagian besar termasuk daratan di belahan bumi selatan, serta meliputi Semenanjung Arab dan anak benua India) dan Laurentia (kira-kira setara dengan Amerika Utara pada masa sekarang). Hal ini meyakinkan bahwa pada akhir masa eon Proterozoikum, sebagian besar benua bergabung dalam posisi di sekitar kutub selatan.



Gambar 7.1 Peta geologi Amerika Utara, kode warna menunjukkan usia. Warna merah dan pink menunjukkan batuan dari masa eon Arkean.

7.1.3 Bukti Tersusunnya Benua Kuno

Terdapat bukti dari para ahli yang digunakan untuk memperkirakan tersusunnya benua kuno. Menurut Alfred Wegener(1880-1930), bahwa semua benua pernah bersatu kemudian berpecah menjadi sekarang ini dan benua yang bersatu itu dinamakan Pangaea (Benua Besar)[21]. Kemudian para ahli meneliti tentang benua dan membuat spekulasi-spekulasi teoritis yaitu melihat pada peta bahwa benua saling melengkapi dilihat dari garis pantai yang saling melengkapi seperti bagian puzzle. Kemudian meneliti fosil, bukti lain dari kehidupan lampau yaitu Mesosaurus. Mesosaurus adalah reptilia purba yang hanya hidup di air tawar dan ternyata hanya ada dua kawasan di dunia yang memiliki fosil Mesosaurus ini yaitu Pantai Timur Amerika Selatan dan Pantai Barat Afrika. Kesimpulannya, fosil yang sama telah ditemukan di dalam batuan di kedua sisi lautan. Kemudian bukti korelasi batuan dan pegunungan telah ditemui di kedua belah sisi lautan. Yaitu meneliti banjaran pegunungan di Timur Laut Amerika Serikat dan banjaran pegunungan di Utara Eropa. Keduanya sangat sepadan atau keduanya tersusun daripada jenis batuan yang sama. Kemudian bukti data iklim masa lalu, terbukti pada Glacial Striations atau terdapat bentuk goresan pada batuan dan ini dapat dilihat dari hutan hujan tropika Amerika Selatan dan Afrika saat ini terdapat goresan glasier. Kesimpulannya adalah arang batu telah ditemukan di kawasan sejuk dan bukti glasier telah ditemui di kawasan panas berarti sebelumnya ada kemungkinan benua bersatu.

7.2 Sejarah Koordinat

Menurut ahli sejarah yang bernama Heroditus (450 M) menyatakan bahwa geometri berasal dari Mesir. Rane Descartes seorang matematikawan, yang lahir di sebuah Desa La Haye Prancis pada tahun 1596, adalah orang yang memiliki ketertarikan di bidang geometri. Rane Descrates telah menemukan sebuah metode untuk menyajikan sebuah titik sebagai bilangan berpasangan pada sebuah bidang datar. Bilangan-bilangan tersebut terletak pada dua garis yang saling tegak lurus antara satu dengan lainnya dan



Gambar 7.2 Rekonstruksi benua raksasa Pannotia (warna kuning) pada 550 juta tahun lalu.

berpotongan di sebuah titik (0,0) dinamakan Origin , dan biasanya di simbol kan dengan huruf kapital O (0,0). Bidang tersebut dinamakan bidang KOORDINAT atau yang lebih dikenal dengan bidang KARTESIUS.

7.3 Sistem Koordinat

Sistem koordinat dimaksudkan untuk memberikan peng-alamat-an terhadap setiap lokasi di permukaan bumi. Peng-alamat-an dengan sistem kordinat didasarkan atas jarak timur-barat dan utara-selatan suatu tempat dari suatu titik pangkal tertentu. Jarak diukur dalam satuan derajat sudut yang dibentuk dari titik pangkal ke posisi tersebut melalui pusat bumi. Sedangkan titik pangkal ditetapkan berada di perpotongan belahan utara-selatan bumi (garis katulistiwa) dengan garis yang membelah bumi timur-barat[17]. Koordinat diambil untuk menjadi bilangan riil dalam matematika dasar, tetapi mungkin bilangan kompleks atau elemen-elemen dari sistem yang lebih abstrak. Penggunaan sistem koordinat memungkinkan masalah dalam angka untuk diterjemahkan ke dalam masalah-masalah tentang geometri dan juga sebaliknya.

7.3.1 Sistem Koordinat Dua Dimensi

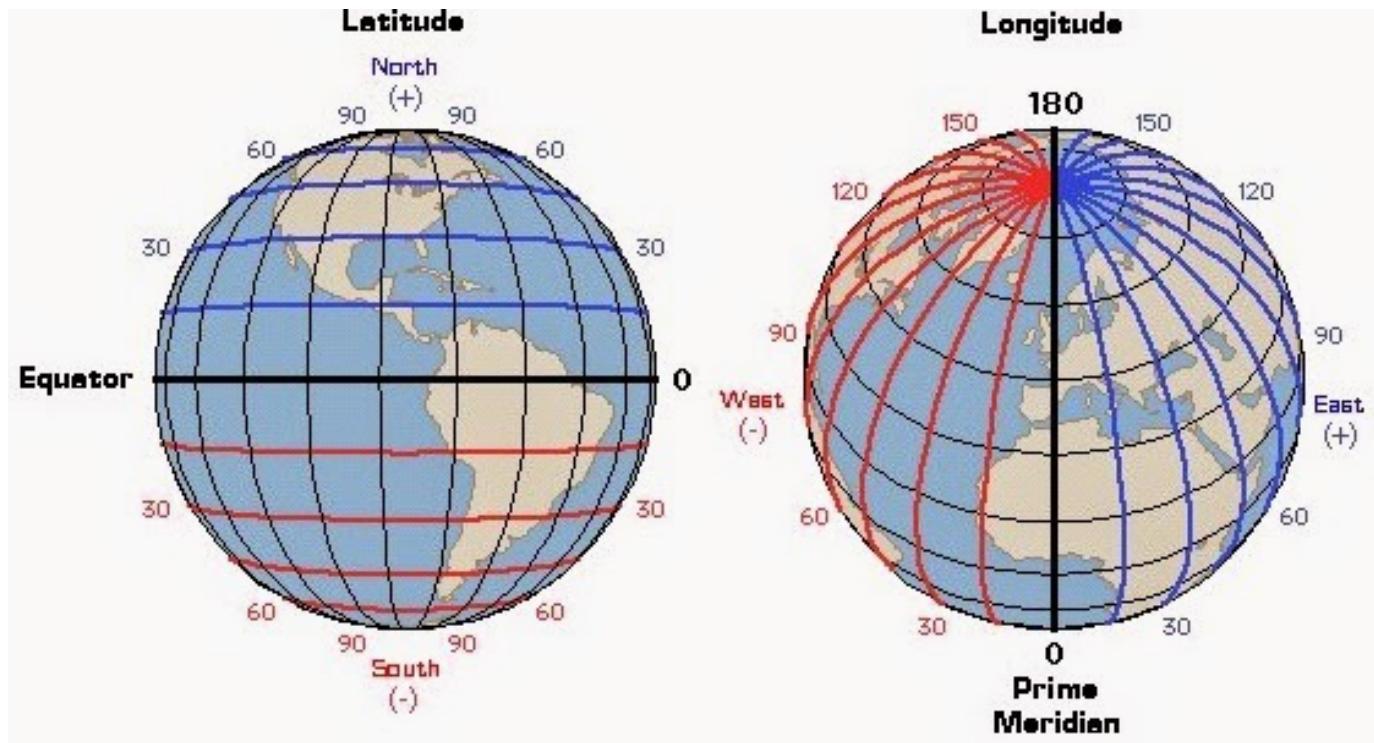
7.3.1.1 Sistem Koordinat Kartesius Koordinat Kartesius bukan merupakan satu-satunya jalan untuk menunjukkan kedudukan suatu titik pada bidang. Karena bentuk geometris di alam tidak selalu berupa kotak-kotak atau persegi panjang, namun ada kalanya berbentuk lingkaran[22]. Sistem koordinat Kartesius pada dua dimensi umumnya didefinisikan dengan dua buah sumbu yang saling tegak lurus antara satu dengan yang lainnya, yang keduanya terletak pada satu bidang (bidang xy). Sumbu horizontal(x), dan sumbu vertikal(y). Lalu, pada sistem koordinat tiga dimensi, ditambahkan sumbu yang lain yang sering diberi label z. Sumbu-sumbu tersebut ortogonal antar satu dengan yang lainnya. Titik pertemuan antara kedua sumbu, titik asal, pada umumnya diberi label 0. Setiap sumbu juga memiliki besaran panjang unit, dan setiap panjang tersebut diberi tanda dan membentuk semacam grid. Untuk mendeskripsikan suatu titik tertentu pada sistem koordinat dua dimensi, nilai absis(x), lalu diikuti dengan nilai ordinat(y). Dengan demikian, format yang dipakai selalu (x dan y) dan urutannya tidak dibalik-balik.

Gambar 7.3 Tanda panah yang ada pada sumbu berarti panjang sumbu nya tak terhingga pada arah panah tersebut.

Pilihan huruf-huruf didasari oleh konvensi, yaitu huruf-huruf yang dekat akhir(x dan y) yang digunakan untuk menandakan variabel dengan nilai yang tidak diketahui, sedangkan huruf-huruf yang lebih dekat awal digunakan untuk menandakan nilai yang diketahui. Karena kedua sumbu saling bertegak lurus satu sama lain, bidang xy terbagi menjadi empat bagian yang disebut dengan kuadran, yang pada Gambar 3 ditandai dengan angka I, II, III, dan IV. Menurut konvensi yang berlaku, keempat kuadran tersebut diurutkan mulai dari yang kanan atas (kuadran I), melingkar melawan arah jarum jam (lihat Gambar 3). Pada kuadran I, kedua koordinat (x,y) bernilai positif. Pada kuadran II, koordinat x bernilai negatif dan koordinat y bernilai positif. Pada kuadran III, kedua koordinat mempunyai nilai negatif, dan pada kuadran IV, koordinat x bernilai positif dan y bernilai negatif. (lihat gambar 7.4 di bawah ini).

7.3.1.2 Sistem Koordinat Polar Pada sistem koordinat polar, sepasang koordinat polar suatu titik ditulis dengan (,)[22]. Konsep dari sudut dan radius sudah diterapkan oleh orang-orang pada zaman dahulu se-abad sebelum masehi. Para astronom Yunani dan astrolog Hipparchus (190-120 BCE) menemukan sebuah tabel dari fungsi dawai yang memberikan panjang dawai dari setiap sudut dan terdapat referensi dari penggunaan koordinat polar untuk mengetahui posisi bintang.

Sejak abad ke-8 yang lalu, astronom mengembangkan cara untuk mengira-ngira dan menghitung arah dari mekah, ka'bah beserta jaraknya-dari seluruh lokasi dari bumi. Penghitungan penting yaitu penggantian dari koordinat polar ekuatorial dari



Gambar 7.3 Keempat kuadran sistem koordinat Kartesius

nilai nilai Kuadran

	x	y
I	> 0 > 0	
II	< 0 > 0	
III	< 0 < 0	
IV	> 0 < 0	

Gambar 7.4 Nilai x dan y pada Kuadran I,II,III,IV

mekkah kedalam bentuk koordinat polar hampir sama pada sistem yang merupakan pusat dari lingkaran besar melewati daerah yang dilewati dan kutub bumi, serta sudut polar yaitu garis yang melewati daerah tersebut dan titik antipodal.

Dalam Method of fluxion (tertulis 16711) Sir Isaac Newton menentukan hubungan antara koordinat polar, yang kemudian ia sebut dengan tujuh cara untuk spiral, dan sembilan sistem koordinat.

7.4 Geometri Koordinat

Pembelajaran subtajuk-subtajuk Geometri Koordinat, yaitu jarak antara dua titik, pembahagian tembereng garis, luas poligon, persamaan garis lurus, garis lurus selari dan garis lurus serenjang, persamaan lokus yang melibatkan jarak antara dua titik dan menentukan hubungan antara pencapaian responden dalam topik pelajaran[23].

7.4.1 Sketsa Grafik Garis

Sketsa grafik garis merupakan salah satu materi yang membahas mengenai penggambaran grafik garis lurus pada bidang kartesius berdasarkan persamaan yang diberikan. Materi ini mirip dengan metode penggambaran garis yang ada atau diajarkan pada aljabar. Maka jika sudah menguasai materi aljabar, sketsa grafik garis bukan masalah untuk dipelajari. Dalam menggambar grafik garis lurus, pertama harus melakukan pencarian pada nilai x dan y pada bidang kartesius dari persamaan yang sudah ada. Setelah nilai x dan y pada bidang kartesius telah di-temukan tentu bisa menentukan titiknya dan langsung menggambar garis tersebut.

7.4.2 Persamaan Garis Lurus

Persamaan garis lurus dapat di definisi kan sebagai perbandingan selisih nilai x dan y yang sudah melangkah 2 titik pada garis. Persamaan garis lurus terdapat satu komponen Gradien yaitu kecenderungan sebuah garis, gradien biasa dilambangkan dengan huruf m. Dalam materi persamaan garis lurus terdapat materi pokok seperti menentukan “gradien” garis lurus, “kedudukan” garis lurus, “persamaan” garis melalui satu titik merupakan gradien, dan “persamaan” garis melalui dua titik.

7.4.3 Persamaan Lingkaran

Persamaan lingkaran adalah persamaan titik koordinat yang membentuk sebuah lingkaran pada bidang kartesius. Pada konsep ini jari lingkaran yang telah terbentuk adalah jarak dari himpunan titik koordinat ke titik pusat atau sebaliknya. Pada persamaan lingkaran yang dapat dipelajari seperti lingkaran yang memiliki pusat $(0,0)$, lingkaran yang memiliki pusat (a,b) .

7.4.4 Program Linear

Program linear adalah metode matematika yang digunakan untuk menyelesaikan soal-soal yang memiliki batas persamaan linear. Secara umum program linear terbagi atas 2 bagian yaitu fungsi kendala dan fungsi objektif. Penyelesaian program linear model matematika adalah suatu metode penerjemahan permasalahan ke dalam bentuk matematika, sehingga soal tersebut bisa diselesaikan secara matematis.

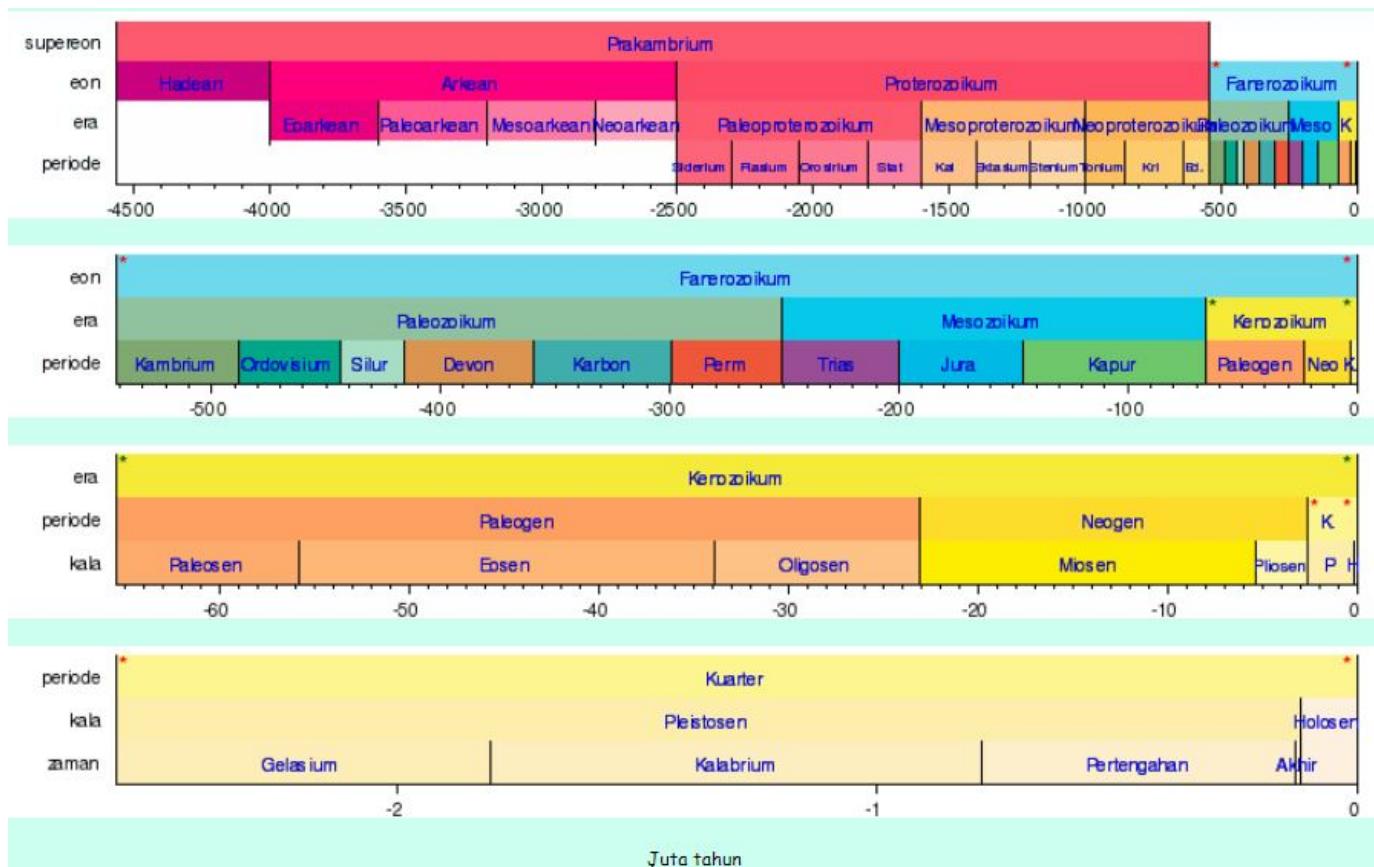
7.4.5 Pembelajaran Geometri Koordinat

Geometri Koordinat merupakan materi yang memberikan pengujian keterampilan dalam geometri dan aljabar. Jika sudah men- guasai materi geometri dan aljabar maka bisa dinyatakan geometri koordinat tidak lagi membuat sulit untuk dipelajari.

PENDAHULUAN SEJARAH PENENTUAN WAKTU

8.1 Sejarah Waktu

Sejarah Penentuan Waktu diurutkan secara kronologis dalam tabel skala waktu geologi yang dibagi menjadi beberapa interval sesuai analisis stratigrafi. Ada 4 garis waktu yang ada, garis waktu yang pertama menunjukkan waktu dari masa terbentuknya Bumi sampai waktu sekarang[20]. Skala waktu kedua menunjukkan eon terbaru dengan skala yang diperluas. Skala waktu kedua, ketiga, dan keempat merupakan sub bagian dari skala waktu sebelumnya yang ditunjukkan oleh tanda bintang. Alasan lain untuk memperluas skala waktu adalah HOLOSEN (jangka waktu) terakhir terlalu kecil untuk dapat ditampilkan dengan jelas pada skala waktu ketiga disebelah kanan. Gambar 8.1 adalah ilustrasi skala waktu.



Gambar 8.1 Gambar skala waktu yang ditetapkan.

8.2 Penentuan Waktu

Pada pembahasan diatas, kita telah membahas tentang sejarah waktu, dan skala waktu. Dari penjelasan tersebut, maka dibuatlah suatu pergerakan rotasi bumi. Gerakan ini disebut gerakan semu Matahari yang digunakan dalam penentuan waktu (jam).

8.2.1 Hari Matahari

Menurut artikel dari rachman planet mengemukakan bahwa satu hari matahari ditentukan oleh selang waktu antara dua kulminasi [24]. Kulminasi Atas disebut tengah hari (pukul 12.00) dan Kulminasi Bawah adalah saat tengah malam (pukul 24.00 atau pukul 00.00). Dalam kegiatan kita sehari-hari, satu hari matahari adalah waktu yang diperlukan Matahari bergerak semu mengelilingi Bumi, terhitung mulai titik Kulminasi Atasnya hingga kembali lagi ke titik Kulminasi Atasnya lagi. Dari hasil pengamatan Simamora, ternyata panjang hari matahari (semu) selama setahun berbeda-beda (tidak konstan), hal ini disebabkan:

1. Bentuk lintasan revolusi Bumi adalah elips. Dalam perputaran Bumi mengelilingi Matahari membuat lintasan berbentuk elips sehingga waktu lintasan mengelilingi Matahari (perihelium) pergerakannya cepat dan pada waktu lintasan terjauh dengan Matahari (aphelium) pergeseran nya pada ekliptika lambat. Dengan adanya kecepatan gerak Bumi mengelilingi matahari (revolusi) tidak sama dengan rotasi bumi tetap, maka terjadilah pergeseran semu pada ekliptika tidak seragam, akibatnya saat Matahari mencapai kulminasi nya tidak sama. Artinya panjang hari pada hari matahari setiap harinya tidak sama.
2. Inklinasi ekliptika pada ekuator langit. Oleh sebab perputaran Bumi pada sumbunya (rotasi) miring maka kedudukan bidang ekuator langit dengan bidang ekliptika membentuk sudut 23,50°. Akibat dari rotasi bumi itu, sepanjang tahun Matahari seolah-olah bergeser ke arah Utara atau ke arah Selatan. Enam bulan berada di belahan Utara dan enam bulan di belahan bumi Selatan. Gerakan tersebut menyebabkan terjadi perbedaan panjang hari terutama pada lintang geografis sedang atau tinggi, baik di belahan Bumi Utara atau belahan Bumi Selatan.

8.2.2 Hari Bintang

Hari Bintang adalah selang waktu yang diperlukan sebuah Bintang untuk berkombinasi

pada tempat yang sama pada saat berikutnya dalam meridian langit yang sama dari suatu tempat. Satu hari bintang (sehari semalam bintang) adalah waktu yang diperlukan sebuah bintang (lebih umum disebut titik Aries) bergerak semu mengelilingi Bumi mulai dari titik Kulminasi Atasnya sampai ke titik Kulminasi Atasnya lagi. Hari Matahari lamanya 24 jam sedangkan hari Bintang adalah 23 jam 56 menit. Jadi perbedaan antara hari Matahari dan hari Bintang adalah $1/365 \times 24$ jam atau $1/365 \times 1440$ menit yaitu 3 menit 56 detik dibulatkan menjadi 4 menit. Jadi pada hari berikutnya Bintang tersebut akan berkulminasi 4 menit lebih awal. Anda dapat menghitung selama 30 hari menjadi 30×4 menit yaitu 120 menit atau 2 jam.

Jadi setelah 12 bulan (1 tahun) yaitu 12×2 jam = 24 jam. Dengan demikian setahun kemudian baru Bintang tersebut akan berkulminasi pada jam yang sama. Jadi seolah-olah langit perbintangan berputar kurang lebih 10 setiap hari. Satu tahun Bintang 3600 dibagi 365,25 hari Matahari.

Sebagai contoh, pada tanggal 23 Maret Bintang Regulus berkulminasi pada pukul 08.00, pada tanggal 23 April bintang tersebut berkulminasi pukul 06.00, dan pada tanggal 23 Mei Bintang tersebut berkulminasi pukul 04.00. Dari pendataan tersebut maka:

1 hari bintang = 1 hari matahari dikurangi 4 menit, 1 jam bintang = 1 jam matahari dikurangi 1 detik. Dari perhitungan yang dijelaskan maka ada tanggal-tanggal istimewa untuk waktu Bintang dan waktu Matahari, yaitu:

Tanggal 21 Maret, pukul 00.00 waktu Bintang = pukul 12.00 waktu Matahari, Tanggal 21 Juni, pukul 00.00 waktu Bintang = pukul 06.00 waktu Matahari, Tanggal 23 September, pukul 00.00 waktu Bintang = pukul 00.00 waktu Matahari Tanggal 22 Desember, pukul 00.00 waktu Bintang = pukul 18.00 waktu Matahari.

Jadi hubungan antara Lokal Siderial Times (LST) atau waktu Bintang, dengan Local Civil Times (LCT) dan jumlah hari perbedaan sejak 22,7 September (dibulatkan 23 September) sampai tanggal yang ditentukan adalah: $LST = LCT + (4.69/70) D$. Catatan: $4.69/70 = 4 \times 69/70 = 3$ menit 56 Detik.

8.2.3 Hari Matahari Menengah/Matahari Khayal dan Perata Waktu

Dari penjelasan diatas kita, dapat mengetahui bahwa Matahari bukanlah penunjuk waktu yang sangat tepat. Oleh sebab itu, untuk keperluan pembagian waktu yang tepat yang kita gunakan sehari-hari, para ahli pun mendasarkan perhitungannya pada Matahari khayal. Matahari khayal ini adalah Matahari yang dianggap atau dimisalkan ada,

yang kecepatan pergeseran nya hampir serupa dengan pergeseran Matahari sebenarnya.

Perbedaannya adalah Matahari khayal ini bergeser sepanjang ekuator langit dengan kecepatan pergeseran yang tetap (konstan) atau seragam, sehingga panjang satu hari matahari khayal = panjang rata-rata hari matahari sebenarnya. Oleh karena itulah hari matahari khayal disebut pula hari matahari menengah.

Pada saat matahari menengah inilah didasarkan pembagian waktu pada jam yang kita gunakan sehari-hari, karena setiap hari matahari menengah panjangnya tetap sama sepanjang tahun.

1 hari matahari menengah = 24 jam waktu matahari menengah 1 jam waktu matahari menengah = 60 menit waktu matahari menengah 1 menit waktu matahari menengah = 60 detik waktu matahari menengah

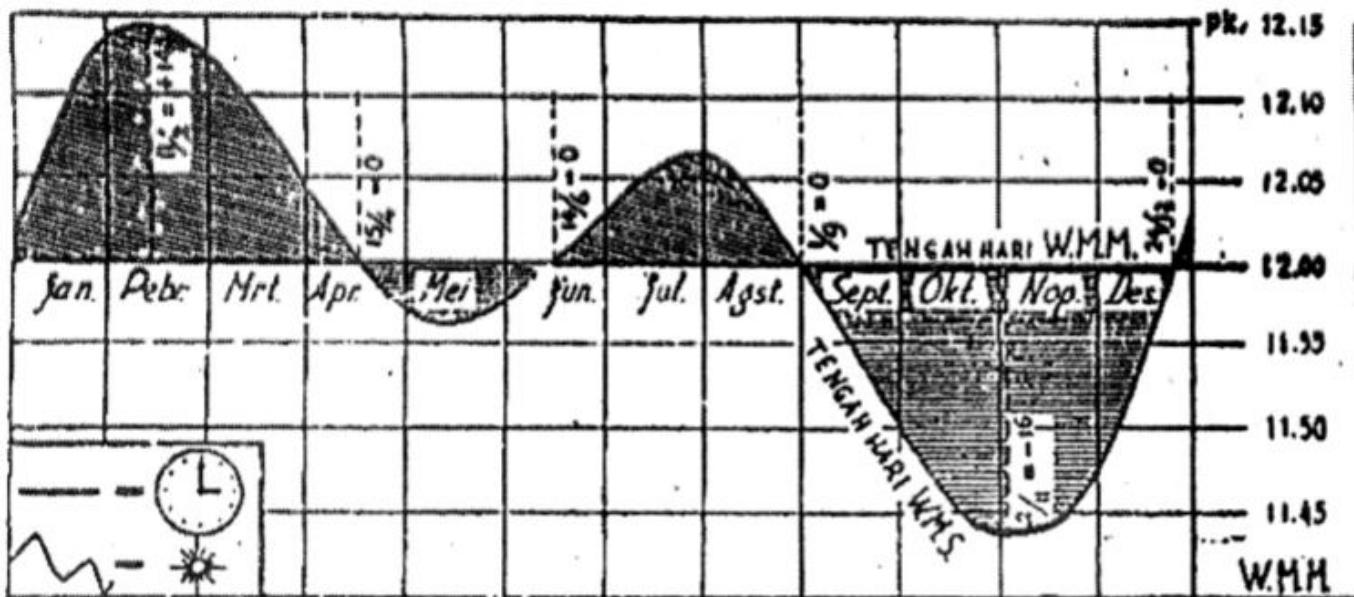
Bandingkan 1 hari matahari menengah = 24 jam waktu matahari menengah (jam kita) = 24 jam 4 menit waktu bintang (24 jam 3menit 57detik) 1 hari bintang = 24 jam waktu bintang = 23 jam 6menit waktu matahari menengah

(tepat nya 23 jam 56 menit 4 detik)

Waktu matahari menengah dimulai ketika matahari menengah berada pada titik Kulminasi bawahnya (pukul 00.00 waktu matahari menengah), cara membedakannya mulai

dari waktu bintang yang dimulai pada saat titik Aries berada pada titik Kulminasi Atasnya (pukul 00.00 waktu bintang). Hari Matahari Menengah kadang-kadang lebih sedikit pendek dari Hari Matahari Sebenarnya tetapi terkadang lebih panjang. Perbedaan maksimal hanyalah sampai kira-kira seperempat jam. Perbedaan waktu ini disebut Perata Waktu, dengan rumus: Perata Waktu = Hari Matahari Menengah - Hari Matahari Sebenarnya
 (Simamora,P., 1975: 72)

Perata waktu ini dinyatakan dengan tanda positif (+) jika matahari menengah mendahului matahari sebenarnya dan tanda negatif (-) jika terjadi sebaliknya. Perata waktu terbesar terjadi pada 11 Februari, yaitu + 14 menit dan 2 November, yaitu 16 menit. Dalam satu tahun terjadi empat kali panjang hari matahari menengah sama dengan pajang hari matahari sebenarnya, yaitu 15 April, 14 Juni, 1 September, dan 24 Desember. Pada hari-hari ini perata waktunya adalah 0 menit. Untuk lebih jelasnya perhatikan gambar 8.2.



Gambar 8.2 Perata Waktu

Dari gambar 8.2 kita dapat mengetahui pula bahwa sekitar bulan Januari, Februari Maret, Juli, dan Agustus matahari sebenarnya lebih lambat sampai titik Kulminasi atasnya, sehingga sore lebih lama terangnya.

Contoh: Pada tanggal 2 November jam di tangan kita (waktu matahari menengah) menunjukkan pukul 12.00, tetapi Matahari di langit masih belum tiba di titik Kulminasi Atasnya, baru -16 menit kemudian hal itu terjadi, yaitu pada pukul 11.44 waktu matahari menengah. Sebaliknya, pada bulan Oktober, November, dan Desember matahari menengah lebih lambat daripada matahari sebenarnya. Pagi hari Matahari telah terbit sedangkan jam kita masih menunjukkan kurang dari pukul 06.00. Pada sore harinya pukul 06.00 sudah gelap. Hal ini terjadi pada sekitar khatulistiwa (termasuk di Indonesia), di daerah-daerah sedang dan kutub tentunya berbeda.

8.2.4 Greenwich Mean Time(GMT)

Greenwich Mean Time (GMT) adalah tempat yang menjadi

patokan waktu dunia berada. Jika ditentukan dengan penentuan waktu GMT lebih mudah kita dapat menghitung waktu-waktu di seluruh permukaan Bumi. Bagi daerah yang berada di belahan barat (meridian barat) waktu setempat adalah waktu GMT ditambah dengan hasil kali perbedaan meridian dengan 4 menit sedangkan daerah yang berada di belahan timur (meridian timur) waktu setempat adalah waktu GMT dikurangi dengan hasil kali antara selisih meridian dengan 4menit. cara perumusannya dengan menggunakan: $LMT = GMT + (M.4)$ (Dardjosoemartp, dkk.,1991: 445)

$LMT = \text{Local Mean Time} / \text{Waktu Setempat}$ $GMT = \text{Greenwich Mean Time} / \text{waktu GMT} + = + \text{bila di BB}$ dan $- \text{bila di BT}$
 $(M.4) = \text{meridian (bujur)} \times 4 \text{ menit}$

8.3 Waktu Standar

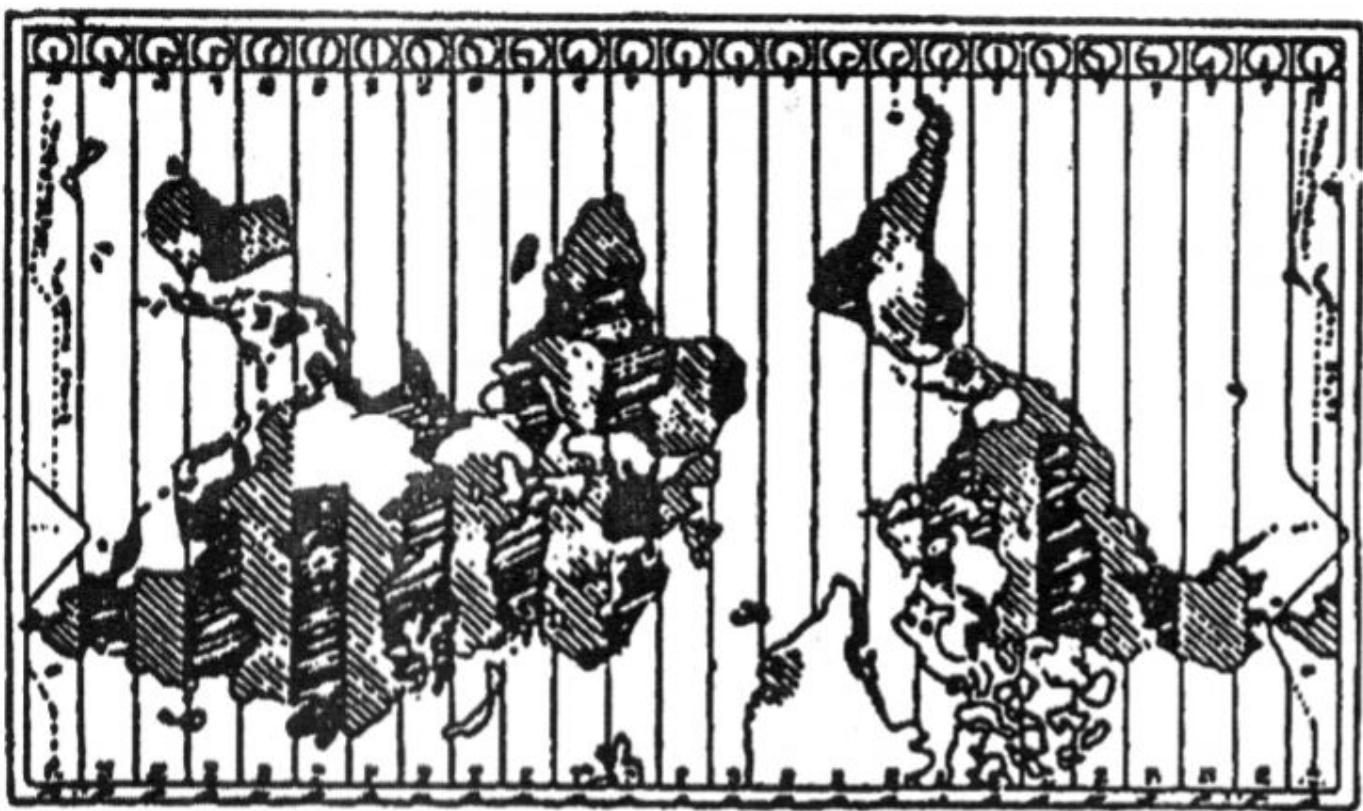
Tempat-tempat yang terletak pada garis meridian yang sama, mempunyai waktu yang sama. Jika demikian, seluruh permukaan Bumi terdapat 360 waktu yang bedanya 4 menit.

Hal ini tentu rumit dalam kehidupan sehari-hari. Oleh sebab itu, disepakatilah untuk membagi permukaan Bumi atas 24 daerah waktu saja yang disebut waktu standar.

Waktu standar disebut juga Zone Time, yaitu waktu yang ditetapkan setiap selisih 150 adalah 60 menit (1 jam) dengan lingkup daerah yang berada pada 00° 150° atau 150° 300°, dan seterusnya baik di Bujur Timur maupun Bujur Barat.

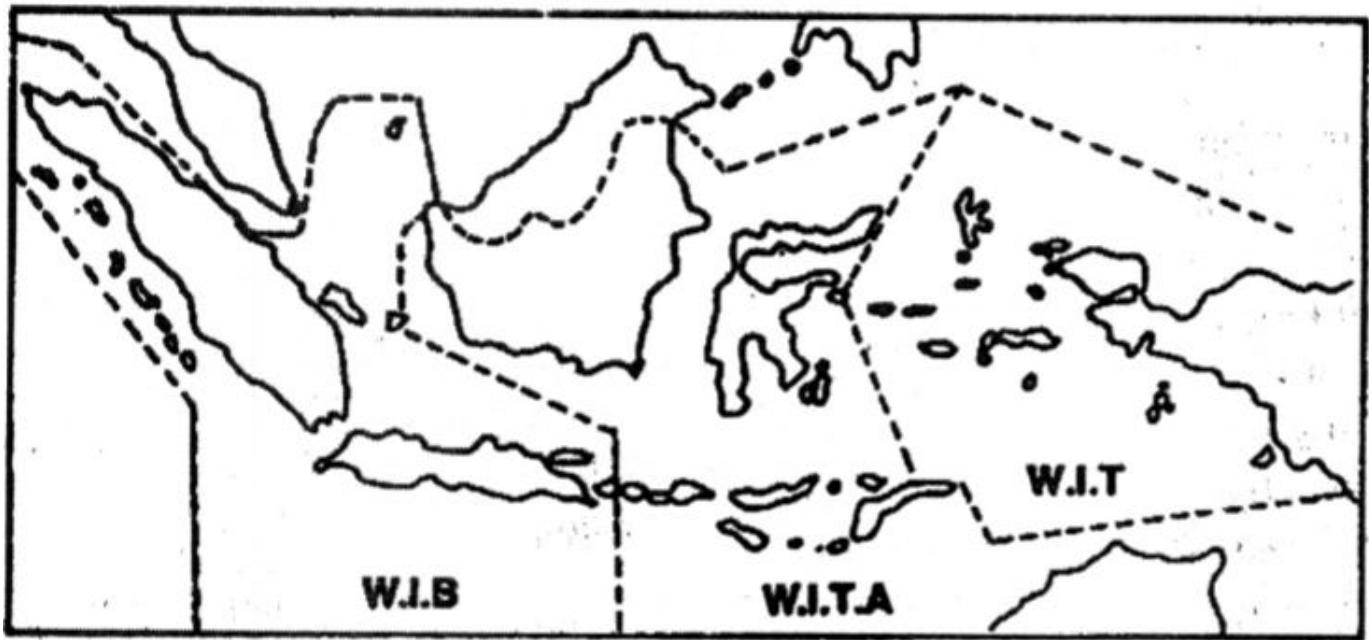
Kongres Internasional memutuskan tentang garis-garis meridian (International Meridian Conference) di Washington mene-tapkan waktu standar dunia yang dibagi menjadi 24 daerah berdasarkan perbedaan meridian 150°. Setiap daerah mempunyai selisih waktu 1 jam. Akan tetapi berdasarkan pembagian wilayah ke pemerintahan atau kontinen (pulau/benua) maka ada sedikit

pergeseran. Batas yang terdapat pada 1800 BT dan 1800 BB berupa garis yang berkelok-kelok. Perhatikan gambar 8.3 di bawah ini:



Gambar 8.3 Pembagian Daerah Waktu di Dunia.

Setiap negara mempunyai pembagian daerah waktu yang berbeda-beda karena letak pada meridiannya berbeda. Indonesia terletak antara 95°BT - 141°BT. Oleh karena Indonesia mempunyai rentang meridian $141^{\circ} - 95^{\circ} = 46^{\circ}$, maka Indonesia dibagi menjadi 3 daerah waktu, yakni Waktu Indonesia bagian Barat (WIB), Waktu Indonesia bagian Tengah (WITA), dan Waktu Indonesia bagian Timur (WIT) dengan selisih satu jam. Untuk lebih jelasnya, perhatikan gambar 8.4.



Gambar 8.4 Pembagian Waktu di Indonesia Tahun 1964

Indonesia mempunyai tiga meridian standar, yaitu meridian 105°BT untuk daerah WIB, 120°BT untuk daerah WITA, dan 135°BT untuk WIT. Dengan demikian waktu lokalnya (LMT)

masing-masing adalah waktu Greenwich ditambah 105/15 untuk WIB, 120/15 untuk WITA, dan 135/15 untuk WIT. Jika waktu GMT pukul 12.00, maka: WIB = $12.00 + (105/15 = 7)$ yaitu pukul 19.00, WITA = $12.00 + (120/15 = 8)$ yaitu pukul 20.00, dan WIT = $12.00 + (135/15 = 9)$ yaitu pukul 21.00 (Hidayat,B.,1978: 42).

BAB 9

PENDAHULUAN SEJARAH PENANGGALAN

9.1 Sejarah penanggalan

Penanggalan merupakan salah satu sebuah mahakarya yang bisa ditemukan oleh umat manusia. Manusia mempelajari dan memanfaatkan alam [Matahari,Bulan dan Bintang] untuk menghitung pergantian tanggal,bulan dan juga tahun. umumnya penanggalan digunakan untuk mengetahui waktu yang telah dilewati oleh umat manusia. Adanya sistem penanggalan ini membuat manusia dapat mengingat seluruh kejadian dan peristiwa yang terjadi di dunia ini. Menurut artikel dari setyanto berdasarkan benda langit yang digunakan sebagai dasar perhitungan sistem penanggalan dapat dikategorikan menjadi 3 kelompok[25].

9.1.1 Solar calendar/Kalender Surya

Kalender surya menggunakan pergerakan bumi mengelilingi matahari sebagai acuannya. Sistem kalender surya ini biasa digunakan oleh orang-orang eropa. Beberapa contoh kalender yang menggunakan sistem ini yaitu:

9.1.1.1 Julian calendar/Kalender Julian Kalender julian merupakan contoh kalender yang menerapkan sistem surya menurut artikel dari rachman planet[24].Kalender ini telah digunakan bahkan 45 tahun sebelum masehi. Awalnya ketika Julius Caesar memimpin pemerintahan romawi terjadi kekacauan pada perhitungan kalender yang menyebabkan Julis Caesar saat itu mengakhirinya dengan membuat perhitungan kalender sendiri dengan ketentuan: 1)Satu tahun ditetapkan 565,25 Hari 2)Tahun biasa, yaitu tiga tahun berturut-turut yang harinya berjumlah 365 Hari 3)Tahun Kabisat, yaitu tahun keempat ditambah satu hari menjadi 366 Hari. Tambahannya dilakukan pada bulan februari yang jika pada tahun biasa 28 hari pada tahun kabisat ini menjadi 29 hari 4)Titik permulaan musim semi/bunga ditetapkan pada tanggal 24 Maret 5)Permulaan tahun ditetapkan pada tanggal 1 Januari (Sebelumnya awal tahun ditetapkan pada tanggal 24 Maret) Meskipun kalender julian sudah sangat baik namun ternyata masih terdapat cacat pada kalender tersebut. Sebelum orang romawi menggunakan kalender julius caesar, orang romawi sudah menggunakan nama-nama bulan seperti:

1. Martius = 31 hari
2. Aprilis = 29 hari
3. Majus = 31 hari
4. Junius = 29 hari
5. Quintilis = 31 hari
6. Sextilis = 29 hari
7. September = 29 hari
8. October = 31 hari
9. November = 29 hari

10. Dcember = 29 hari
11. Januarius = 29 hari
12. Februarius = 28 hari

9.1.1.2 Gregorian calendar/Kalender Gregorius Pada tahun 1582 Masehi Paus Gregorius menyaksikan musim semi/bunga pada tanggal 11 maret, bukan lagi pada tanggal 24 maret seperti pada kalender julian. Kemudian paus gregorius memperbaikinya dengan cara:

1)Musim semi/bunga ditetapkan pada tanggal 21 Maret 2)Tahun biasa menjadi 365 hari dan tahun kabisat menjadi 366 hari
Kalender gregorius lebih dikenal dengan nama kalender masehi yang jumlah hari pada setiap bulan dan penetapan awal tahun seperti yang digunakan kalender umumnya saat ini. Kalender masehi dimulai dari tanggal 1 januari tahun 1, pukul 00.00.Penamaan bulan pada kalender gregorius yang digunakan hingga sekarang:

1. January = 31 hari
2. February = 28/29 hari
3. March = 31 hari
4. April = 30 hari
5. May = 31 hari
6. June = 30 hari
7. July = 31 hari
8. August = 31 hari
9. September = 30 hari
10. October = 31 hari
11. November = 30 hari
12. December = 31 hari

9.1.2 lunar calendar/Kalender candra

Pembahasan Kalender hijriah terkait dengan sistem penanggalan yang berpedoman pada pergerakan Bulan tampak dari Bumi yaitu ketika Matahari dan Bulan yang berada pada posisi bujur astronomi yang sama. Konjungsi merupakan pergerakan pada posisi Bulan dan Matahari yang telah disepakati sebagai batas penentuan secara astronomis pada Kalender Hijriah.

Bulan yang berkonjungsi searah dengan Matahari akan tampak gelap pada permukaannya ketika dilihat dari Bumi dengan bentuk cahaya sabit kecil. Bulan baru adalah piringan kecil Bulan yang muncul setelah mengalami satu putaran penuh pada fase Bulan mengelilingi Bumi. Kemunculan hilal (bulan baru) merupakan penentuan awal bulan dalam Kalender Hijriah di Indonesia, terkhusus pada bulan Ramadhan, Syawal, dan Zulhijah. Kalender merupakan sistem pengorganisasian waktu yang berfungsi sebagai penanda perhitungan dalam jangka panjang. Kalender hijriah termasuk jenis Kalender yang penanggalannya berpatokan pada Bulan ketika mengorbit kepada Bumi. Perbedaan antara tahun syamsia dan tahun kamariah yaitu umur hari dalam satu tahun yang 11 hari juga berbeda dalam penentuan awal perhitungan hari. Penanggalan kamariah memiliki perhitungan yang dimulai sejak terbenamnya Matahari dan berakhir ketika Matahari terbenam pada hari esoknya.

Sistem penanggalan Islam atau kalender hijriah adalah sistem penanggalan yang memiliki dua belas bulan, dimulai sejak Bulan baru hingga penampakan bulan baru berikutnya berkisar selang waktu antara 29 sampai 30 hari. refolusi bulan mengelilingi bumi memiliki bentuk lintasan yang elips dengan kecepatan tempuh total dalam satu tahun adalah 354 hari 48 menit dan 34 detik. Bulan sebagai salah satu komponen penting dalam penanggalan kamariah yakni merupakan satelit tunggal yang dimiliki Bumi. Bulan memiliki 3 pergerakan, diantaranya pergerakan rotasi atau Bulan berputar pada porosnya, revolusi terhadap bumi dan revolusi bersamaan dengan bumi terhadap matahari.

9.1.2.1 Sejarah Kalender Hijriah [25] Pada saat Sebelum peristiwa haji Wada yang dilaksanakan oleh Nabi dan kaum Muslimin, sistem penanggalan masyarakat Arab di Makkah kala itu masih menggunakan konsep penanggalan al-Nas. Keberadaan istilah waktu al-Nas tersebut telah mempersulit untuk mengurutkan fenomena/peristiwa yang terjadi sebelum haji Wada. Hal ini dikarenakan aturan penggunaan waktu al-Nas tidak berjalan dengan baik. Bangsa Arab dikenal sering mundur dan memajukan kegiatan-kegiatan yang dilakukan pada bulan-bulan Haram sesuai dengan kebutuhannya.⁴ Hal inilah yang menjadikan penanggalan masyarakat Arab sebelum Haji Wada dapat dikatakan tidak konsisten.

Maksud istilah waktu al-Nas (waktu pengunduran) yaitu diundurnya waktu untuk melaksanakan suatu kegiatan pada waktu tertentu. Salah satunya adalah pengunduran waktu ibadah haji oleh masyarakat Arah ketika itu. Mereka terkadang melaksanakan ibadah haji pada waktunya, terkadang pula pada bulan Muharam, afar, dan bulan-bulan lainnya di antara dua belas bulan. Dampaknya, adalah hal-hal yang mereka yang biasanya dilakukan pada bulan-bulan haram menjadi terabaikan. Hal ini dikarenakan pada saat mereka sedang melaksanakan ibadah haji, mereka bertemu dengan pembunuh ayah mereka, atau bertemu dengan pembunuh sanak saudara mereka, yang menyebabkan mereka membala dendam pada waktu tersebut. Padahal Allah telah

Kalender 2015 Masehi / 1436 Hijriyah

Januari							Februari							Maret							April																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
MINGGU		SENIN		SELASA		RABU		KAMIS		JUM'AT		SABTU		MINGGU		SENIN		SELASA		RABU		KAMIS		JUM'AT		SABTU																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
1		2		3		4		5		6		7		8		9		10		11		12		13		14		15																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
11		12		13		14		15		16		17		18		19		20		21		22		23		24		25																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
25		26		27		28		29		30		31		1		2		3		4		5		6		7		8																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
1 Januari : Tahun Baru Masehi 2015		10 KULON		11 EGI		12 HAWA		13 PEGI		14 WAGE		15 KULON		16 HAWA		17 PEGI		18 KULON		19 HAWA		20 PEGI		21 KULON		22 HAWA		23 PEGI																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
3 Januari : Maulid Nabi Muhammad SAW		24 KULON		25 HAWA		26 PEGI		27 KULON		28 HAWA		29 PEGI		30 KULON		31 HAWA		1 KULON		2 HAWA		3 PEGI		4 KULON		5 HAWA		6 PEGI																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
19 Februari : Tahun Baru IMLEK 2056																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
21 Maret : Hari Raya Nyepi (Tahun baru Saka 1937)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
1 April : Wafat Isa Al Mash																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
infonews.web.id																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
Mei							Juni							Juli							Agustus																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
MINGGU		SENIN		SELASA		RABU		KAMIS		JUM'AT		SABTU		MINGGU		SENIN		SELASA		RABU		KAMIS		JUM'AT		SABTU		MINGGU		SENIN		SELASA		RABU		KAMIS		JUM'AT		SABTU																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
26		27		28		29		30		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28		29		30		31		1		2		3		4		5		6		7		8		9		10		11		12		13	

pada proses pencapaian kesatuan dalam beribadah yaitu dapat diawali dengan menyepakati penggunaan kalender tunggal yang digunakan bersama, sedangkan pelaksanaan ibadah dikembalikan kepada masing-masing. Berikut adalah nama bulan dan hari pada kalender Hijriah berdasarkan pada hisab urfi:

1. Muharram = 30 hari
2. Shafar = 29 hari
3. Rabiul Awwal = 30 hari
4. Rabiul Akhir = 29 hari
5. Jumadil Awwal = 30 hari
6. Jumadil Akhir = 29 hari
7. Rajab = 30 hari
8. Shaban = 29 hari
9. Ramadhan = 30 hari
10. Syawal = 29 hari
11. Dzulka' idah = 30 hari
12. Dzhulhijjah = 29/30 hari

9.1.3 Lunisolar calendar/kalender surya candra

Menurut wicaksono dalam artikelnya Lunisolar kalender merupakan sistem kalender candra yang disesuaikan dengan matahari [26]. Karena kalender candra dalam 1 tahun mempunyai 11 hari lebih cepat dari kalender surya, maka dalam kalender surya candra memiliki bulan interkalasi(bulan tambahan/bulan ke -13)setiap 3 tahun, agar kembali seusai dengan perjalanan matahari. beberapa contoh kalender yang mengacu pada sistem surya candra adalah kalender imlek/cina, saka, dan budha. Semua kalender tersebut tidak ada yang sempurna ,karena jumlah hari dalam satu tahun itu tidak bulat, dan untuk memperkecil error itu maka dibuat kesepakatan sehari lebih panjang atau terdapat bulan tambahan dalam kalender cina pada tahun kabisat[26]. Pada kalender surya, pergantian hari terjadi tengah malam dan awal setiap bulan (tanggal 1) yang tidak tergantung pada posisi bulan dan pada kalender candra dan surya candra pergantian hari terjadi ketika matahari terbenam dan awal setiap bulan adalah saat konjungsi(imlek, sakka, budhha) atau dalam Hijriah saat munculnya hilal.

Bagian II

DASAR DASAR PEMETAAN

DASAR PEMETAAN MENGENAL BANGUN RUANG

10.1 Bangun Ruang

Bangun ruang merupakan suatu bagian ruang yang dibatasi oleh himpunan titik-titik yang terdapat pada seluruh permukaan bangun tersebut. Permukaan bangun tersebut disebut sisi. Bangun ruang memiliki tiga unsur, yaitu panjang : merupakan suatu dimensi dalam benda yang menunjukkan sebuah jarak antar ujung satu ke ujung lainnya. lebar : merupakan lintasan dalam sebuah bidang. tinggi : merupakan ukuran sebuah objek yang diukur secara vertikal. Bangun ruang memiliki volume. Rumus volume umum pada bangun ruang adalah $panjang(p) \times lebar(l) \times tinggi(t)$. Tujuan menghitung volume adalah untuk menghitung berapa banyak ruang yang dapat diisi atau ditempati pada suatu objek.

Sisi bangun ruang adalah suatu himpunan pada titik-titik yang terdapat pada permukaan atau yang membatasi suatu bangun ruang tersebut [27]

Dalam memilih model untuk permukaan atau sisi, dapat karena kedudukan semua unsur bangun ruang dapat diamati untuk dialihkan dalam gambar[28]. Ada beberapa contoh benda yang mewakili gambar bangun ruang 10.1.

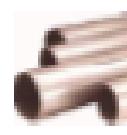
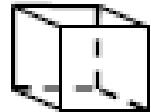
Bangun ruang sering disebut bangun 3 dimensi karena memiliki 3 komponen utama sebagai berikut.

1. Sisi merupakan bidang pada bangun ini memiliki ruang yang membatasi antara bangun ruang dengan ruangan sekitarnya
2. Rusuk merupakan pertemuan antar dua sisi yang berupa ruas garis pada bangun.
3. Titik sudut merupakan titik hasil pertemuan rusuk yang berjumlah tiga atau lebih

Unsur-unsur Bangun Ruang Sisi, rusuk dan titik sudut. Sebagai mengingatkan bahwa setiap model bangun ruang pasti memiliki sisi, rusuk, dan titik sudut , kecuali bola, tabung dan kerucut. Bangun ruang, limas, prisma, dan sisi, rusuk, titik sudut serta dikembangkan pada diagonalsasi, diagonal ruang, dan garis-garis sejajar. menggunakan model bangun ruang yang transparan melihat sisi bangun ruang tersebut, model transparan, bangun ruang dengan model transparan ini juga dapat untuk menggambar bangun ruang, karena semua unsur bangun ruang dapat diamati untuk dialihkan dalam gambar. Setelah mengamati, menelusuri, dan memahami unsur-unsur bangun ruang tersebut.

Jenis-Jenis Bangun Ruang yang umum dikenal adalah:

1. kubus merupakan bangun ruang yang dibatasi oleh enam buah bidang sisi berbentuk persegi dengan ukuran yang sama.
2. balok yaitu bangun ruang dengan dibatasi dengan enam bidang sisi yang memiliki bentuk persegi panjang yang setiap sepasang-pasang sejajar dan sama ukurannya.
3. prisma yaitu adalah sebuah bangun ruang yang diberikan batas oleh dua buah daerah segitiga yang sejajar sehingga tiga daerah persegi panjang tersebut yang saling berpotongan menurut garis-garis yang sejajar.
4. limas merupakan bangun ruang yang dibatasi oleh sebuah daerah segi empat dan empat daerah segitiga yang mempunyai satu titik sudut persekutuan.
5. kerucut merupakan bangun ruang yang dibatasi oleh sebuah bidang lengkung yang simetris terhadap porosnya yang melalui titik pusat lingkaran tersebut.

Bentuk Bangun Ruang	Bentuk Benda			
				
Bola	bakso	kelereng	buah melon	semangka
				
Tabung	tong sampah	pipa pralon	kue astor	drum
				
Kubus	dadu	bak mandi	kotak kardus	puzzle warna
				
Balok	almari	kotak snack	kotak kapur	kotak TV

Gambar 10.1 beberapa kumpulan gambar yang termasuk dalam bangun ruang

6. tabung merupakan bangun ruang yang setiap sisinya dibatasi dengan dua bidang lingkaran yang sama-sama sejajar dan sama-sama ukurannya dan satu buah bidang yang memiliki jarak sama jauhnya ke arah poros dan sisi yang simetris ke arah porosnya itu akan memotong dua daerah bidang lingkaran tepat di kedua lingkaran itu .

7. Bola

Jenis-Jenis Bangun Ruang yang umum dikenal adalah dan di dalam kehidupan sehari hari:

1. Kubus : dadu, rubik
2. Balok : lemari, tv
3. Prisma : atap rumah, tenda pramuka
4. Limas : piramida, monas
5. Kerucut : nasi tumpeng yang berbentuk kerucut
6. Tabung : minuman kaleng, gas elpiji
7. Bola : bola basket, bola tenis

Dalam pembelajaran bangun ruang dan unsur-unsur nya maka harus diperkenalkan model-model bangun ruang, misalnya model kubus, balok, prisma, limas, tabung, kerucut, dan bola. Apabila diambil contoh - contoh dari benda-benda yang dapat ditemukan dalam kehidupan sehari-hari, misalnya kaleng roti untuk menunjukkan tabung, tumpeng untuk menunjukkan kerucut dan seterusnya. Yang tidak transparan, transparan dan kerangka. Hal tersebut akan lebih memudahkan dalam pemahaman bangun

ruang dan unsur-unsur nya, menentukan sifat-sifat bangun ruang, serta dapat menerjemahkan gambar dalam bangun ruang dan sebaliknya. Contoh di bidang bangun ruang yaitu dalam bidang geometri materi matematika bentuk bangun datar 2D maupun bangun ruang 3D. Manfaat yang dapat diperoleh dari penelitian memberikan gambaran 3D dari permodelan bangun geometri halnya alat peragaan dalam membangun siswa dalam mempelajari bentuk bangun geometri. Bangun ruang dalam bentuk geometris yang terdiri atas tiga dimensi (panjang lebar dan tinggi) bangun ruang yang di bahas di dalam geometri antara lain:

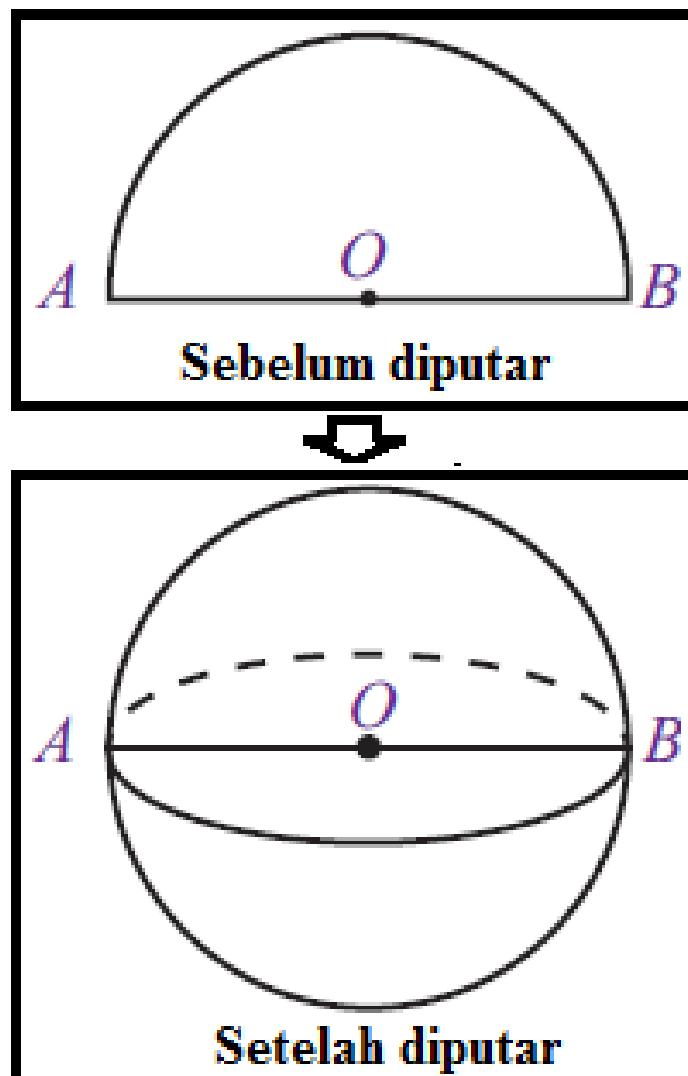
1. Kubus
2. Balok
3. Prisma
4. Limas
5. Tabung
6. Kerucut
7. Bola

Kebutuhan di bangun ruang dapat disimpulkan bahwa diperlukan:

1. Pengertian dan ciri-ciri berapa bangun datar dan bangunan ruangan.
2. Data rumus luas bangun datar.
3. Data rumus volume bangun datar dan bangun ruang.

Kebutuhan di sini sudah diperoleh dari buku matematika sekolah dasar.

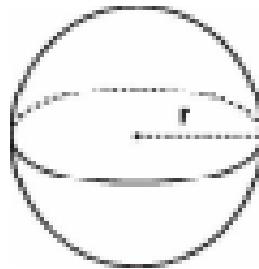
10.1.1 Bola



Gambar 10.2 contoh bola

Dalam bangun ruang, bola adalah bangun ruang tiga dimensi yang dibentuk sehingga lingkaran yang berjari-jari sama panjangnya dan berpusat pada satu titik yang sama. Bola merupakan bangun ruang sisi lengkung yang dibatasi oleh satu bidang lengkung. contoh bangun ruang bola dalam kehidupan sehari-hari adalah dalam sebuah olah raga sepak bola, basket, kasti, boling, dan sebagainya. bola dapat menggelinding dan dapat memantul dengan sempurna, karena tidak adanya sudut pada bola. Bentuk bumi pun seperti bola, terlihat pada sebuah dokumentasi dari pesawat ruang angkasa, maupun dalam hal perjalanan lurus, pasti akan kembali lagi ke tempat kita memulai perjalanan. Bola dapat dibentuk dari bangun setengah lingkaran yang diputar sejauh 360 pada garis tengahnya. Pada gambar 10.2 merupakan setengah lingkaran dengan diameter AB tersebut dan dapat diputar satu putaran dengan diameter sebagai suatu sumbu putar maka akan tampak gambar seperti di bawahnya yang disebut bangun ruang.

Bola merupakan bangun ruang sisi lengkung (BRSL) yang terjadi dari tumpukan empat buah lingkaran 10.3. Keempat



Gambar 10.3 contoh sisi lengkung

lingkaran ini dinamakan kulit bola. Kulit bola berada pada sisi luar bola atau mengelilingi bola [29].

Rumus bola:

a) Luas permukaan

$$L = 4\pi r^2 \quad (10.1)$$

b) Volume

$$V = \frac{4}{3}\pi r^3 \quad (10.2)$$

10.1.1.1 Sifat-sifat pada bola

- Memiliki 1 sisi yang berbentuk bidang lengkung (selimut bola)
- Tidak memiliki rusuk
- Tidak memiliki titik sudut

Adapun unsur-unsur bangun ruang bola yang terdapat pada gambar 10.4 sebagai berikut:

1. Titik pada titik O dinamakan titik pusat bola.
2. Ruas garis pada OA disebut sebagai jari-jari pada bola. Sebutkan jari-jari pada bola lainnya.
3. Ruas garis pada CD diberi nama sebagai diameter pada bola. Jika kita amati, ruas pada garis AB tersebut merupakan diameter bola. AB dapat pula disebut sebagai tinggi bola.
4. Sisi bola merupakan kumpulan titik - titik yang mempunyai jarak yang sama terhadap titik O. Sisi tersebut dinamakan selimut atau kulit bola.
5. Ruas garis ACB dinamakan tali busur bola.
6. Ruas-ruas pada garis selimut bola yaitu ACBDA dinamakan garis pelukis bola.

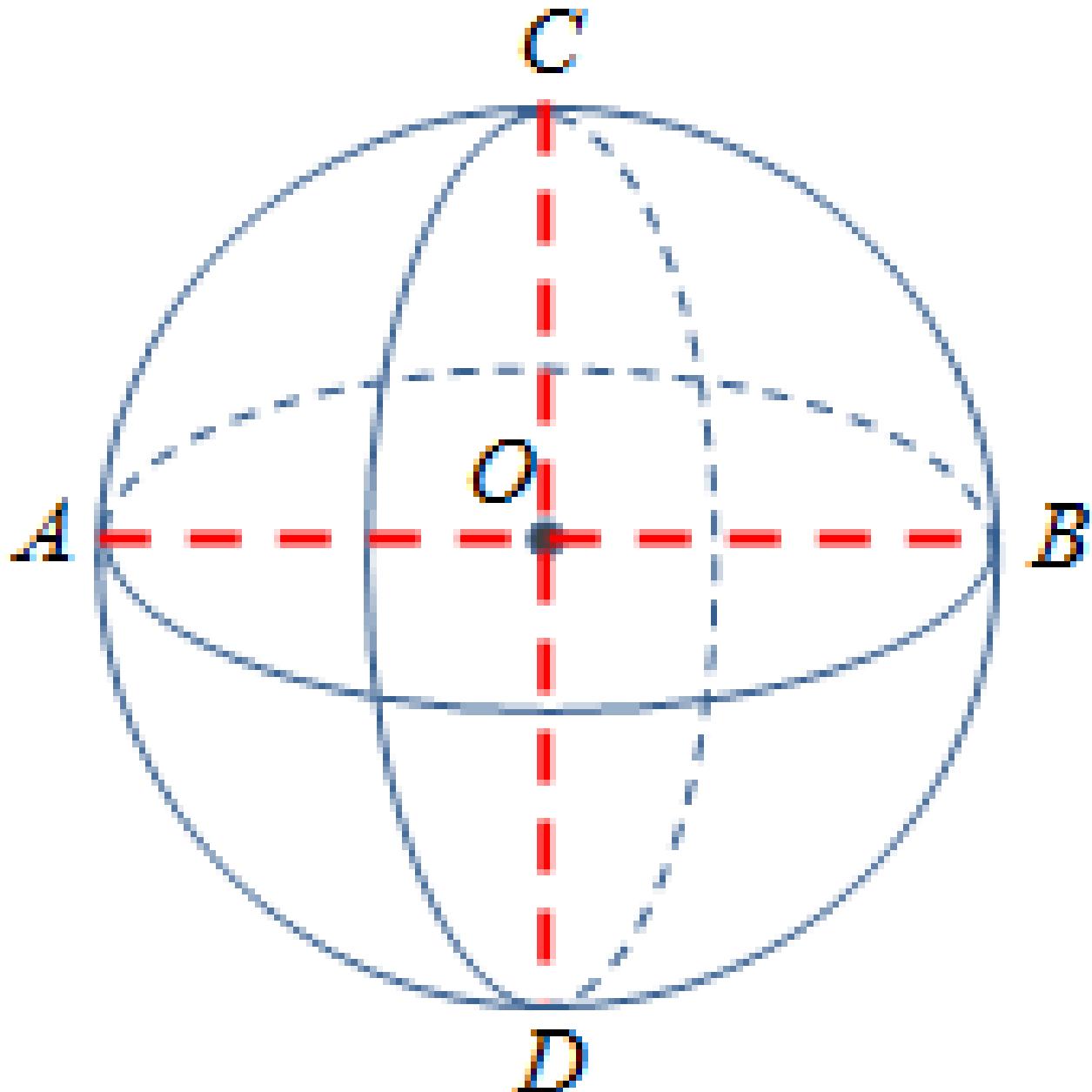
10.1.1.2 Konsep luas permukaan Bola Penentuan luas sisi (permukaan) bola dapat kita lakukan dengan sebuah percobaan archimedes, yaitu: Sebuah bola menempati sebuah tabung yang memiliki diameter dan tinggi tabung sama tepat dengan yang dimiliki oleh diameter bola, maka luas bola itu sama dengan luas selimut tabung 10.5. Berdasarkan gambar maka diperoleh :

Luas selimut tabung

$$L = 2pr.T = 2pr.2r = 4pr^2 \quad (10.3)$$

10.1.1.3 Konsep volume bola Apabila kita mengisi air ke dalam bangun bola secara penuh kemudian menuangkannya ke bangun ruang tabung maka air yang diperoleh adalah 2/3 bagian dari volume bangun ruang tabung tersebut. Dengan ketentuan bahwa kedua bangun tersebut memiliki jari-jari yang sama sehingga diperoleh:

$$\text{Volume bola} = 2/3 \cdot \text{volume tabung (silinder)} = 2/3 \cdot (pr^2 \cdot 2r) \quad (10.4)$$

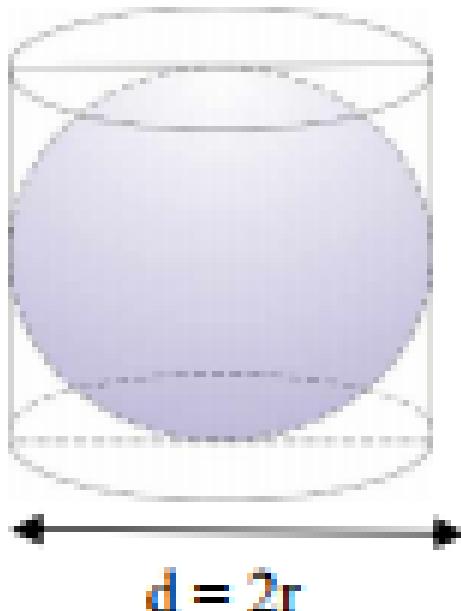


Gambar 10.4 contoh unsur bola

10.1.1.4 Asal-usul rumus permukaan bola Jika ingin mendapatkan rumus permukaan bola, kita mulai kegiatan berikut ini untuk menguji rumus tersebut.

1. Sediakan sebuah bola berukuran sedang seperti bola sepak atau bola basket.
2. Ukurlah setiap keliling bola tersebut menggunakan benang.
3. Lilit kan benang tersebut pada permukaan setengah bola sampai penuh, seperti gambar 10.6.
4. Buatlah persegi panjang dari kertas karton dengan ukuran panjang sama dengan keliling bola dan lebar sama dengan diameter bola seperti gambar 10.7.
5. Lilitkan benang yang telah digunakan untuk melilit permukaan setengah bola pada persegi panjang yang kamu buat tadi. Lilit kan sampai habis.
6. Jika kamu melakukannya dengan baik, tampak benang tersebut menutupi persegi panjang selebar jari-jari bola (r).
7. Hitunglah luas dari persegi panjang yang telah ditutupi benang tersebut.

$$\text{Luas permukaan setengah bola} = \text{luas persegi panjang} = pl = 2rr = 2r^2 \quad (10.5)$$



Gambar 10.5 sebuah bola yang terdapat dalam tabung, untuk mengukur luas permukaan tabung



Gambar 10.6 gambar bola

Jadi, luas permukaan bola dirumuskan sebagai berikut :

$$\text{Luas permukaan bola} (L = 4r^2) \quad (10.6)$$

Keterangan : L = luas permukaan bola. r = jari-jari bola. $\pi = 22/7$ atau 3,14

10.1.1.5 Asal-usul rumus volume bola Cara - cara untuk mengetahui rumus volume bola, dapat dilakukan dengan cara - cara seperti berikut ini:

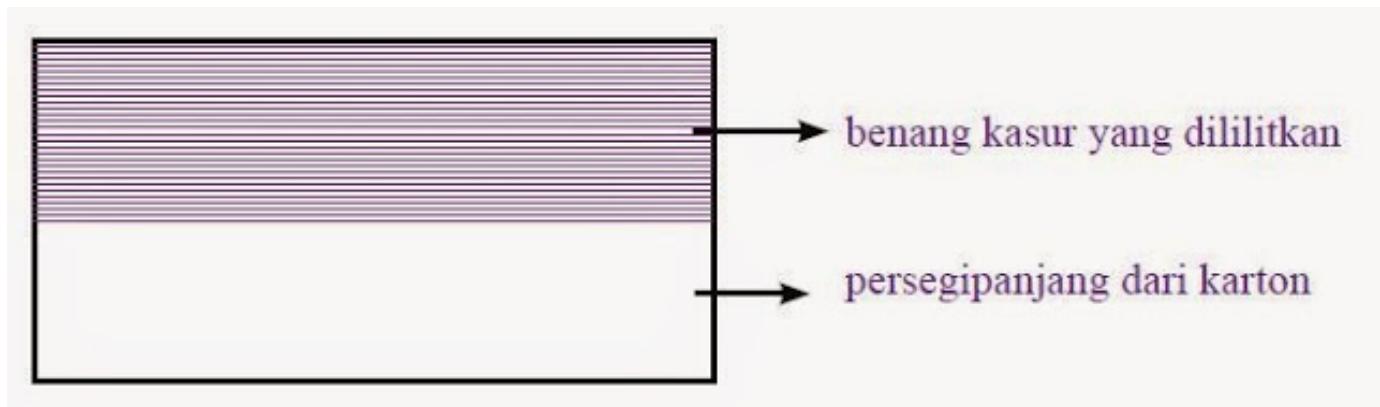
1. Siapkan sebuah tempat yang berbentuk setengah bola berjari-jari r (10.8) dan sebuah wadah yang berbentuk kerucut berjari-jari r dan tingginya $2r$ (10.9).
2. Isikan pasir ke 10.9 sampai penuh.
3. Pindahkan pasir di dalam 10.9 ke 10.8. Apakah yang terjadi?

Dari cara seperti di atas tersebut, dapat dilihat bahwasanya volume dari pasir yang dituangkan ke dalam wadah setengah bola tidak dapat berubah. Ini berarti, untuk membangun setengah bola, dan kerucut yang berjari-jari sama, dan tinggi kerucut sama dengan dua kali jari-jarinya maka:

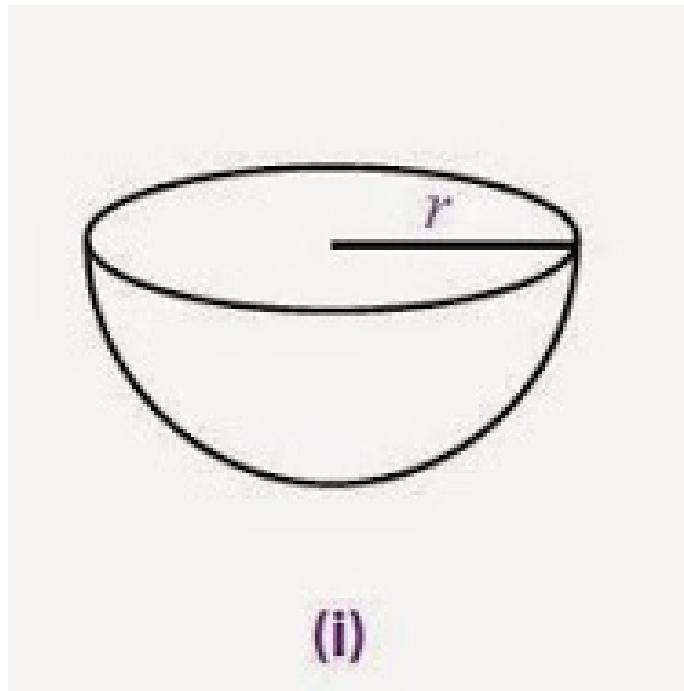
$$\text{Volume setengah bola} = \text{volume kerucut} / 2 = \text{volume bola} = 1/3\pi r^2 h / 2 = 2/3\pi r^2 (2r) = 4/3\pi r^3 \quad (10.7)$$

Jadi, volume bola tersebut dirumuskan sebagai berikut :

$$\text{Volume bola} (V = 4/3\pi r^3) \quad (10.8)$$



Gambar 10.7 beberapa kumpulan gambar yang termasuk dalam bangun ruang



Gambar 10.8 wadah dalam bola

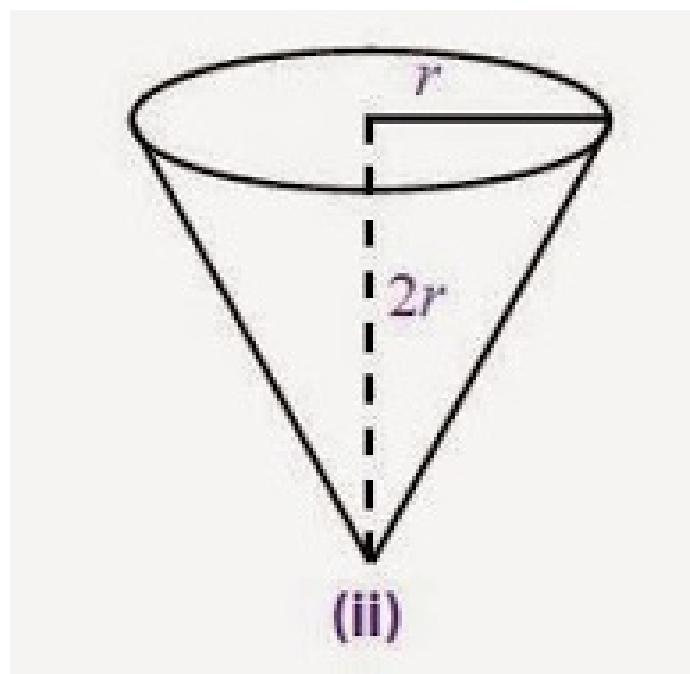
Keterangan : V = volume bola. r = jari-jari bola. $=22/7$ atau $3,14$.

Contoh soal : bola memiliki jari-jari 9 cm, hitunglah volume bola tersebut ?

Jawab : Diketahui : $r = 9$ cm Ditanyakan : volume bola ? Penyelesaian :

$$V = \frac{4}{3}\pi r^3 = \frac{4}{3}/3,14.(9)^3 = 3.052,08 \quad (10.9)$$

Jadi, volume bola tersebut $3.052,08 \text{ cm}^3$



Gambar 10.9 pasir dalam wadah

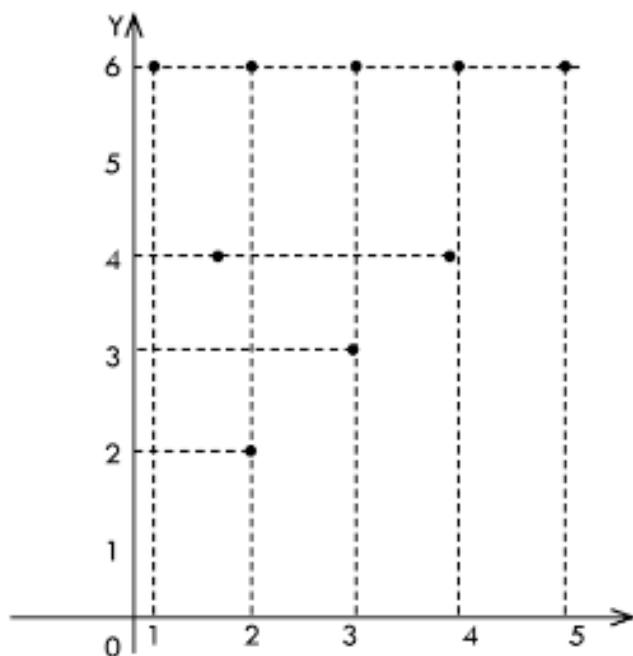
DASAR PEMETAAN KARTESIUS

11.1 Pengertian Diagram Kartesius

Diagram Kartesius adalah sistem koordinat yang terdiri dari dua sumbu yang berisi titik-titik sebagai simbol relasi. Domain sebagai sumbu horizontal dan ko domain sebagai sumbu vertikal. Pada koordinat kartesius daerah asal (domain) diletakkan pada sumbu X (sumbu mendatar) dan daerah kawan (ko domain) diletakkan pada sumbu Y (sumbu tegak). Sedangkan daerah hasilnya merupakan titik (noktah) koordinat pada diagram kartesius. Dari relasi di atas, dapat ditunjukkan diagram kartesius nya seperti di bawah:

Contoh:

Relasi X "faktor dari" Y dengan $X = \{1, 2, 3, 4\}$ dan $Y = \{2, 3, 4, 6\}$.



Gambar 11.1 hubungan antar titik pada diagram kartesius.

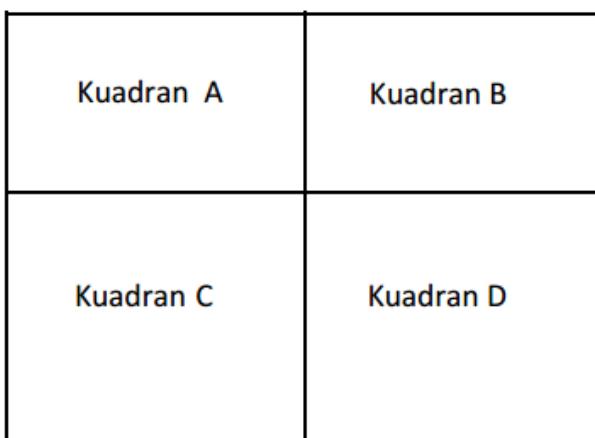
Diagram Kartesius merupakan suatu bangunan atas empat bagian yang batasi oleh dua buah garis yang berpotongan tegak lurus pada titik-titik (X, Y). Dimana X merupakan rata-rata dari rata-rata skor tingkat pelaksanaan atau kepuasan konsumen dari sebuah faktor atribut dan Y adalah rata-rata skor tingkat kepentingan seluruh faktor atau atribut yang mempengaruhi kepuasan konsumen.

Seluruhnya ada K faktor. Rumus berikutnya yang digunakan ada di gambar 11.2. Dimana K adalah Banyaknya faktor atau atribut yang mempengaruhi kepuasan konsumen.

$$\bar{\bar{Y}} = \frac{\sum_{i=1}^K \bar{Y}_i}{n} \quad \dots \dots \dots \quad (6)$$

Gambar 11.2 rumus mencari K faktor.

Dimana: K = Banyaknya faktor atau atribut yang mempengaruhi kepuasan konsumen Diagram Kartesius



Gambar 11.3 penentuan kuadran pada diagram kartesius.

Dari Gambar 11.3 terbadi menjadi beberapa kuadran Kartesius.

- Kuadran A : Pada posisi ini, jika dilihat dari kepentingan konsumen, atribut-atribut produk berada pada tingkat tinggi, tetapi jika dilihat dari kepuasannya, konsumen merasakan tingkat yang rendah, sehingga konsumen menuntut adanya perbaikan atribut tersebut.
 - Kuadran B : Pada posisi ini, jika dilihat dari kepentingan konsumen, atribut-atribut produk berada pada tingkat tinggi, dan dilihat dari kepuasannya, konsumen merasakan tingkat yang tinggi juga.
 - Kuadran C : Pada posisi ini, jika dilihat dari kepentingan konsumen, atribut-atribut produk kurang dianggap penting, tetapi jika dilihat dari tingkat kepuasan konsumen cukup baik.

Namun, konsumen mengabaikan atribut-atribut yang terletak pada posisi ini. Kuadran D Pada posisi ini, jika dilihat dari kepentingan konsumen, atribut-atribut produk kurang dianggap penting, tetapi jika dilihat dari tingkat kepuasannya, konsumen merasa sangat puas.

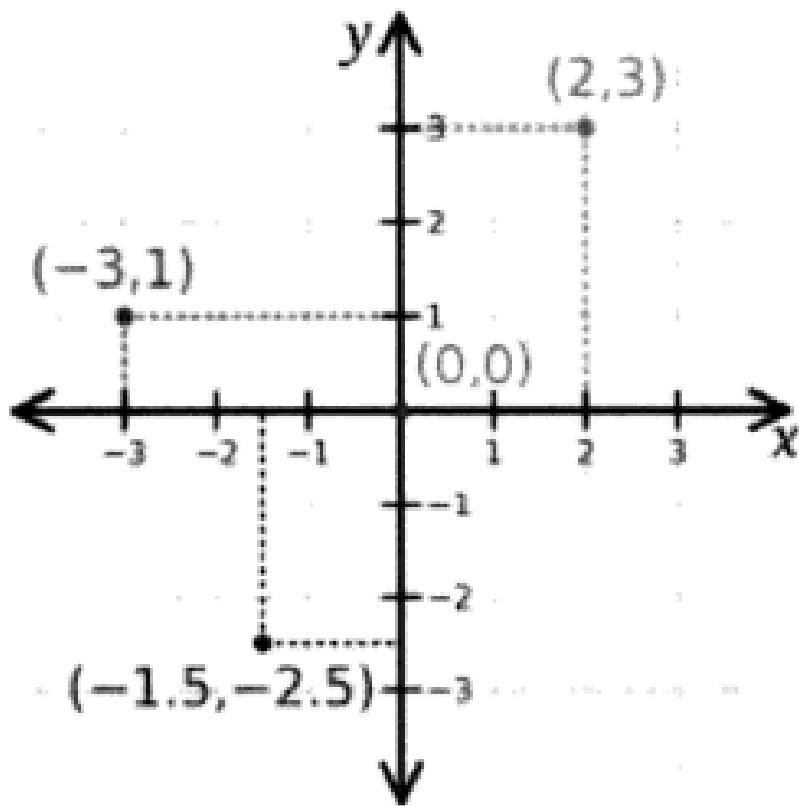
11.2 Penghitungan Rumus Diagram Kartesius

11.2.1 menghitung rumus, mencari titik

Kartesius digunakan untuk menentukan tiap titik dalam bidang dengan menggunakan dua bilangan yang biasa disebut koordinat x dan koordinat y.

Sebuah titik dalam Diagram Kartesius, mengandung dua buah informasi yakni sumbu (x , y) seperti tampak pada Gambar 1.2. Yaitu titik $(2, 3)$ adalah titik dimana nilai $x=2$ dan $y=3$. Daerah ini dikenal dengan kuadran I, dimana nilai x dan y adalah positif.

Dari dua buah titik diagram kartesius, bisa ditarik menjadi sebuah garis(gambar 11.5). Artinya pada sebuah garis memiliki titik awal



Gambar 1.2 Titik Dalam Diagram Kartesius

Gambar 11.4 penentuan titik pada kuadran kartesius.

11.3 Contoh Penerapan/Pemetaan Diagram Kartesius

Tujuan digunakannya diagram kartesius adalah untuk melihat secara lebih terperinci mengenai atribut-atribut yang perlu untuk dilakukan perbaikan.

Langkah-langkah sebelum memetakan data ke diagram kartesius ini, adalah terlebih dahulu dengan menentukan nilai rata-rata setiap atribut yaitu X dan Y, dimana nilai perhitungannya telah kita peroleh dari perhitungan yang dilakukan sebelumnya. Adapun hasil pembagian setiap atribut pada setiap kuadran ditampilkan pada gambar 2

Setelah dilakukan perhitungan menggunakan diagram kartesius didapat hasil atribut-atribut yang harus diperbaiki adalah atribut yang berada pada kuadran A.

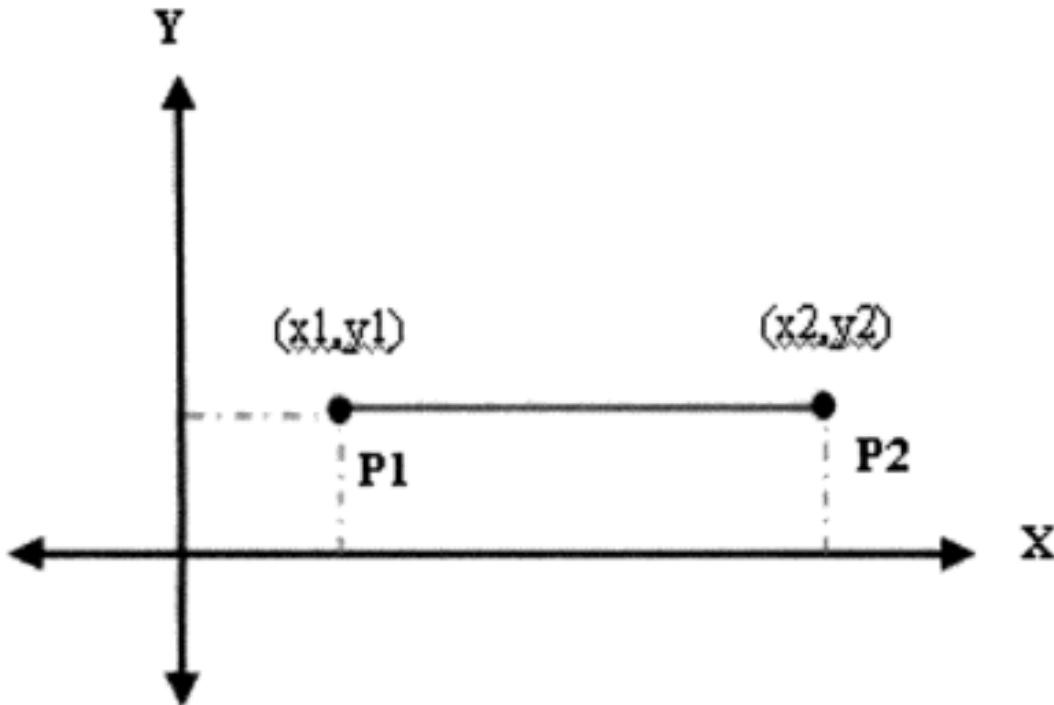
Adapun atribut yang harus diperbaiki pada kuadran A pada gambar 11.7.

Untuk atribut-atribut yang harus dipertahankan oleh pihak perusahaan setelah dilakukannya perhitungan menggunakan diagram kartesius adalah atribut-atribut yang berada pada kuadran B, karena pada atribut yang berada pada kuadran B dianggap pelanggan sudah dapat memenuhi apa yang mereka inginkan. Adapun atribut yang harus dipertahankan dapat dilihat pada gambar 11.8.

Atribut yang memiliki penilaian yang rendah karena atribut-atribut ini kurang dianggap penting oleh pelanggan dan perusahaan juga tidak memberikan pelayanan atau perhatian khusus, atribut ini dianggap tidak memberikan dampak yang besar bagi perusahaan. Adapun atribut-atribut yang berada pada kuadran C dapat dilihat pada gambar 11.9.

Untuk atribut yang ada pada kuadran D adalah atribut yang tidak dianggap penting bagi pelanggan, namun pihak perusahaan memberikan pelayanan yang berlebihan sehingga atribut ini dianggap berlebihan. Adapun atribut yang berada pada kuadran D dapat dilihat pada gambar 11.10.

Diagram Kartesius Dari hasil perhitungan yang telah dilakukan sebelumnya, terdapat 17 atribut yang perlu dilakukan perbaikan (Action) dan terdapat 10 atribut yang perlu mendapat perhatian untuk dipertahankan oleh pihak perusahaan (Hold). Diagram Kartesius Dari hasil pemetaan yang dilakukan pada diagram kartesius dapat terlihat beberapa atribut yang perlu untuk dilakukannya perbaikan dan atribut-atribut perlu untuk dipertahankan oleh pihak perusahaan yang terbagi kedalam kuadran-kuadran (A, B,



Gambar 1.3 Garis dalam Diagram Kartesius

Gambar 11.5 penentuan garis pada kuadran kartesius.

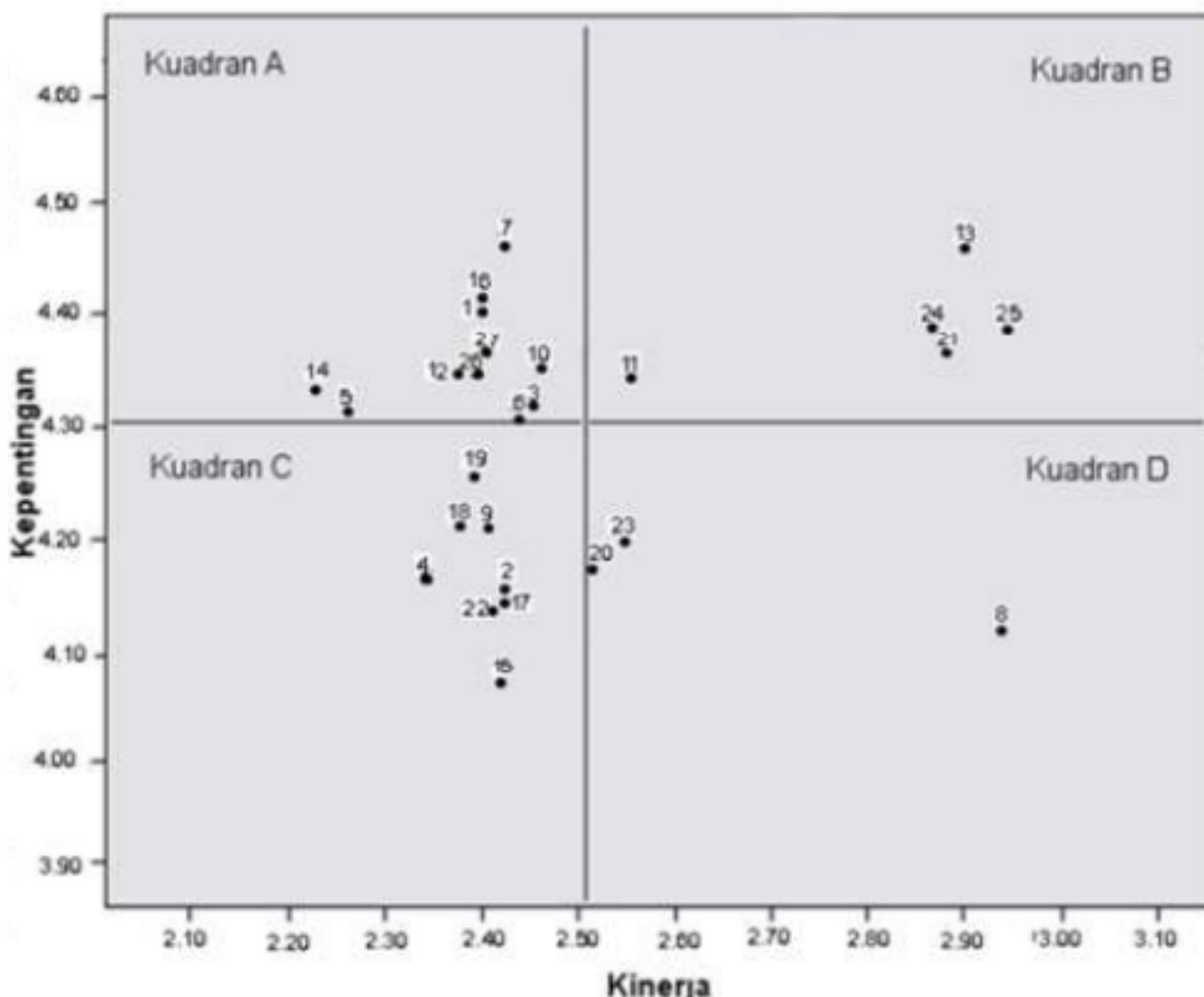
C dan D) sesuai dengan tingkat kesesuaian antara tingkat kepentingan pelanggan dan kinerja perusahaan, yaitu dengan tingkat kesesuaian sebesar 58.374. Adapun hasil pemetaannya adalah sebagai berikut: Kuadran A

Kuadran A adalah wilayah yang berisikan atribut-atribut yang dianggap penting oleh pelanggan, namun dalam kenyataannya atribut-atribut ini masih belum sesuai dengan yang diharapkan oleh pelanggan. Dalam hal ini perusahaan perlu melakukan perbaikan sebaik mungkin untuk meningkatkan kepuasan pelanggan terhadap atribut yang termasuk kedalam kuadran A. Dari diagram kartesius yang dibuat, diketahui bahwa atribut yang termasuk dalam kuadran A yaitu atribut 1, 3, 5, 6, 7, 10, 12, 14, 16, 26, 27. Adapun beberapa hal yang sebaiknya perlu dilakukan guna perbaikan atau penyesuaian terhadap beberapa hal yang menjadi prioritas diatas yang pertama antara lain perlunya dilakukan penambahan alat pendingin ruangan untuk dapat menjaga suhu ruangan demi kenyamanan pelanggan, Penambahan ukuran meja kasir agar barang-barang belanjaan yang telah dipilih tidak merepotkan pelanggan ataupun kasir. Selain itu juga perlu dilakukannya perbaikan ataupun pembersihan ruangan toilet dan pendukung lainnya seperti ketersediaan air sehingga pelanggan yang menggunakan akan merasa lebih nyaman, penambahan jumlah keran-jang belanjaan yang disediakan perusahaan, Lebih melengkapi jenis-jenis produk yang ditawarkan dengan mempertimbangkan tempat penyimpanan serta waktu-waktu tertentu seperti hari-hari besar nasional dan lain sebagainya, Memberikan pengarahan kepada para karyawan mengenai pentingnya berinisiatif dalam melayani pelanggan yang membutuhkan bantuan tanpa harus dimintai tolong terlebih dahulu oleh pelanggan. Dapat juga dilakukan penambahan papan informasi berupa lokasi produk yang tersedia untuk dapat mengurangi frekuensi terjadi atau timbulnya pertanyaan dari para pelanggan mengenai produk yang akan mereka beli, perbaikan ataupun penyesuaian secara berkala antara label-label harga yang tertera pada produk yang ditawarkan dengan perubahan-perubahan harga yang terjadi, penataan tempat parkir yang dapat dilakukan dengan memberikan garis-garis pembatas kendaraan, ataupun dengan menambahkan tukang parkir untuk dapat menanggulangi keamanan dan penataan tempat parkir kendaraan, penyusunan program-program promo secara berkala, seperti pemberian diskon dengan jumlah pembelian tertentu ataupun dengan memberikan voucher belanja dengan nilai tertentu untuk dapat lebih menarik pelanggan, dan sebaiknya perusahaan memiliki atau beberapa jenis produk tertentu yang diunggulkan dengan harga yang lebih murah dibandingkan dengan kompetitor lainnya sebagai penarik.

11.3.1 Kuadran B

Kuadran B adalah daerah yang memuat atribut-atribut yang dianggap penting oleh pelanggan, dan atribut-atribut tersebut dianggap telah sesuai dengan keinginan

pelanggan sehingga tingkat kepuasan pelanggan relatif lebih tinggi, sehingga perlu untuk dipertahankan oleh pihak perusahaan karena sudah bisa memberikan pelayanan sesuai dengan keinginan pelanggan sehingga konsumen merasa puas. Adapun atribut yang termasuk kedalam kuadran ini adalah: 11, 13, 21, 24, 25. Kuadran C Kuadran C adalah Daerah yang berisikan atribut-atribut



Gambar 11.6 Diagram Kartesius

yang dianggap kurang penting oleh pelanggan dan pada kenyataannya kinerja pihak perusahaan pun dinilai kurang memuaskan. Tetapi tidak

menutup kemungkinan Kuadran C pada waktu yang akan datang menjadi perhatian yang penting oleh pelanggan, sehingga perusahaan juga harus mempertimbangkan hal tersebut. Adapun atribut yang termasuk kedalam kuadran ini adalah: 2, 4, 9, 15, 17, 18, 19, 22.

11.3.2 Kuadran D

Kuadran D adalah wilayah yang memuat atribut-atribut yang dianggap kurang penting oleh pelanggan dan kinerja yang dilakukan oleh pihak perusahaan dirasakan terlalu tinggi atau berlebihan, sehingga perusahaan tidak perlu melakukan perbaikan. Adapun atribut yang termasuk kedalam kuadran ini adalah: 8, 20, 23.

11.4 Pengertian Bidang atau Diagram Kartesius

Dalam mempelajari materi himpunan, fungsi, dan persamaan garis lurus kita akan mengenal yang namanya bidang atau diagram Kartesius. Apa itu bidang atau diagram Kartesius?

Diagram Kartesius adalah sistem koordinat yang digunakan untuk meletakan titik pada penggambaran objek berdasarkan pemasukan nilai pada sumbu x dan nilai pada sumbu y dimana titik pertemuan ini nilai dari sumbu x dan sumbu y titik koordinat dibentuk. Jadi, diagram Kartesius digunakan untuk menentukan tiap titik dalam bidang dengan menggunakan dua bilangan yang biasa disebut koordinat x dan koordinat y dari titik tersebut. Di mana x disebut absis dan y disebut ordinat.

Titik-titik pada koordinat kartesius merupakan pasangan titik pada sumbu-x dan sumbu-y (x, y). Perpotongan antara sumbu-x dan sumbu-y di titik 0 (nol) disebut pusat koordinat. Untuk bagian atas sumbu y bernilai positif, sedangkan pada bagian bawah sumbu y bernilai negatif. Begitu juga pada sebelah kanan sumbu x bernilai positif, sedangkan pada sebelah kiri sumbu x bernilai negatif. Untuk contohnya silahkan lihat gambar di bawah ini.

Perhatikan diagram Kartesius pada gambar di atas. Warna ungu (violet) merupakan pusat koordinat yaitu titik $(0,0)$ yang artinya sumbu x dan y bernilai nol. Untuk warna hijau, pada sumbu x bernilai 2 dan sumbu y bernilai 3 maka koordinat dalam bidang kartesius ditulis $(2,3)$. Untuk warna merah, pada sumbu x bernilai -3 dan sumbu y bernilai 1 maka koordinat dalam bidang

Atribut	Pertanyaan
1	Suhu Ruangan toko yang nyaman
3	Ukuran meja kasir.
5	Kebersihan toilet
6	Jumlah keranjang belanjaan yang tersedia.
7	Kelengkapan produk yang ditawarkan
10	Cepat tanggap karyawan toko melayani konsumen dalam berbelanja
12	Keakuratan pembayaran pada kasir dengan label harga pada produk
14	Teraturnya penataan tempat parkir
16	Keamanan tempat parkir
26	Adanya potongan harga yang diberikan untuk beberapa jenis produk
27	Perusahaan memiliki beberapa jenis produk unggulan dengan harga yang lebih murah dibanding kompetitor lain.

Gambar 11.7 Hasil Perhitungan Diagram Kartesius pada Kuadran A

kartesius ditulis (3, 1). Sedangkan untuk warna biru, pada sumbu x bernilai 3 dan sumbu y bernilai 1 maka koordinat dalam bidang kartesius ditulis (1.5 , 2.5).

Menurut Wikipedia, istilah Kartesius digunakan untuk mengenang ahli matematika sekaligus filsuf dari Perancis bernama Descartes. Beliau memiliki peranan yang sangat besar dalam menggabungkan aljabar dan geometri (Kartesius adalah latinisasi untuk Descartes). Hasil kerjanya sangat berpengaruh dalam perkembangan geometri analytic, kalkulus, dan kartografi.

Atribut	Pertanyaan
11	Kemampuan para karyawan dalam membantu konsumen mendapatkan produk yang dibutuhkan
13	Keakuratan pengembalian uang pada konsumen
21	Karyawan bersedia membantu mengantarkan barang belanjaan konsumen ke kendaraan konsumen.
24	Harga produk yang ditawarkan cukup bersaing dengan kompetitor setempat.
25	Terdapat label harga pada produk

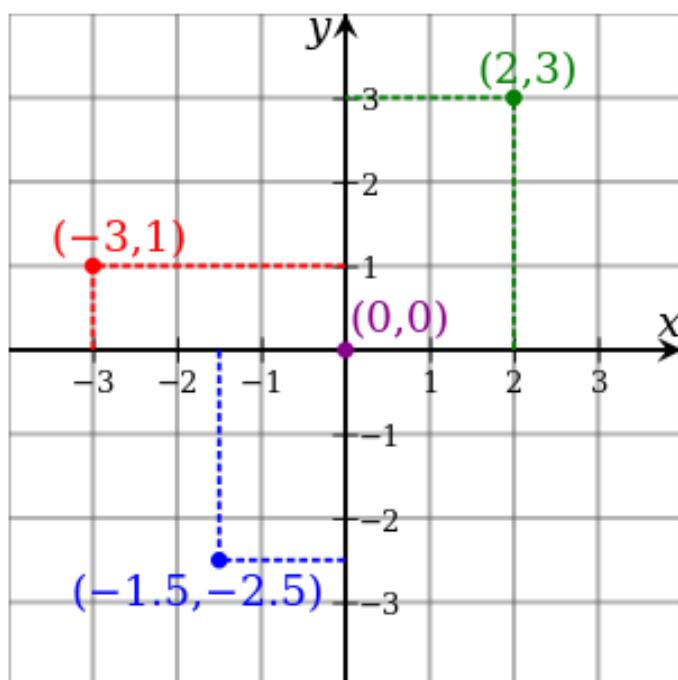
Gambar 11.8 Hasil Perhitungan Diagram Kartesius pada Kuadran B

Atribut	Pertanyaan
2	Jumlah kasir
4	Kerapian meja kasir.
9	Kondisi tempat pembuangan sampah yang memadai.
15	Biaya parkir kendaraan.
17	Pelayanan untuk penukaran produk yang sejenis jika terdapat keluhan pada konsumen.
18	Dapat melakukan penukaran produk yang telah dibeli dengan produk yang berlainan jenis/merk bila terjadi kesalahan dalam pembelian.
19	Pelayanan terhadap keluhan konsumen yang berbelanja
22	Promosi produk yang ditawarkan produsen dapat diproses melalui bantuan pihak toko.

Gambar 11.9 Hasil Perhitungan Diagram Kartesius pada kuadran C

Atribut	Pertanyaan
8	Kebersihan di dalam dan luar area
20	Karyawan berkenan membantu konsumen dalam memilih produk yang diinginkan
23	Harga produk yang ditawarkan sesuai dengan label harga pada iklan produsen

Gambar 11.10 Hasil Perhitungan Diagram Kartesius pada Kuadran D



Gambar 11.11 penentuan garis/titik dalam diagram kartesius

BAB 12

DASAR PEMETAAN GARIS KHATULISTIWA

Sejarah Garis Khatulistiwa Dan Prime Meridian

12.1 Definisi

12.1.1 pengertian garis khatulistiwa

Dalam sebuah artikel dari Muhammad Adieb yang menyebutkan bahwa garis khatulistiwa merupakan garis lintang dari 0 derajat sampai dengan 90 derajat di kutub bumi. Jadi, nilai lintang berkisar antara 0 sampai dengan 90. Di sebelah selatan garis khatulistiwa disebut lintang selatan (LS) dengan tanda negatif (-) dan di sebelah utara garis khatulistiwa disebut lintang utara (LU) yang diberi tanda positif (+)[30].

12.1.2 pengertian prime meridian

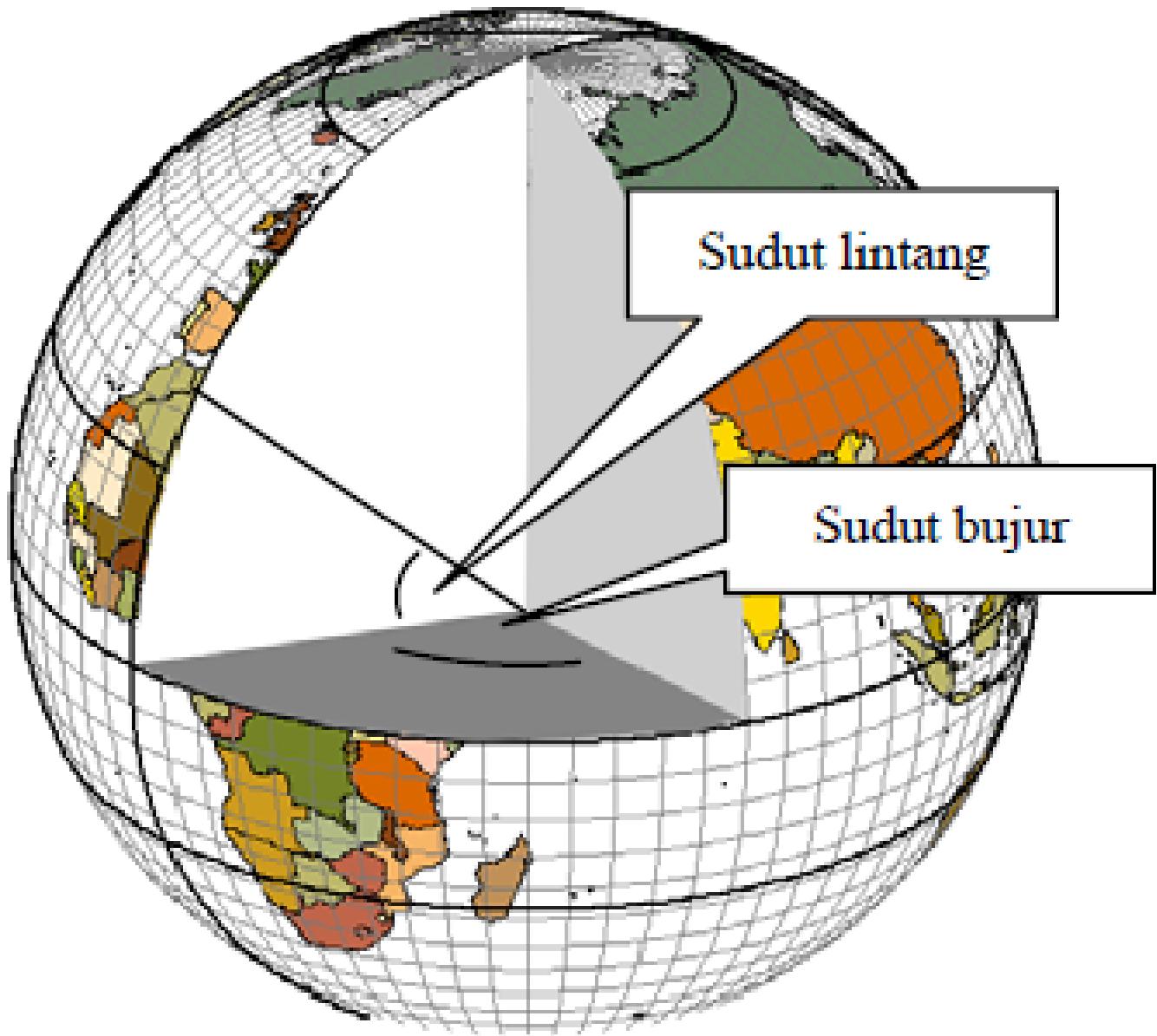
Dalam sebuah artikel dari Mohd Zuhdi yang menyebutkan bahwa Prime meridian atau meridian Greenwich adalah nilai koordinat garis bujur dimulai dari bujur 0 derajat yaitu di Greenwich, kemudian membesar ke arah timur dan barat sampai bertemu kembali di Garis batas internasional yaitu terletak di Selat Bering dengan nilai 180 derajat. [17]. Dalam sebuah artikel lain oleh Andi Sunyoto yang menyebutkan bahwa Prime meridian adalah sebuah garis virtual yang melewati sebuah kota bernama Greenwich di Inggris[31].

12.2 Kordinat dan Wilayah

12.2.1 sistem koordinat bumi

Menurut sebuah artikel dari Mohd Zuhdi yang menyebutkan bahwa dalam sebuah artikel dari Mohd Zuhdi yang menyebutkan bahwa sistem koordinat dimaksudkan untuk memberikan pengalamanan terhadap setiap lokasi di permukaan bumi. Pengalamanan dengan sistem koordinat didasarkan atas jarak timur sampai dengan barat dan utara sampai dengan selatan suatu tempat dari suatu titik pangkal tertentu. Jarak diukur dalam satuan derajat dengan sudut yang dibentuk dari titik pangkal ditetapkan yang berada di perpotongan belahan utara sampai dengan selatan bumi (garis khatulistiwa) dengan garis yang membelah bumi bagian timur sampai dengan barat melewati kota Greenwich di Inggris.

Pada gambar 12.1 menjelaskan tentang sudut lintang dan bujur pada bumi. Posisi suatu tempat di alamatkan dengan nilai koordinat garis bujur (longitude) dan lintang (latitude) yang melalui tempat itu. Garis bujur (longitude), biasanya juga disebut garis meridian, yaitu merupakan garis lurus yang menyambungkan dari kutub utara sampai selatan bumi. Nilai koordinat garis bujur ini dimulai dari bujur 0 derajat yaitu di Greenwich, kemudian membesar ke arah timur dan barat sampai bertemu kembali di Garis batas internasional yaitu terletak di Selat Bering dengan nilai 180 derajat. Garis bujur 0 derajat sering disebut prime meridian atau meridian Greenwich. Garis bujur ke arah barat diberi nilai negatif dan disebut bujur barat (west longitude) serta disingkat BB. Sedangkan garis bujur yang ke arah timur diberi nilai positif dan disebut bujur timur (east longitude) disingkat BT. Nilai koordinat nya didasarkan atas besarnya sudut yang terbentuk dari bujur 0 ke garis bujur tersebut melalui pusat bumi.



Gambar 12.1 menjelaskan tentang sudut lintang dan bujur pada bumi.

Adapun nilai koordinat lintang dimulai dari garis lingkaran khatulistiwa yang diberi nilai 0 derajat. Selanjutnya garis-garis lintang yang lain berupa lingkaran-lingkaran paralel (sejajar) khatulistiwa berada di sebelah utara dan selatan khatulistiwa. Lingkaran paralel di selatan disebut garis lintang selatan (LS) dan diberi nilai negatif, sedangkan lingkaran paralel di utara diberi nilai positif dan disebut garis lintang utara (LU). Nilai maksimum koordinat garis lintang adalah 90 derajat yaitu terletak di kutub-kutub bumi.

Lingkaran paralel yang merupakan representasi garis lintang ini semakin mengecil ukurannya dengan semakin jauh dari khatulistiwa. Sehingga jarak 1 derajat

timur sampai barat hanya beberapa meter saja. Itu sebabnya grid yang dibuat dari garis lintang dan garis bujur, tampak berupa bujur sangkar di khatulistiwa dan berubah menjadi persegi panjang di daerah dekat kutub. [17].

12.2.2 pemanfaatan prime meridian

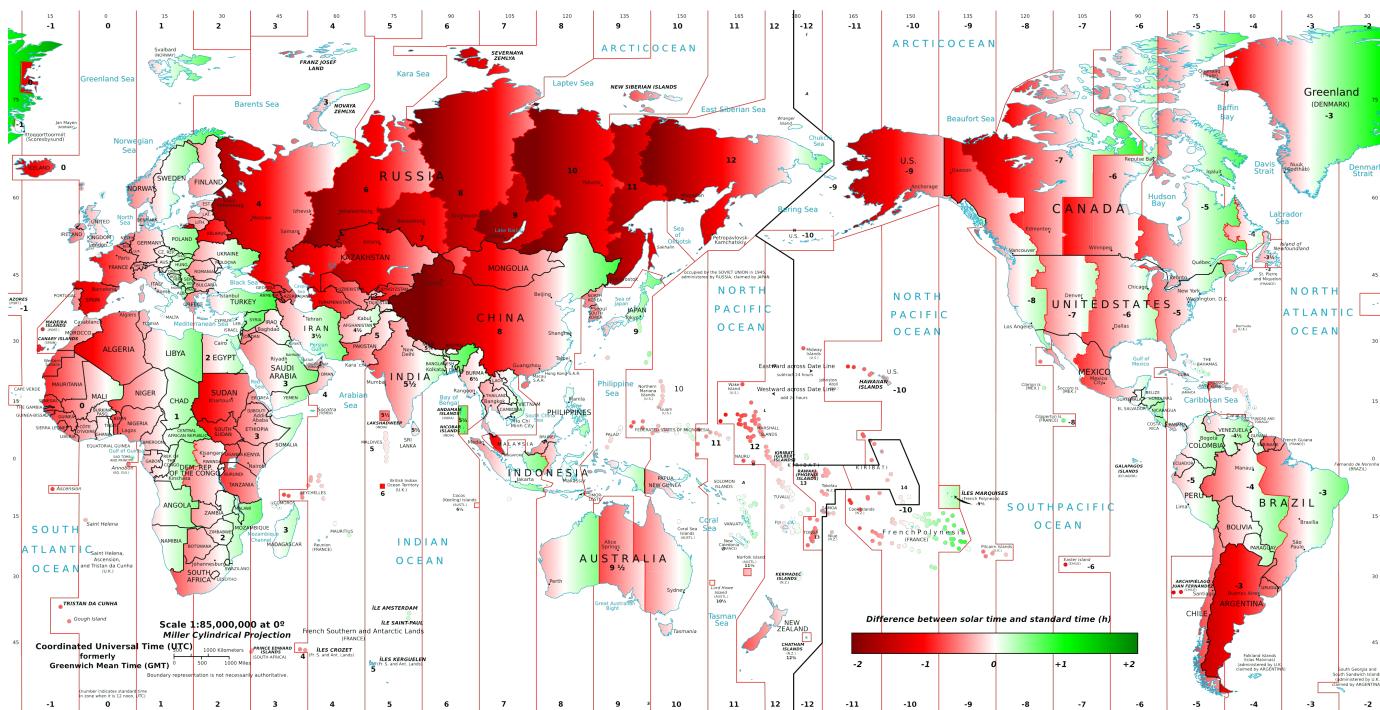
Meridian Utama atau Prime Meridian digunakan untuk menentukan waktu di dunia, metode penentuannya akan dijelaskan sebagai berikut

12.2.2.1 sistem penentuan waktu dunia Menurut sebuah artikel dari Misbah Khusurur dan Jaenal Arifin yang menyebutkan bahwa Waktu Universal (bahasa Inggris Universal Time, disingkat UT) adalah satu ukuran waktu yang didasari oleh rotasi bumi. Satuan ini adalah model perhitungan modern dari GMT (Greenwich Mean Time), yaitu mean waktu matahari di meridian di Greenwich, Inggris, yang biasanya dianggap sebagai bujur geografis 0 derajat GMT ini merupakan waktu 4 pertengahan yang di dasari oleh garis bujur yang melalui Greenwich (BB/BT 0) dan digunakan sebagai standar waktu Dunia Internasional.

Sebelum diperkenalkan nya standar waktu, setiap kota menyetel waktunya sesuai dengan posisi matahari di tempat masing-masing. Sistem ini bekerja dengan baik sampai diperkenalkan nya transportasi kereta api untuk bepergian dengan cepat. akan tetapi, memerlukan seseorang untuk terus-menerus mencocokkan jamnya dengan waktu lokal yang berbeda-beda dari satu kota

ke kota lain. Standar waktu, dimana semua jam di dalam satu daerah menggunakan waktu yang sama, dibuat untuk memecahkan masalah perbedaan waktu seperti dalam perjalanan kereta api di atas.

Standar waktu ini membagi bumi kedalam beberapa bagian ‘zona waktu’, masing-masing bagiannya mencakupi dengan paling sedikit 15 derajat. Semua jam di dalam zona waktu ini disetel sama dengan jam lainnya, tapi berbeda sebanyak satu jam dari jam-jam di zona waktu yang bertetanggaan. Waktu lokal di Royal Greenwich Observatory di Greenwich, Inggris, dipilih sebagai standard waktu dunia setelah terjadi Konferensi Meridian Internasional tahun 1884, yang memicu penyebaran pemakaian Greenwich Mean Time untuk menyetel jam di dalam suatu daerah. Lokasi ini dipilih sampai tahun 1884, 66 % dari semua peta dan tabel menggunakan greenwhich sebagai meridian utama (prime meridian).



Gambar 12.2 menjelaskan tentang zona waktu pada tiap belahan dunia.

Pada gambar 12.2 menjelaskan tentang zona waktu pada tiap belahan dunia. Perbedaan GMT dengan waktu pertengahan setempat di luar Greenwich adalah tergantung besar kecilnya Garis Bujur (BB/BT) dan dapat dirumuskan sebagai berikut[32]:

$$WPx = GMT + BT, \quad (12.1)$$

atau

$$WPx = GMTBB \quad (12.2)$$

$$GMT = WPx - BT, \text{ atau} \quad (12.3)$$

$$GMT = WPx + BB \quad (12.4)$$

Contohnya sebagai berikut:

1. Diketahui BT Semarang = 110 26 Pada saat GMT menunjukkan pukul 11.30,

$$WPx = WP Semarang = 11.30 + 11026 = 11.30 + 7j21m44dt = 18j51m44dt \quad (12.5)$$

2. Diketahui BT Semarang = 100 26 Pada saat WP Semarang menunjukkan pukul 19.54,

$$GMT = 19.54 - 10026 = 19.54 - 8j24m0dt = 11.30 \quad (12.6)$$

Menurut sebuah artikel dari Yanti, Ari Hepi and Dhewiyanty, Varla and Setyawati, Tri Rima yang menyebutkan bahwa daerah yang dilalui garis khatulistiwa memiliki iklim tropis dengan suhu udara cukup tinggi dan kelembaban yang tinggi. Contoh daerah yang dilalui garis khatulistiwa yaitu Kalimantan Barat suhu udara di Kalimantan Barat pada tahun 2013 berkisar antara 21,5oC-34,3oC (BPS Kalbar, 2014). [33]

Menurut sebuah artikel dari Yanti, Ari Hepi and Dhewiyanty, Varla and Setyawati, Tri Rima yang menyebutkan bahwa daerah yang dilalui garis khatulistiwa memiliki iklim tropis dengan suhu udara cukup tinggi dan kelembaban yang tinggi. Contoh daerah yang dilalui garis khatulistiwa yaitu Kalimantan Barat suhu udara di Kalimantan Barat pada tahun 2013 berkisar antara 21,5oC-34,3oC (BPS Kalbar, 2014)[33]. Masih ada lagi beberapa negara yang dilalui oleh garis khatulistiwa yang terdapat pada gambar 12.3.

No.	Nama Negara	Benua
1	Sao Tome dan Pricipe	Afrika
2	Gabon	Afrika
3	Republik Congo	Afrika
4	Republik Demokratik Congo	Afrika
5	Uganda	Afrika
6	Kenya	Afrika
7	Somalia	Afrika
8	Maladewa	Asia
9	Indonesia	Asia
10	Kiribati	Oseania
11	Ekuador	Amerika Selatan
12	Kolombia	Amerika Selatan
13	Brasil	Amerika Selatan

Gambar 12.3 list negara yang dilalui garis khatulistiwa.

Pada gambar 12.3 disebutkan negara - negara yang dilalui oleh garis khatulistiwa yaitu Sao Tome dan Pricipe yang terdapat pada benua Afrika, Gabon yang terdapat di benua Afrika, Republik Congo yang terdapat di benua Afrika, Republik Demokratik Congo yang terdapat di benua Afrika, Uganda yang terdapat di benua Afrika, Kenya yang terdapat di benua Afrika, Somalia yang terdapat di benua Afrika, Maladewa yang terdapat di benua Asia, Indonesia yang terdapat di benua Asia, Negara Kiribati , Ekuador yang terdapat di benua Amerika Selatan, Kolombia yang terdapat di benua Amerika Selatan, dan Brasil yang terdapat di benua Amerika Selatan

untuk lebih detail nya terdapat pada gambar 12.4



Gambar 12.4 wilayah di dunia yang dilewati garis khatulistiwa.

12.2.2.2 Peristiwa Equinox Dalam sebuah artikel dari Mutoha Arkanudin yang menyebutkan bahwa selama setahun Matahari berubah posisi dari Utara ke Selatan dan sebaliknya. Posisi tersebut sering disebut sebagai Gerak Musim Matahari. Equinox adalah saat dimana posisi matahari berada tepat di Ekuator atau garis khatulistiwa. Ini adalah bagian dari siklus tahunan pergerakan harian semu matahari saat terbit, melintas dan terbenam yang disebabkan oleh kemiringan sumbu bumi terhadap bidang orbitnya yaitu sebesar 66,56 derajat. Selama setahun terjadi dua kali Equinox yaitu Maret Equinox yang terjadi setiap tanggal 21 Maret dan September Equinox yang terjadi setiap tanggal 23 September.

Saat terjadi peristiwa Equinox posisi Matahari terbenam akan tepat berada di titik Barat sehingga dengan menambah sudut kemiringan arah kiblat terhadap titik Barat maka arah kiblat yang sesungguhnya akan kita dapatkan.

Selain Equinox matahari juga akan berada di titik paling Utara pada 21 Juni dan berada di titik paling Selatan pada 22 Desember yang dikenal dengan istilah Solstice. Pada saat Juni Solstice, Matahari akan terbenam tepat di sudut serong terhadap arah Barat sebesar 23,5 derajat ke arah Utara sehingga untuk menuju ke arah kiblat yang tepat dapat tinggal menambahkan kekurangan penyerongan angka arah kiblat yang didapatkan dari hasil perhitungan menggunakan rumus segitiga bola. Sedangkan pada saat Desember Solstice matahari terbenam di Selatan titik Barat sebesar 23,5 derajat.[34].

12.3 Penutup

12.3.1 Kesimpulan

Garis khatulistiwa merupakan garis lintang dari 0 derajat sampai dengan 90 derajat di kutub bumi. Prime meridian atau meridian Greenwich adalah nilai koordinat garis bujur dimulai dari bujur 0 derajat yaitu di Greenwich, kemudian membesar ke arah timur dan barat sampai bertemu kembali di Garis batas internasional yaitu terletak di Selat Bering dengan nilai 180 derajat. Sistem koordinat dimaksudkan untuk memberikan pengalaman terhadap setiap lokasi di permukaan bumi. Meridian Utama atau Prime Meridian digunakan untuk menentukan waktu di dunia, metode penentuan mengikuti Waktu Universal (bahasa Inggris Universal Time, disingkat UT) adalah satu ukuran waktu yang didasari oleh rotasi bumi. Daerah yang dilalui garis khatulistiwa memiliki iklim tropis dengan suhu udara cukup tinggi dan kelembaban yang tinggi.

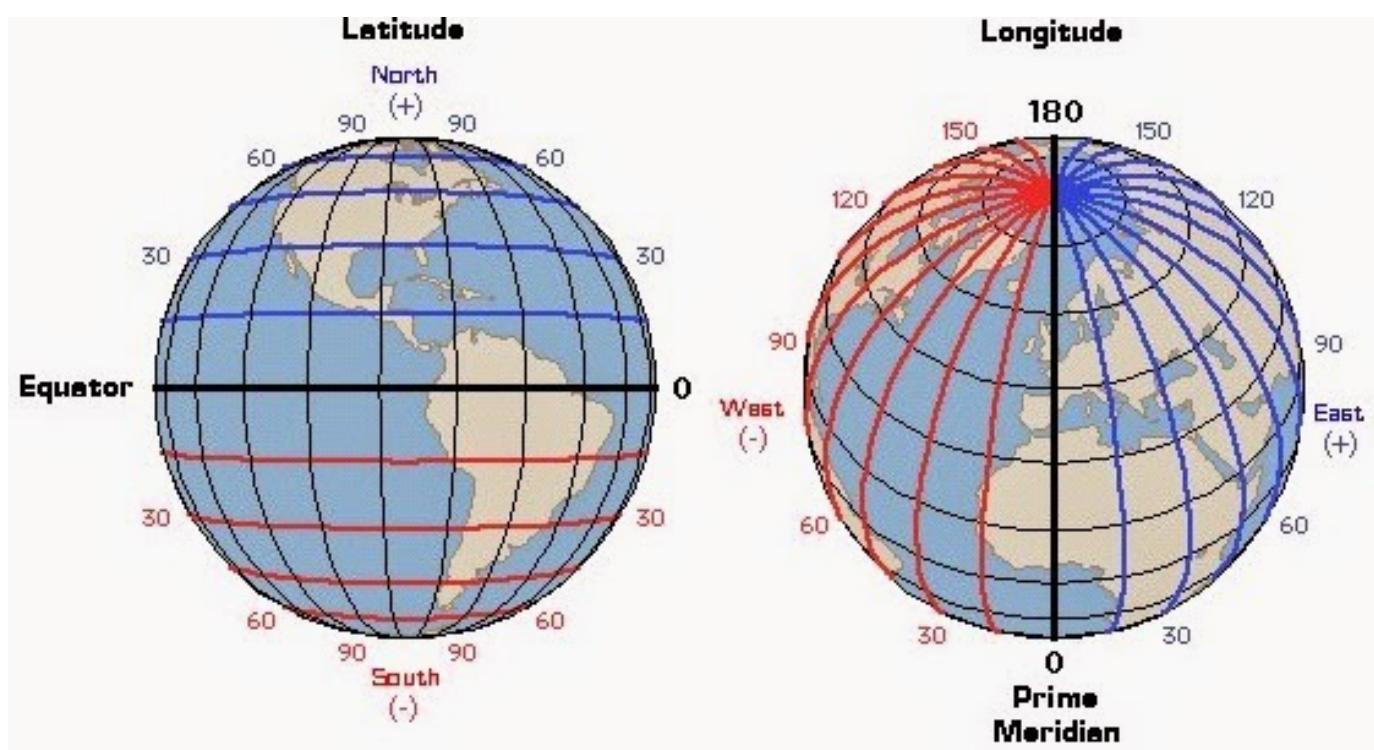
12.3.2 Saran

Dalam artikel ini belum ada penjelasan mengenai sejarah garis khatulistiwa dan prime meridian, maka diharapkan untuk ke depannya dilengkapi dengan informasi mengenai sejarah dari garis khatulistiwa dan prime meridian.

DASAR PEMETAAN KORDINAT INDONESIA

13.1 Koordinat Lintang Utara, Lintang Selatan, Bujur Timur, Bujur Barat

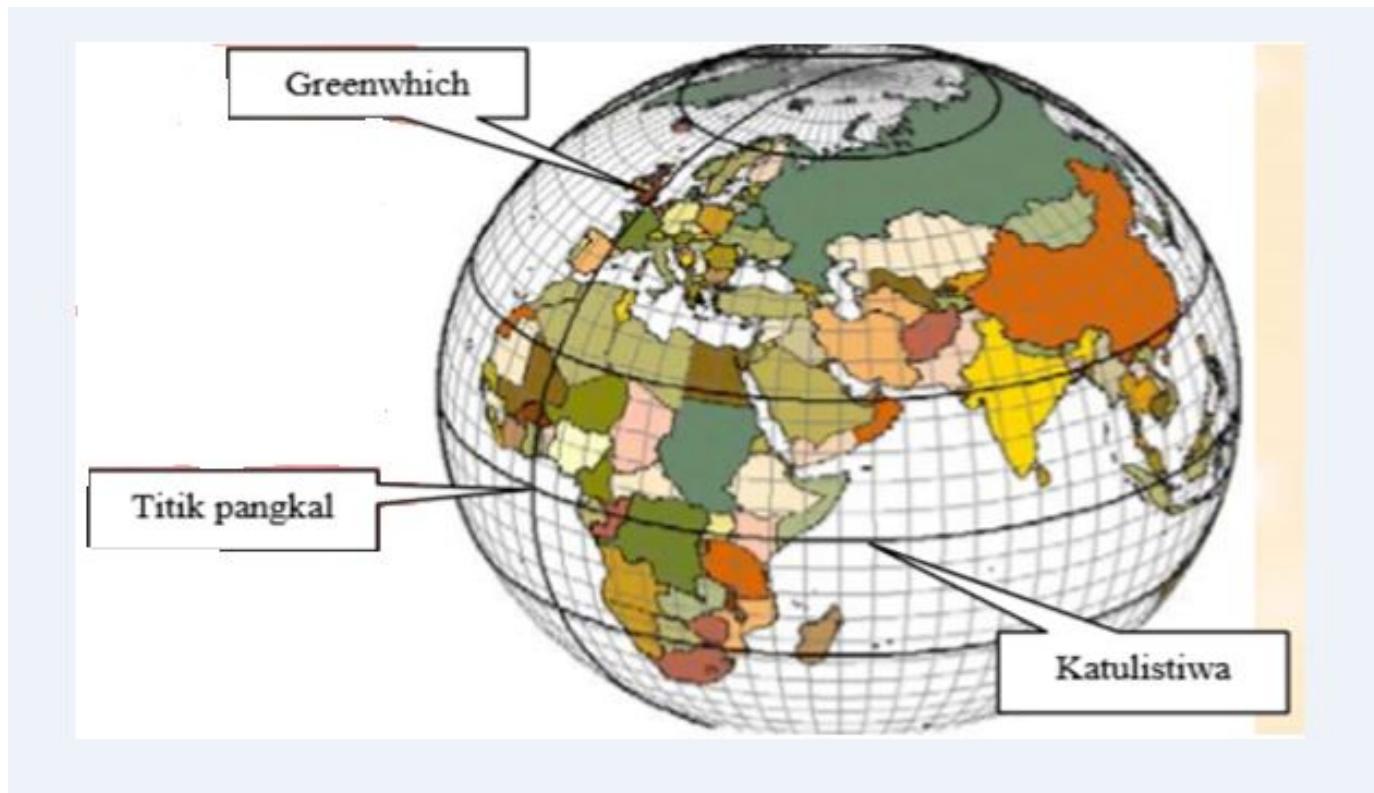
Koordinat digunakan untuk menunjukkan suatu titik di Bumi berdasarkan garis lintang dan garis bujur. Koordinat dibagi menjadi dua bagian irisan yaitu irisan melintang yang disebut dengan garis lintang mulai dari khatulistiwa, membesar ke arah kutub (utara maupun selatan) sedangkan yang lain membujur mulai dari garis Greenwich membesar ke arah barat dan timur. Satuan skala koordinat dibagi dalam derajat lintang 0^* sampai 90^* dan bujur 0^* sampai 180^* . Koordinat ini ditulis dalam satuan derajat, menit, dan detik, misalnya $110^{\circ}35'32''$, dan seterusnya. Untuk membagi dunia dalam wilayah utara dan selatan, maka ditentukan sebuah garis yang tepat berada di tengah, yaitu garis Equator / Khatulistiwa. Untuk membagi wilayah timur dan barat, maka ditentukan sebuah garis Prime meridian yang terletak di kota Greenwich (Inggris), dan perpotongannya bertemu di wilayah laut pasifik, yakni memotong kepulauan Fiji. Koordinat pada gambar 13.1 di jelaskan garis Lintang dan Bujur



Gambar 13.1 Koordinat Lintang dan Bujur

13.1.1 Sistem Koordinat

Dalam artikel Zuhdi menjelaskan Koordinat dimaksudkan untuk memberikan pengalamanan terhadap setiap lokasi di permukaan bumi. Pengalamanan dengan sistem koordinat didasarkan atas jarak timur-barat dan utara-selatan suatu tempat dari suatu titik pangkal tertentu. Jarak diukur dalam satuan derajat sudut yang dibentuk dari titik pangkal ke posisi tersebut melalui pusat bumi. Sedangkan titik pangkal ditetapkan berada di perpotongan belahan utara-selatan bumi (garis khatulistiwa) dengan garis yang membelah bumi timur-barat melalui kota Greenwich di Inggris. Untuk lebih jelas tentang bentuk titik koordinat lihat pada gambar 13.2 di bawah ini :

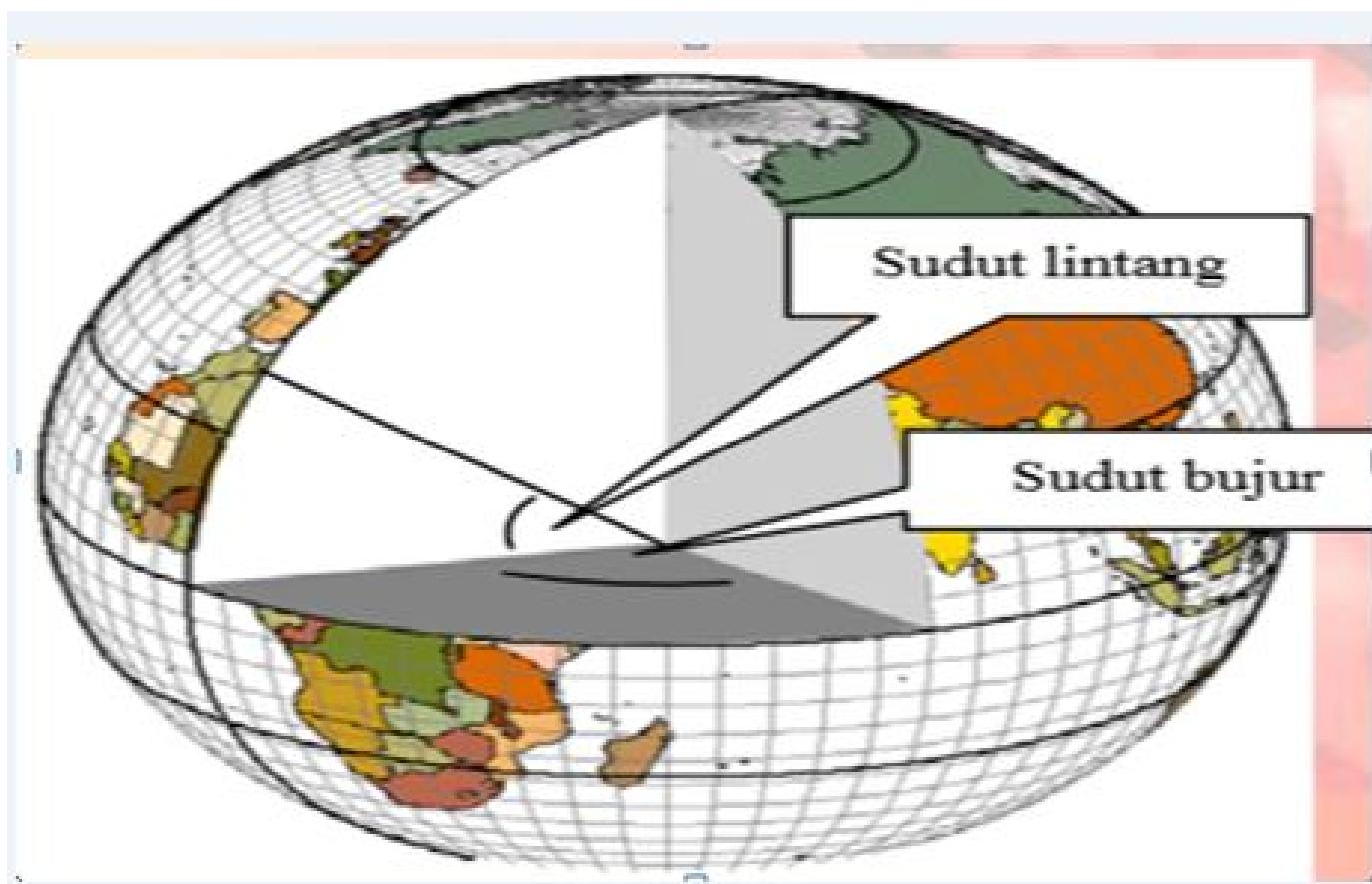


Gambar 13.2 Bentuk titik Koordinat

Baik garis lintang maupun garis bujur diukur dalam derajat dan dibagi lagi dalam menit dan detik. 1 derajat garis bujur diukur lapangan sama dengan 11,32 km. Satuan derajat bisa juga disebut jam sehingga setiap derajat terbagi menjadi 60 menit dan setiap menit terbagi menjadi 60 detik. Dalam penulisan letak astronomis contohnya 60 derajat 23' 15"S, maka dibaca sebagai 60 derajat 23 menit 15 detik lintang selatan. Pada sistem pemetaan internasional huruf U sebagai lintang utara diganti dengan huruf N (north). Besar sudut dalam sistem koordinat geografik dapat dinyatakan dalam dua cara, yaitu dengan satuan DMS (Degree Minute Second) atau satuan DD (Decimal Degree), dalam sistem satuan DMS, setiap derajat sudut dibagi menjadi 60 menit dan setiap menitnya dibagi lagi menjadi 60 detik. Penulisannya dinyatakan sebagai ddmm'ss". Sedangkan pada sistem satuan setiap derajatnya dinyatakan dalam pecahan decimal (pecahan berkomma). Baik dalam DMS maupun DD, perlu diketahui berapa ketelitian suatu nilai koordinat. Karena di wilayah khatulistiwa jarak 1 sama dengan jarak 111321 meter. Maka perlu diperhatikan kesalahan yang terjadi jika kita mengabaikan suatu angka menit atau detik pada DMS atau suatu nilai digit dalam koordinat DD. Pada sistem DD, perlu diperhatikan jarak yang diwakili oleh setiap digit dibelakang koma. Perubahan satu satuan pada digit pertama di belakang koma mempunyai nilai jarak lebih dari 11 Km. Perubahan satu unit pada digit kedua dibelakang koma berarti 1,1 Km. Demikian seterusnya. Berarti jika kita misalnya hanya mentolerir kesalahan sampai 100 m, maka koordinat DD harus dibuat setidaknya sampai 4 digit di belakang koma. Kombinasi antara garis lintang dan garis bujur akan membentuk suatu koordinat lokasi di permukaan bumi dengan sumbu x sebagai garis lintang dan sumbu y sebagai garis bujur dalam koordinat kartesius. Pada Bujur/Longitude (X) merupakan garis yang perpindahannya secara vertical dan pada Lintang/Latitude (Y) merupakan garis yang mempunyai perpindahan secara horizontal. [17].

Lihat pada gambar 13.3 dibawah ini:

13.1.1.1 Garis Lintang Sebuah garis khayal yang digunakan untuk menentukan lokasi di Bumi terhadap garis khatulistiwa (utara atau selatan). Posisi lintang merupakan penghitungan sudut dari 0 derajat di khatulistiwa sampai ke +90 derajat di kutub utara dan -90 derajat di kutub selatan. Dalam bahasa Indonesia lintang di sebelah utara khatulistiwa diberi nama Lintang Utara (LU), demikian pula lintang di sebelah selatan khatulistiwa diberi nama Lintang Selatan (LS). Lintang Utara dan Lintang Selatan menyatakan besarnya sudut antara posisi lintang dengan garis Khatulistiwa. Garis Khatulistiwa sendiri adalah lintang 0 derajat. Nilai koordinat lintang dimulai dari garis lingkaran khatulistiwa yang diberi nilai 0 derajat. Selanjutnya garis lintang yang lain berupa lingkaran paralel (sejajar) khatulistiwa berada disebelah utara dan selatan khatulistiwa. Lingkaran paralel di selatan disebut garis lintang selatan (LS) dan diberi nilai negatif, sedangkan lingkaran paralel di utara diberi nilai positif dan disebut garis lintang utara (LU). Nilai maksimum koordinat garis lintang adalah 90 derajat yaitu terletak di kutub-kutub bumi. Lingkaran paralel merupakan representasi garis lintang ini semakin mengecil ukurannya dengan semakin jauh dari khatulistiwa. Sehingga jarak 1

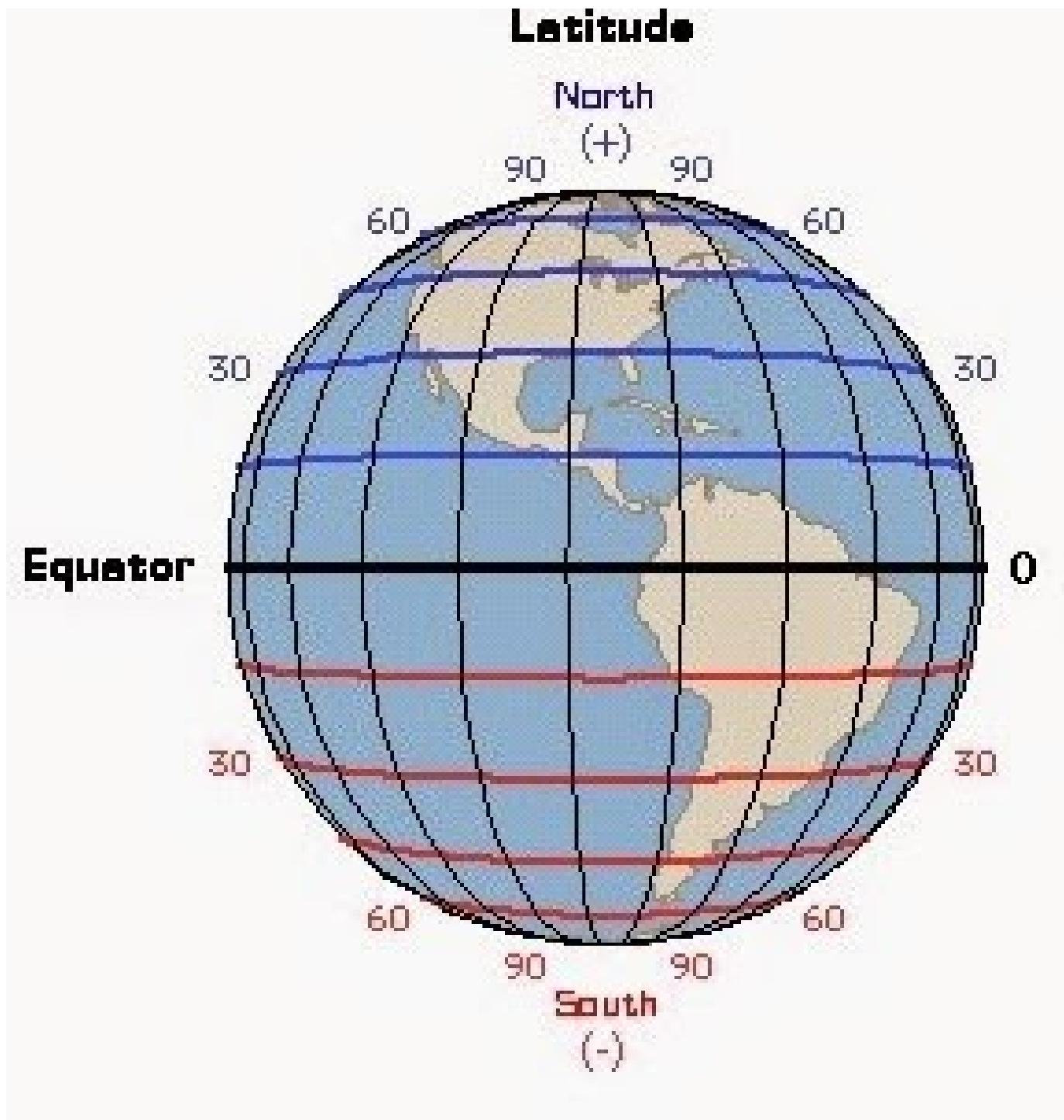


Gambar 13.3 Titik Lintang dan Bujur

derajat timur-barat dari khatulistiwa jauh lebih besar dari pada jarak 1 derajat timur-barat di tempat yang jauh dari khatulistiwa. Di khatulistiwa 1 derajat timur-barat sama dengan 111.321 km, tapi di dekat kutub 1 derajat timur-barat hanya beberapa meter saja. Itu sebabnya grid yang dibuat dari garis lintang dan garis bujur, tampak berupa sangkar di khatulistiwa dan berubah menjadi persegi di daerah kutub lintang memiliki symbol phi dan menunjukkan sudut antara garis lurus di titik tertentu dengan bidang ekuator. Lintang ditentukan dalam angka derajat dimulai dari 0 derajat dan berakhir dengan 90 derajat. Garis lintang ini membagi bumi menjadi belahan bumi utara dan selatan. Garis ekuator atau khatulistiwa berada di lintang 0 derajat. Garis lintang biasa digunakan untuk melihat penyebaran iklim di bumi. Latitude atau garis lintang adalah garis yang menentukan lokasi berada di sebelah utara atau selatan ekuator. Garis lintang diukur mulai dari titik 0 derajat dari khatulistiwa sampai 90 derajat di kutub. Garis lintang digunakan untuk membatasi corak iklim di permukaan bumi, berikut ini merupakan pembagian iklim di bumi menurut batas garis lintang:

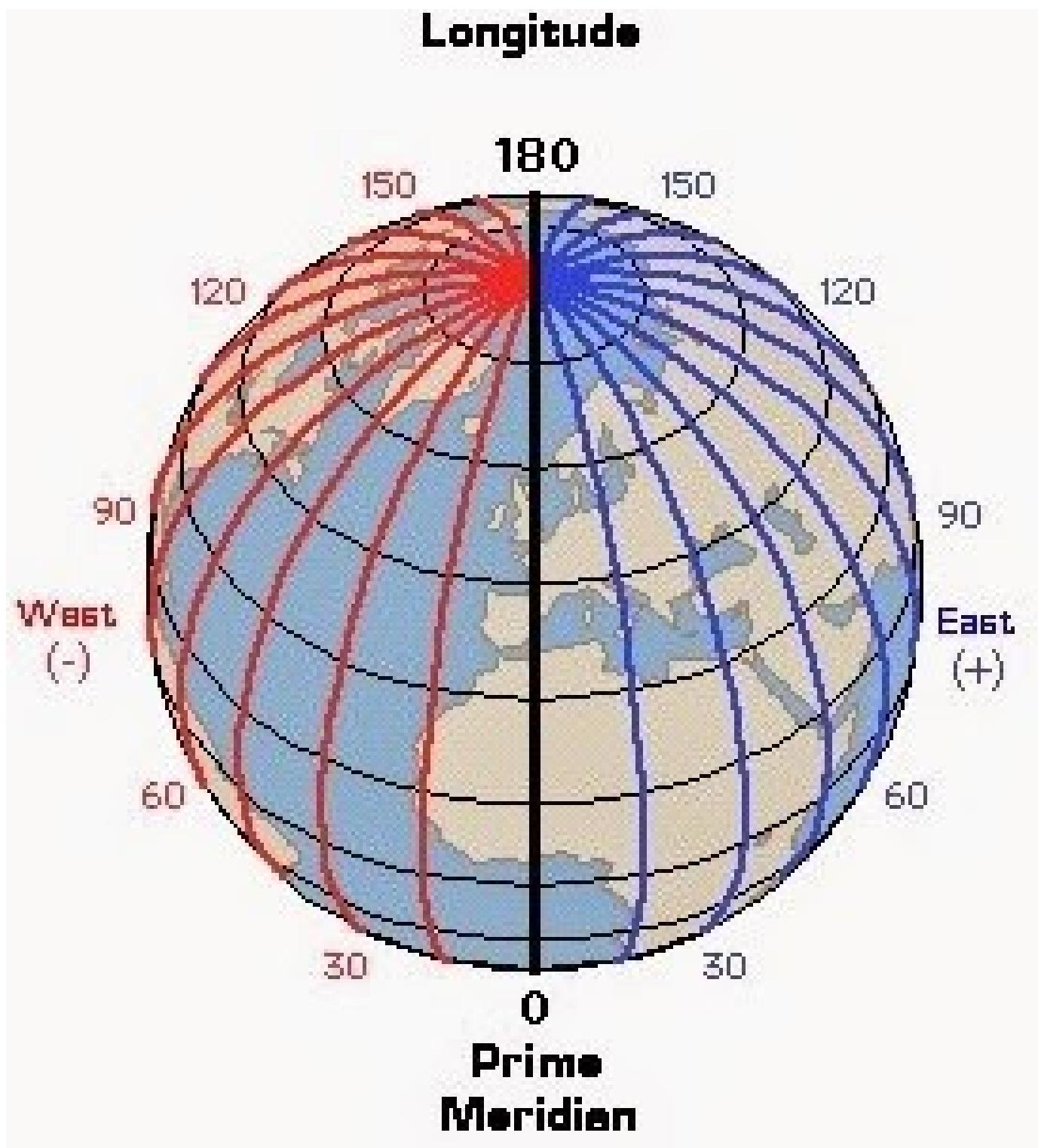
1. 23,5-23,5 LU/LS = iklim tropis 2. 23,5-40 LU/LS = iklim subtropis 3. 40 Lu-66,5 LU/LS = iklim sedang 4. 66,5-90 LU/LS = iklim kutub Indonesia terletak antara 6 derajat Lintang Utara (LU) 11 derajat Lintang Selatan (LS) dan diantara 95 derajat bujur timur 141 derajat Bujur timur. Adapun wilayah indonesia itu pada bagian paling utara yang berada di Pulau Weh di Nanggroe Aceh Darussalam yang terletak pada 6 derajat lintang utara, dan untuk daerah indonesia yang paling berada di selatan yaitu Pulau Roti di Nusa Tenggara Timur yang terletak pada 11 derajat lintang selatan. Kemudian mengacu pada letak lintangnya, di wilayah Indonesia berada pada 6 derajat lintang utara 11 derajat lintang selatan, hal tersebut disebabkan indonesia mempunyai iklim tropis dengan beberapa ciri-ciri yaitu mempunyai hutan hujan tropis yang begitu luas dan mempunyai nilai ekonomis yang sangat tinggi, mendapatkan sinar matahari yang lama setiap sepanjang tahun, mempunyai curah hujan yang tinggi dan memiliki banyak penguapan sehingga akan meningkatkan kelembaban udara. Pada gambar 13.4 dijelaskan titik koordinat Lintang pada sumbu Y:

13.1.1.2 Garis Bujur Menggambarkan lokasi sebuah tempat di timur atau barat Bumi dari sebuah garis utara-selatan yang disebut Meridian Utama. Longitude diberikan berdasarkan pengukuran sudut yang berkisar dari 0 derajat Meridian Utama ke +180 derajat arah timur dan -180 derajat arah barat. Tidak seperti lintang yang memiliki ekuator sebagai posisi awal alami, tidak ada posisi awal alami untuk bujur. Bujur di sebelah barat Meridian diberi nama Bujur Barat (BB), demikian pula bujur di sebelah timur Meridian diberi nama Bujur Timur (BT). Nilai koordinat garis bujur dimulai dari bujur 0 derajat yaitu Greenwich, kemudian membesar ke arah timur dan barat sampai bertemu kembali di garis batas tanggal internasional yaitu terletak di selat bering dengan nilai 180 derajat. Garis bujur 0 derajat disebut prime meridian atau meridian Greenwich. Garis bujur ke arah barat diberi nilai negatif dan disebut bujur barat (west longitude) serta disingkat BB. Sedangkan garis bujur yang ke arah timur diberi nilai positif dan disebut bujur timur (east longitude) disingkat BT. nilai koordinat nya didasarkan atas besarnya sudut yang terbentuk dari bujur 0 ke garis bujur tersebut melalui pusat bumi. Longitude atau garis bujur memiliki symbol lamda. Garis bujur ini merupakan garis yang menunjukkan bagian barat dan timur dilihat dari titik pangkal yaitu di Greenwich meridian. Garis bujur memiliki batas maksimum yaitu 180 derajat ke arah timur darat GMT dan 180 derajat ke arah barat dari GMT. Keduanya bertemu di garis internasional date line di sekitar pasifik. Longitude atau garis bujur digunakan untuk menentukan lokasi di wilayah barat atau timur dari garis utara selatan yang sering disebut juga garis meridian. Garis bujur digunakan untuk menentukan waktu dan tanggal. Titik



Gambar 13.4 Titik koordinat Lintang pada sumbu Y

di barat bujur 0 dinamakan Bujur Barat sedangkan titik di timur 0 dinamakan Bujur Timur. Kombinasi garis lintang dan garis bujur ini berguna untuk menentukan suatu lokasi di permukaan bumi. Garis Lintang menandakan sumbu y dan garis bujur menandakan sumbu x dalam sistem koordinat kartesian. Sebagai contoh kota Sabang di pulau We berada pada koordinat 60°LU 95°BT, dan kota Merauke di Papua mme koordinat 110°LS dan 141°BT. Indonesia berada pada 95 derajat bujur timur 141 derajat bujur timur menyebabkan Indonesia mempunyai tiga waktu dan pada setiap waktu memiliki daerah tersendiri, sehingga Indonesia memiliki beberapa pembagian waktu yaitu Waktu Indonesia bagian timur atau WIT mencakup Papua, kepulauan Maluku dan pulau-pulau kecil di sekitarnya. Untuk waktu Indonesia bagian timur mempunyai selisih waktu sebanyak 9 jam lebih awal dari Greenwich Mean time atau GMT. Kemudian untuk Waktu Indonesia bagian tengah atau WITA mencakup Nusa tenggara, Kalimantan selatan, Pulau Sulawesi, Bali dan pulau-pulau kecil yang ada di sekitarnya. Untuk Indonesia bagian tengah mempunyai selisih waktu sebanyak 8 jam yang lebih awal dari Greenwich mean time (GMT). Kemudian, untuk daerah waktu Indonesia bagian barat atau WIB yang mencakup Madura, Jawa, Kalimantan barat, Kalimantan tengah, Sumatera dan pulau-pulau kecil yang ada di sekitarnya. Adapun waktu Indonesia bagian barat mempunyai selisih waktu sebanyak 7 jam yang lebih awal dari Greenwich mean time. Pada gambar 13.5 dijelaskan titik koordinat Lintang pada sumbu X:



Gambar 13.5 Titik koordinat Bujur pada sumbu X

DASAR PEMETAAN KORDINAT INTERNASIONAL

14.1 latitude longitude

14.1.1 Latitude

Latitude merupakan terjemahan bahasa inggris dari garis lintang. Garis lintang dapat disebut juga sebagai garis khatulistiwa (0 derajat), atau bisa disebut juga sebagai garis tengah bumi yang membagi antara belahan bumi bagian atas dan bumi bagian bawah.

Dalam sebuah buku karangan Maling & Derek Hylton yang berjudul ‘Coordinate System and Map Projections’ mengatakan bahwa garis lintang suatu titik dapat didefinisikan secara formal sebagai sudut yang diukur di tengah bumi di antara bidang equator dan jari-jari yang ditarik ke titik.

Pada garis lintang bagian utara bumi dilambangkan dengan tanda ‘+phi’ sedangkan garis lintang bagian selatan bumi dilambangkan dengan tanda ‘-phi’ [35].

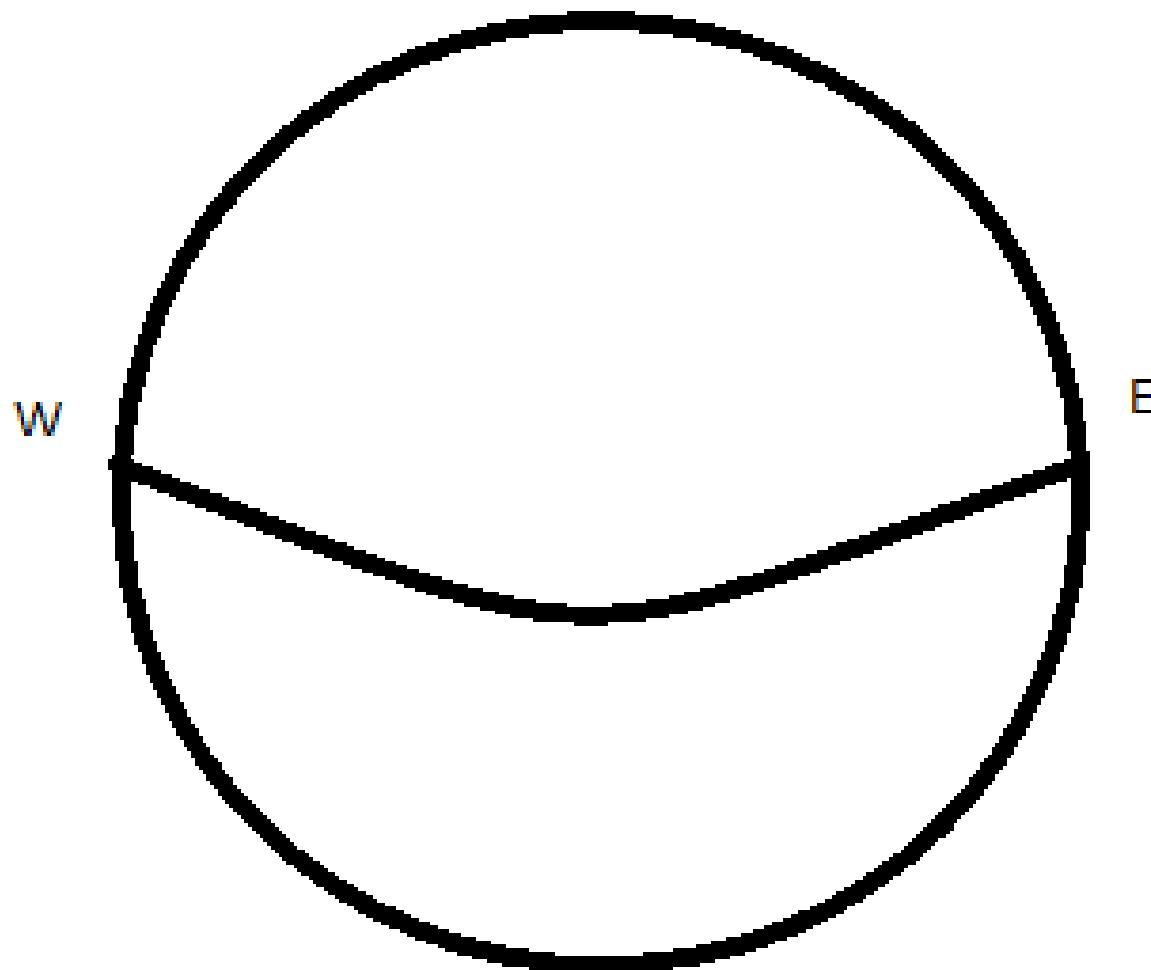
Pada gambar 14.1 merupakan gambar latitude atau garis lintang yang membentang antara west (barat) sampai east (timur). Garis lintang digunakan sebagai penanda dalam zona iklim di dunia. Dari +23 setengah derajat Lintang Utara sampai -23 setengah Lintang Selatan memiliki zona iklim tropis. Zona iklim tropis hanya memiliki dua musim, yaitu kemarau atau panas dan penghujan saja. Kemudian dari +23 setengah derajat Lintang utara sampai +66 setengah derajat Lintang utara memiliki zona iklim subtropis. Sama halnya bagian utara, bagian selatan yaitu -23 setengah derajat lintang selatan sampai -66 setengah derajat lintang selatan memiliki zona iklim subtropis. Daerah subtropis memiliki 4 musim, yaitu spring, summer, fall, dan winter.

14.1.2 Longitude

Longitude merupakan terjemahan bahasa Inggris dari garis bujur. Garis bujur biasa digunakan untuk menentukan waktu dan tanggal di dunia yang kita huni sekarang ini. Jika garis lintang atau latitude atau daerah khatulistiwa dianggap sebagai 0 derajat, maka garis bujur merupakan 0 derajat yang menghubungkan kutub utara dengan kutub selatan yang melawati kota Greenwich di Inggris. Garis bujur bagian barat kota Greenwich disebut sebagai Bujur Barat sedangkan garis bujur yang berada pada sebelah timur kota Greenwich disebut sebagai Bujur Timur. Inilah penyebab kenapa orang Indonesia disebut sebagai orang timur. Pada gambar 14.2 merupakan gambar longitude atau garis bujur yang menghubungkan kutub utara dengan kutub selatan. Garis ini melewati kota Greenwich di Inggris. Garis bujur digunakan untuk pembagian zona waktu di dunia.

14.2 LINTANG

Sudut lintang 1 Bayangan Bumi adalah bola transparan (sebenarnya bentuknya agak oval; karena rotasi bumi, nya Khatulistiwa sedikit menonjol). Melalui Bumi yang transparan (gambar) kita bisa melihat bidang ekuator nya, dan bagian tengahnya titik adalah O, pusat bumi. Untuk menentukan garis lintang beberapa titik P di permukaan, tariklah radius OP ke titik itu. Maka sudut elevasi titik itu di atas garis ekuator adalah garis lintang 1 - lintang utara jika utara dari garis khatulistiwa, lintang selatan (atau negatif) jika selatan nya. Garis lintang. Di dunia bumi, garis lintang adalah lingkaran dengan ukuran yang berbeda. Itu terpanjang adalah khatulistiwa, yang garis lintangnya nol, sementara di kutub - di garis lintang 90° utara dan 90° selatan (atau -90°) lingkaran menyusut ke titik tertentu.



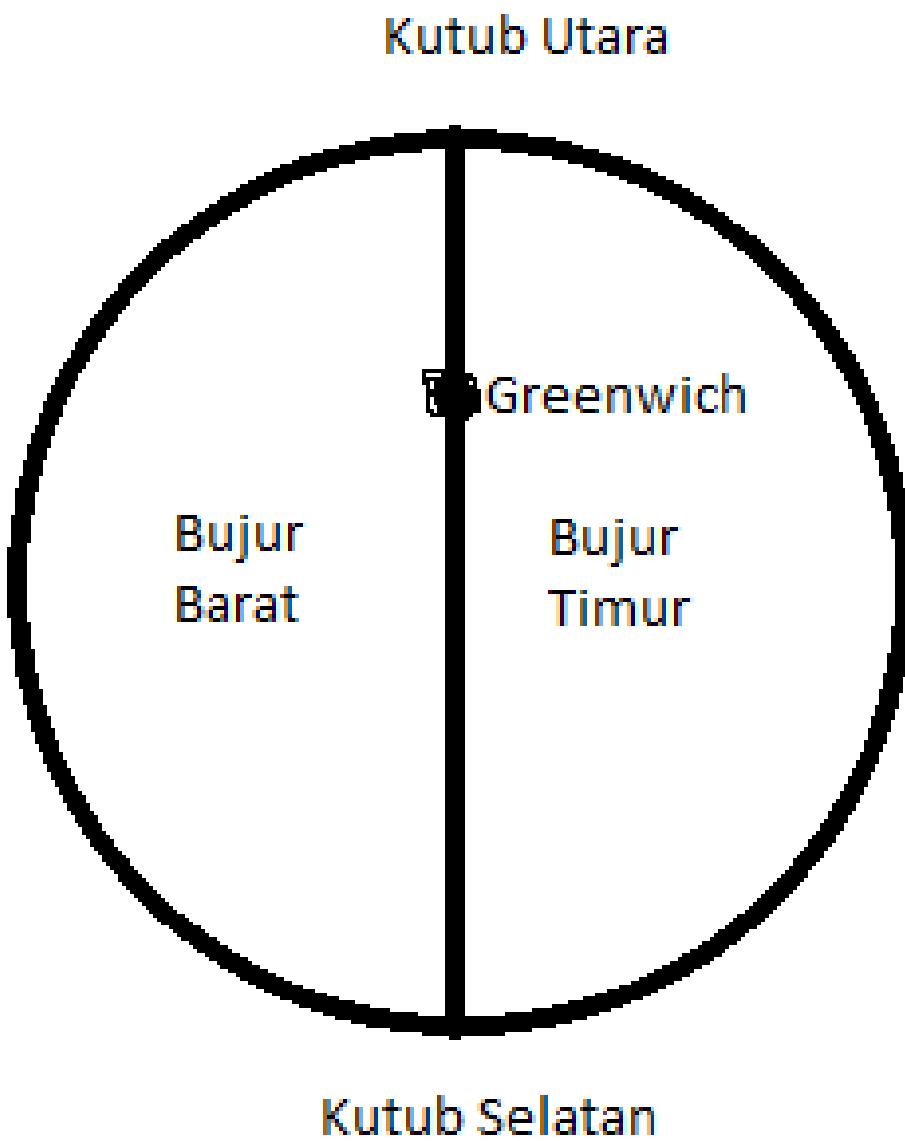
Gambar 14.1 Garis Lintang atau Latitude.

14.3 GARIS BUJUR

Di dunia, garis bujur konstan ("meridian") meluas dari tiang ke kutub, seperti batas segmen pada jeruk kupas. Garis bujur atau "garis meridian" Setiap meridian harus melewati garis khatulistiwa. Karena ekuator adalah lingkaran, kita bisa bagilah itu - seperti lingkaran - ke dalam 360 derajat, dan bujur f dari sebuah titik adalah maka nilai yang ditandai dari divisi mana meridinya memenuhi khatulistiwa. Apa nilai itu tergantung tentu saja dari mana kita mulai menghitung di mana nol bujur adalah Untuk alasan historis, garis meridian melewati Astronomi Kerajaan yang lama Observatorium di Greenwich, Inggris, adalah yang dipilih sebagai nol bujur. Bertempat di Jl Tepi timur London, ibu kota Inggris, observatorium sekarang menjadi museum umum dan a band kuningan yang membentang di halamannya menandai "garis meridian utama". Wisatawan sering mendapatkan difoto saat mereka mengangkang nya satu kaki di belahan bumi bagian timur, yang lainnya masuk belahan barat. Garis bujur juga disebut meridian, berasal dari bahasa Latin, dari meri, a variasi "medius" yang menunjukkan "tengah", dan diem, yang berarti "hari". Kata itu pernah berarti "siang", dan waktu sehari sebelum siang hari dikenal sebagai "ante meridian", sementara waktu setelah itu adalah "posting meridian." Singkatan hari ini a.m. dan p.m. datang Dari istilah ini, dan Matahari pada siang hari dikatakan "melewati meridian". Semua poin di garis bujur yang sama mengalami siang hari (dan jam lainnya) pada saat bersamaan dan oleh karena itu dikatakan sama "garis meridian", yang menjadi "meridian" untuk pendek.

14.4 Waktu Lokal (LT) dan Zona Waktu

Garis bujur diukur dari nol sampai 180 BT dan 180 BB (atau -180), dan kedua 180- Gelombang longitudinal berbagi jalur yang sama, di tengah Samudera Pasifik. Saat Bumi berputar mengelilingi porosnya, kapan pun satu garis bujur - "siang hari meridian "- menghadap Matahari, dan pada saat itu, akan ada siang hari di mana-mana di atasnya jam Bumi telah mengalami rotasi penuh sehubungan dengan Matahari, dan meridian yang sama lagi wajah siang hari Jadi setiap jam Bumi berputar $360/24 = 15$ derajat. Bila di lokasi Anda waktu 12 siang, 15 ke timur waktu adalah 1 p.m., karena itu adalah meridian yang dihadapi Matahari sejam



Gambar 14.2 Garis Bujur atau Longitude.

yang lalu. Di sisi lain, 15° ke barat waktu adalah 11 a.m., untuk satu jam lagi, meridian itu akan menghadapi Matahari dan mengalami siang hari.

14.4.1 Glosarium

Khatulistiwa-Garis yang mengelilingi Bumi pada jarak yang sama dari Utara dan Selatan Polandia Koordinat geografis - Koordinat nilai yang diberikan sebagai garis lintang dan bujur. Lingkaran besar - Sebuah lingkaran terbentuk di permukaan bola oleh sebuah pesawat yang melewati pusat bola. Khatulistiwa, masing-masing meridian, dan satu sama lain keliling penuh Bumi membentuk lingkaran besar. Arus lingkaran besar menunjukkan jarak terpendek antara titik-titik di permukaan bumi.

14.4.1.1 Meridian Lingkaran besar di permukaan Bumi, melewati kutub geografis dan beberapa titik ketiga di permukaan bumi. Semua poin pada meridian tertentu memiliki hal yang sama

14.4.1.2 Paralel Lingkaran atau perkiraan lingkaran di permukaan Bumi, sejajar dengan Khatulistiwa dan titik penghubung dengan garis lintang yang sama.

14.4.1.3 Prime Meridian Garis meridian bujur 0 derajat, digunakan sebagai asal untuk pengukuran bujur. Garis meridian Greenwich, Inggris, adalah internasional menerima meridian utama dalam banyak kasus.

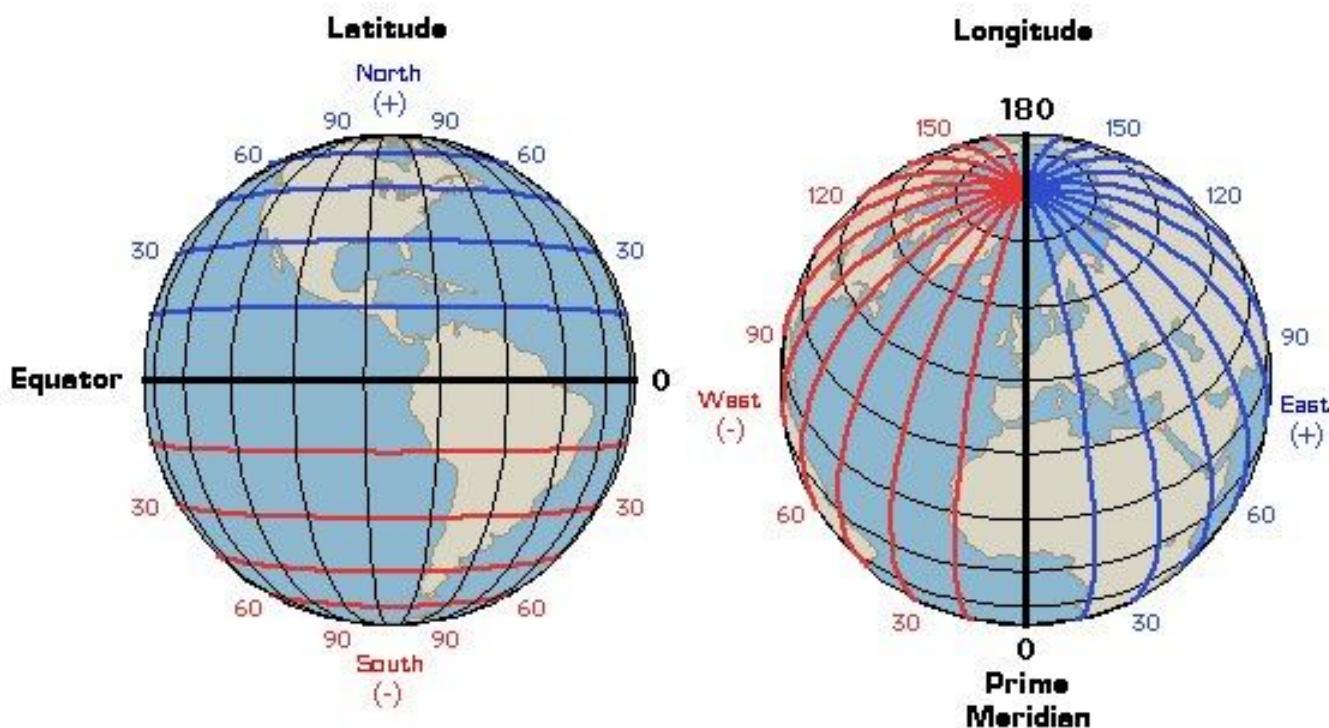
14.5 Konversi antara koordinat geografis dan Kartesian koordinat

Asumsikan bahwa koordinat geografis dari suatu titik M adalah l dan f ; asumsikan bahwa jari - jari Bumi adalah R . Masalahnya adalah penentuan koordinat kartesius M dalam a Sistem koordinat asal pusat bumi, dengan bidang horizontal bidang Khatulistiwa, dengan sumbu x melewati meridian Greenwich, sumbu y secara langsung tegak lurus dengan sumbu x, dan akhirnya sumbu z melewati kutub. Tujuannya adalah untuk menemukan x , y dan z .

Tunjukkan pada gambar sudut l dan f ; Berapakah jarak OM? Hitung jarak OH menurut l . Berapakah nilai x dan y menurut l dan f ; Berapakah nilai z ? Asumsikan bahwa koordinat geografis dari suatu titik V adalah: garis lintang: $45^{\circ} 41' 47.59''$ N Bujur: $4^{\circ} 52' + 49.98'$ E Apa koordinat kartesian V (dengan $R = 1$) Sebenarnya, titik ini persis sekolah kita!

14.6 LINTANG/LATITUDE

Latitude adalah garis mendatar. Titik 0 adalah sudut ekuator tanda + menunjukkan arah ke atas menuju kutub utara, sementara tanda minus di koordinat menuju ke kutub selatan. Bayangkan bila bumi hanyalah sebuah bola transparan (sebenarnya bentuk bumi adalah oval; ini dikarenakan rotasi bumi itu sendiri, karena garis khatulistiwa sedikit terlihat). Dengan bumi yang transparan, kita bisa lihat (gambar) garis khatulistiwa bumi, dan garis tengahnya adalah 0, pusat bumi. Untuk menentukan latitude (garis



Gambar 14.3 gambar Latitude.

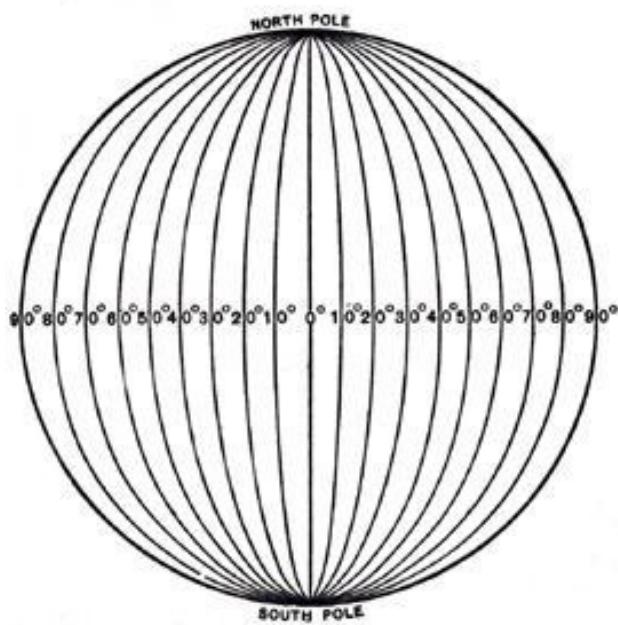
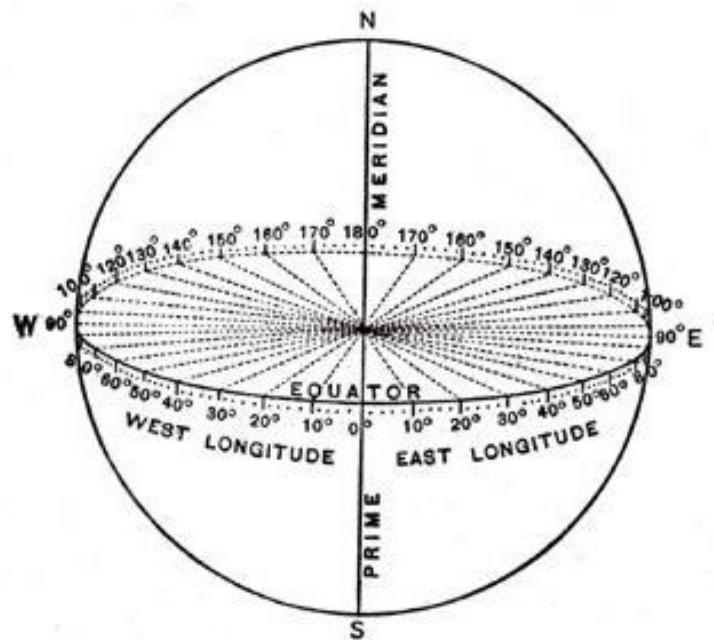
lintang) di beberapa titik P di permukaan, buatlah suatu jarak OP ke suatu titik. Lalu sudut elevasi titik tersebut berada diatas garis ekuator adalah garis lintang 1 - lintang utara jika dari utara, lintang selatan (negatif) jika dari selatan. Garis Lintang, dalam bola bumi, garis lintang dalam lingkaran memiliki perbedaan ukuran. Garis paling panjang adalah Khatulistiwa, dimana yang lintangnya 0 (nol), sementara di daerah kutub, garis lintangnya 90 utara dan 90 selatan (atau bisa juga -90) lingkarannya menyusut ke titik tertentu.

14.7 BUJUR/LONGITUDE

Longitude adalah garis bujur, dimana garis bujur ini diawali dari titik 0 sampai 180 ke arah sebaliknya. Titik 0 dimulai dari garis negara Inggris, mengarah ke Indonesia akan menjadi angka positif. Jika koordinat longitude (lintang) akan menjadi minus ke arah kebalikan. Di bola bumi, garis bujur konstan meluas dari kutub ke kutub seperti batas segmen pada jeruk kupas. Garis Bujur atau Meridian (gambar)

Setiap meridian harus menyebrangi garis khatulistiwa. Karena khatulistiwa adalah sebuah lingkaran, kita bisa membaginya seperti lingkaran yang lain ke dalam 360, dan garis bujur f dari sebuah titik yang ditandai dimana meridian bertemu khatulistiwa.

Nilai tersebut tentu bergantung pada saat kita mulai menghitung titik 0 garis bujur. Untuk alasan sejarah, garis bujur melewati Old Royal - Astronomical Observatory di Greenwich, Inggris, dimana garis 0 bujur di tetapkan. Berlokasi di tepi timur Inggris, ibukota Inggris, Observatorium sekarang adalah Museum Umum dan suatu tanda yang membentang diatas halamannya yang menandai sebagai "garis meridian utama". Garis bujur atau dengan nama lain meridian, berasal dari bahasa latin, yaitu meri, variasi dari "merius" yang berarti "tengah" dan diem yang berarti "hari". Kata tersebut juga bisa berarti "sore", dan waktu pada satu hari sebelum sore kita sebut sebagai "ante meridian" dimana waktu setelahnya berarti "post meridian". Pada saat ini disingkat

**Meridians of longitude****Degrees of longitude****Gambar 14.4** gambar longitude.

menjadi a.m. dan p.m. yang berasal dari istilah ini, dan matahari pada saat menjelang malam hari disebut sebagai "passing meridians". Semua titik pada setiap garis yang sama dalam garis bujur disebut sore (dan pada jam lainnya) pada saat yang sama dan oleh karena itu disebut "garis meridian", yang menjadi "meridian" untuk lebih singkat.

Bagian III

DATA GEOSPASIAL TIPE DATA

DATA GEOSPASIAL DATA RASTER

15.1 Data Raster

15.1.1 Pengertian Data Raster

Data raster [36] adalah data yang disimpan dalam bentuk persegi empat sama sisi (grid) sel sehingga terbentuk suatu ruang yang teratur. Foto digital seperti areal fotografi atau satelit merupakan bagian dari data raster pada peta. Raster memiliki data grid continue. Nilainya menggunakan gambar berwarna seperti fotografi, yang ditampilkan dengan level merah, hijau, dan biru pada sel. Data Raster (atau disebut juga dengan sel grid) merupakan data yang dihasilkan dari sistem penginderaan jauh. Pada data raster. Obyek geografis direpresentasikan sebagai struktur sel grid yang disebut dengan pixel (picture element). Pada data raster. Resolusi (definisi visual) tergantung pada ukuran pixelnya. Dengan kata lain. Resolusi pixel menggambarkan ukuran sebenarnya diperlukan bumi yang diwakili oleh setiap pixel pada citra. Pada data raster, Obyek geografis direpresentasikan sebagai struktur sel grid yang disebut sebagai pixel (picture element). Resolusi (definisi visual) tergantung pada ukuran pixelnya, semakin kecil ukuran permukaan bumi yang direpresentasikan oleh sel, semakin tinggi resolusinya. Data Raster dihasilkan dari sistem penginderaan jauh dan sangat baik untuk merepresentasikan batas-batas yang berubah secara gradual seperti jenis tanah, kelembaban tanah, suhu, dan lain-lain. Peta raster adalah peta yang diperoleh dari fotografi suatu areal. foto satelit atau foto permukaan bumi yang diperoleh dari komputer. Contoh peta raster yang diambil dari satelit cuaca. Di dalam Sig, data raster dan analisis data raster banyak digunakan untuk pemetaan obyek yang bersifat kontinu (batasnya tidak terlihat jelas di lapangan/gradual) dan pemodelan spasial, baik statis maupun dinamis. Analisis data raster banyak menggunakan peta-peta hasil analisa digital citra satelit karena peta raster dan citra satelit mempunyai struktur data yang sama, yaitu grid cell, sehingga kompatibel satu dengan yang lain. Hal ini berbeda dengan data vector, dimana agar bisa dianalisis secara bersama, Data raster hasil analisis harus dikonversi dulu ke struktur data vektor [36].

15.1.2 Pengertian Data Vektor

Data vektor adalah data yang direkam dalam bentuk koordinat titik yang menampilkan, menempatkan dan menyimpan data spasial dengan menggunakan titik, garis atau area (polygon). Ada tiga tipe data vektor (titik, garis, dan polygon) yang bisa digunakan untuk menampilkan informasi pada peta. Titik bisa digunakan sebagai lokasi sebuah kota atau posisi tower radio. Garis bisa digunakan untuk menunjukkan rute suatu perjalanan atau menggambarkan boundary. Poligon bisa digunakan untuk menggambarkan sebuah danau atau sebuah Negara pada peta dunia.

15.1.3 Kelebihan dan kekurangan Data Raster

15.1.3.1 Kelebihan Data Raster

Adapun kelebihan yang dimiliki oleh data raster menurut [37] adalah:

1. memiliki struktur data yang sederhana
2. mudah dimanipulasi dengan menggunakan fungsi-fungsi matematis sederhana
3. teknologi yang digunakan cukup murah dan tidak begitu kompleks sehingga pengguna dapat membuat sendiri program aplikasi yang menggunakan citra raster

4. compatible dengan citra-citra satelit penginderaan jauh dan semua image hasil scanning data spasial.
5. Overlay dan kombinasi data raster dengan data inderaja mudah dilakukan.
6. memiliki kemampuan-kemampuan pemodelan dan analisis spasial tingkat lanjut.
7. metode untuk mendapatkan citra raster lebih mudah
8. Gambar permukaan bumi dalam bentuk citra raster yang didapat dari radar atau satelit penginderaan jauh selalu aktual dari pada bentuk vektornya
9. prosedur untuk memperoleh data dalam bentuk raster lebih mudah, sederhana dan murah.
10. Harga sistem perangkat lunak aplikasinya cenderung lebih murah.

15.1.3.2 Kekurangan Data Raster Adapun Kekurangan yang dimiliki oleh data raster menurut [37] adalah :

1. secara umum memerlukan ruang atau tempat penyimpanan (disk) yang besar dalam komputer, banyak terjadi redundancy data baik untuk setiap layer-nya maupun secara keseluruhan.
2. Penggunaan sel atau ukuran grid yang lebih besar untuk menghemat ruang penyimpanan akan menyebabkan kehilangan informasi dan ketelitian.
3. sebuah citra raster hanya mengandung satu tematik saja sehingga sulit digabungkan dengan atribut-atribut lainnya dalam satu layer.
4. tampilan atau representasi dan akurasi posisi sangat bergantung pada ukuran pikselnya (resolusi spasial)
5. sering mengalami kesalahan dalam menggambarkan bentuk dan garis batas suatu obyek. sangat bergantung pada resolusi spasial dan toleransi yang diberikan
6. transformasi koordinat dan proyeksi lebih sulit dilakukan
7. sangat sulit untuk mewujudkan hubungan topologi (juga network)
8. metode untuk mendapatkan format data vektor melalui proses yang lama, cukup melelahkan dan relatif mahal.

15.1.4 perbedaan data raster dan data vektor

Masing-masing format data mempunyai kelebihan dan kekurangan. Pemilihan format data yang digunakan sangat tergantung pada tujuan penggunaan. Data yang tersedia, volume data yang dihasilkan, ketelitian yang diinginkan, serta kemudahan dalam Analisa.

Data Vektor relatif lebih ekonomis dalam hal ukuran file dan presisi dalam lokasi. Tetapi sangat sulit untuk digunakan dalam komputasi matematik. Sebaliknya Data raster biasanya membutuhkan ruang penyimpanan file yang lebih besar dan presisi lokasinya lebih rendah. Tetapi lebih mudah digunakan secara matematis. Model data raster mempunyai struktur data yang tersusun dalam bentuk matriks atau pixel dan membentuk grid. Setiap pixel memiliki nilai tertentu dan memiliki atribut tersendiri, termasuk nilai kordinat yang unik.

Tingkat keakurasaian model ini sangat tergantung pada ukuran piksel atau biasa disebut resolusi. Model data ini biasanya digunakan dalam remote sensing yang berbasiskan citra satelit maupun airborne (pesawat terbang). Selain itu model ini digunakan pula dalam membangun model ketinggian digital(DEM-Digital Elevation Model) dan model permukaan digital(DTM-Digital Terrain Model). Model Raster Memberikan Informasi spasial terhadap permukaan di bumi dalam bentuk gambaran yang digeneralisasi. Representasi dunia nyata disajikan sebagai elemen matriks atau piksel yang membentuk grid yang homogen. Pada setiap piksel mewakili setiap obyek yang terekam dan ditandai dengan nilai-nilai tertentu. Secara konseptual model data raster merupakan model data spasial yang paling sederhana.

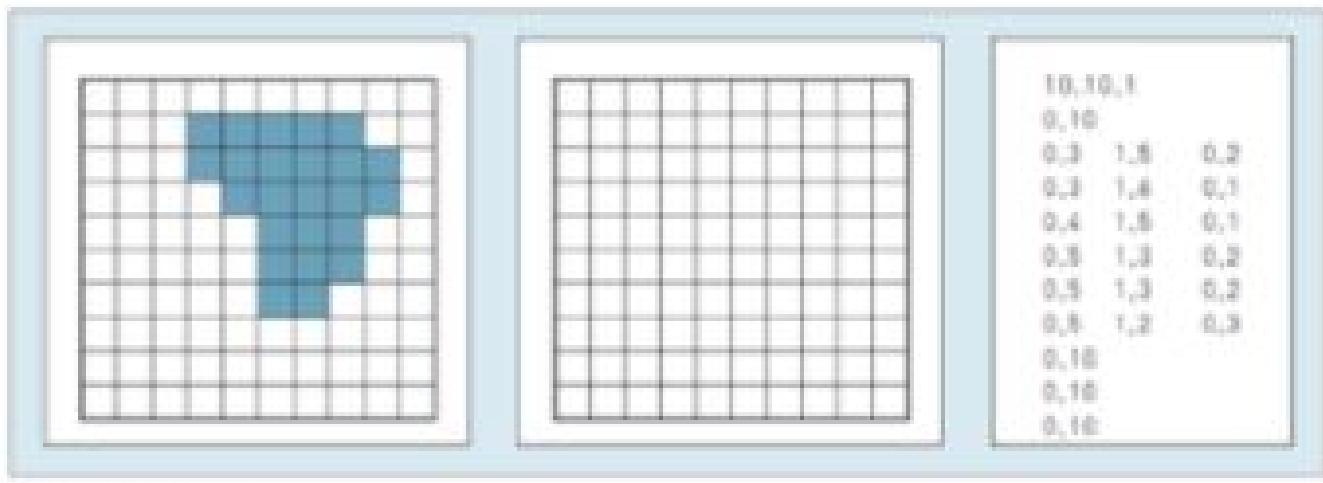
15.1.5 karakteristik data raster

Resolusi suatu data raster akan merujuk pada ukuran permukaan bumi yang direpresentasikan oleh setiap piksel. Makin kecil ukuran atau luas permukaan bumi yang dapat direpresentasikan oleh setiap pikselnya, makin tinggi resolusi spasialnya. Piksel-piksel di dalam zone atau area yang sejenis memiliki nilai (isi piksel atau ID number) yang sama. Pada umumnya, lokasi di dalam model data raster, diidentifikasi dengan menggunakan pasangan koordinat kolom dan baris (x,y).

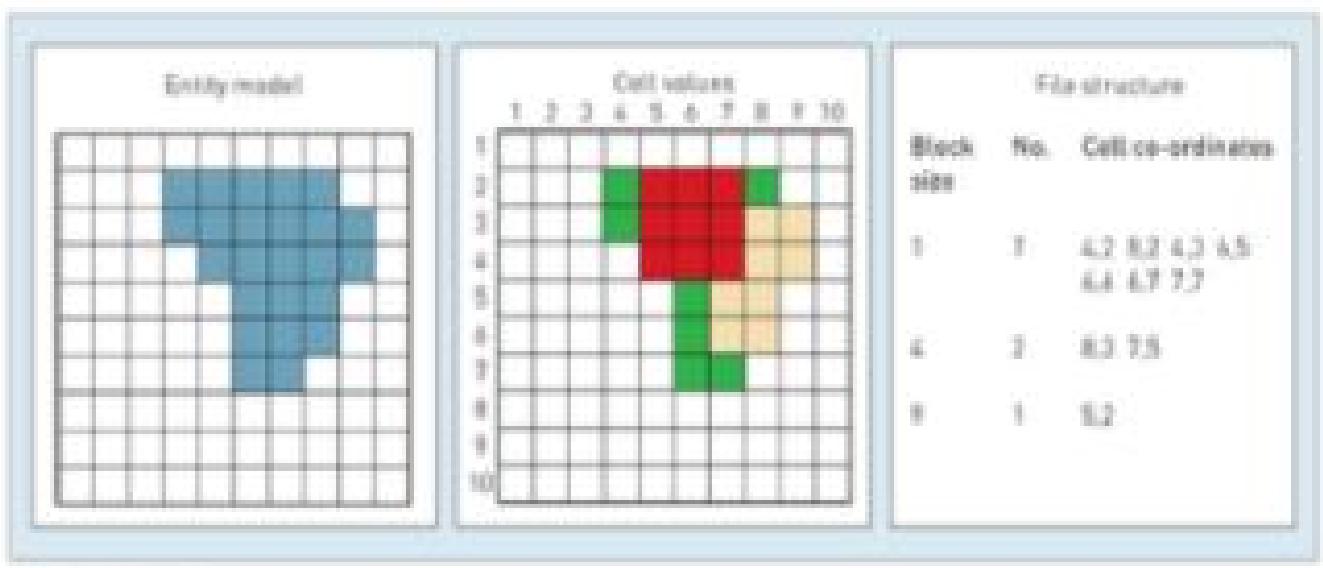
15.1.6 Metode Penyimpanan Data Raster

Data raster mempunyai beberapa metode penyimpanan data yaitu run length encoding, block encoding, chain encoding dan quadtree data structure. Dibawah ini terdapat beberapa contoh gambar metode penyimpanan data raster diantaranya :

- Run Length Encoding(RLE)(Gambar 15.1). Dengan mengurangi jumlah data pada setiap baris. Format RLE, memberikan kelebihan berupa jumlah byte citra yang dapat dimanfaatkan tanpa mengurangi kandungan informasinya. Prinsip penyimpanannya ialah dengan mengekspresikan kembali jumlah piksel yang berurutan dengan nilai yang sama sebagai satu pasangan nilai.

**Gambar 15.1** Gambar Run Length Encoding

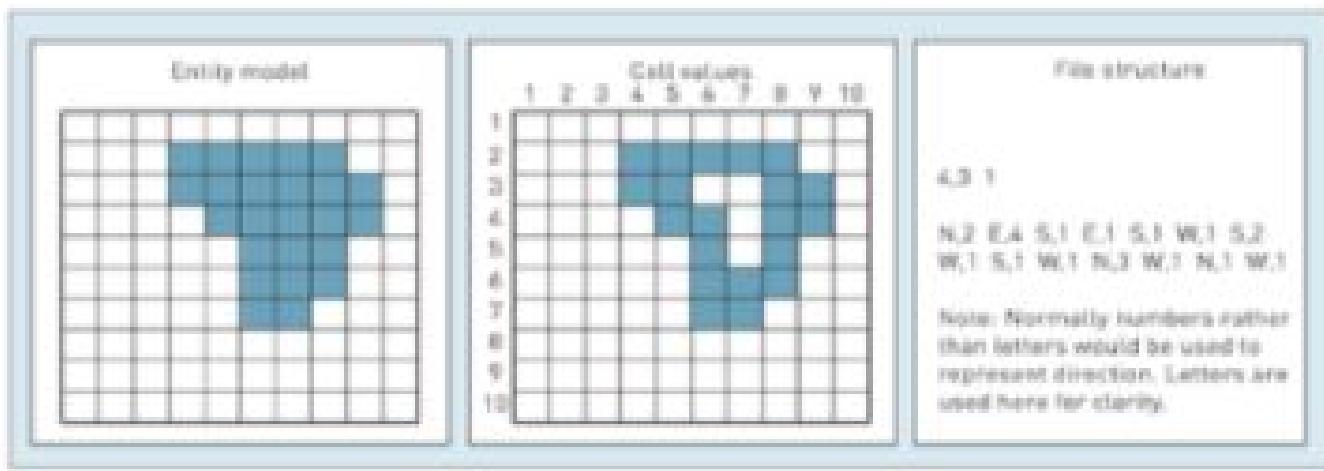
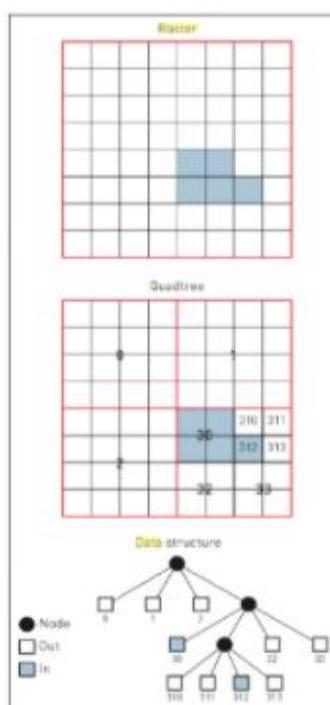
- Block Encoding(Gambar 15.2). Metode ini memperluas dari Run Length encoding menggunakan rangkaian blok untuk menyimpan data. Hampir menyerupai RLE, namun perbedaanya terletak pada dimensionalnya. untuk RLE hanya sepanjang baris saja tetapi block encoding secara dua dimensional (Baris juga Kolom) .

**Gambar 15.2** Gambar Block Encoding

- Chain Encoding(Gambar 15.3). Metode pengurangan data dengan mendefinisikan batas-batas entitas. Metode ini mempresentasikan batas (Boundary) suatu region dengan menggunakan rangkaian arah kardinal dan cell-cell, misal N1 berarti berarah ke utara sejauh 1 cell dan S5 berari berarah keselatan sejauh 5 cell.
- Quadtree Data Structure. Membagi setiap sel dalam image ke dalam empat per empat bagian lalu dibagi lagi ke dalam kelas-kelas. Metode ini menggunakan dekomposisi rekursif dengan membagi grid menjadi hirarki kuadran. Sebuah kuadran yang memiliki nilai yang sama tidak akan dibagi lagi dan disimpan sebagai leaf node. Gambar 15.4 adalah ilustrasi dari Quadtree.

15.1.7 Akses ikonik ke repositori format data raster data monokrom elektronik jarak jauh

Dalam Akses ikonik ke repositori format data raster data monokrom elektronik jarak jauh, Dokumen disimpan dalam sistem menggunakan monokrom, format raster. Dokumen dikirimkan dari repositori ke situs akses jarak jauh untuk ditampilkan kepada pengguna. Kemampuan tambahan disediakan untuk mencari dokumen yang tersimpan; menghasilkan layar antarmuka pengguna sesuai permintaan yang berisi hasil pencarian; memasukkan dokumen ke dalam repositori via transmisi oleh mesin faksimili; dan untuk berkomunikasi secara interaktif antara pengguna sistem. Dokumen elektronik bisa berupa teks dan grafis konvensional; atau dokumen multi media yang berisi teks, video, dan materi audio. Sebuah repositori dokumen fisik tunggal dapat secara logis tersegmentasi menjadi beberapa repositori virtual yang mendukung beragam kelompok pengguna.

**Gambar 15.3** Gambar Chain Encoding**Gambar 15.4** Gambar Quadtree Data Structure

15.1.8 Pengertian PostGIS

Sistem Informasi Geografis (SIG) adalah sistem informasi khusus yang mengelola data yang memiliki informasi spasial (berreferensi keruangan). Atau dalam arti sempit, adalah sistem komputer yang memiliki kemampuan untuk membangun, menyimpan, mengelola dan menampilkan informasi berreferensi geografis, misalnya data yang diidentifikasi menurut lokasinya, dalam sebuah database. SIG juga merupakan sejenis perangkat lunak, perangkat keras (manusia, prosedur, basis data) yang berguna untuk proses pemasukan, penyimpanan, menampilkan data geografis serta attribut-attribut yang terkait. PostGIS adalah extender database spasial gratis untuk PostgreSQL, setiap bit sebaik perangkat lunak berpemilik. Dengan itu, Anda dapat dengan mudah membuat query dengan sadar lokasi hanya dalam beberapa baris kode SQL dan membangun bagian belakang untuk pemetaan, analisis raster, atau aplikasi perutean dengan sedikit usaha. PostGIS dalam Tindakan, mengajarkan untuk memecahkan masalah real- masalah geodata dunia. Ini pertama memberi Anda latar belakang GIS vektor, raster, dan topologi berbasis GIS dan kemudian dengan cepat bergerak untuk menganalisis, melihat, dan memetakan data.

DATA GEOSPASIAL DATA VEKTOR

16.1 Data Vektor

Data vektor dalam Sistem Informasi Geografis. dalam data vektor bumi direpresentasikan sebagai suatu mosaic yang terdiri dari garis (arcline), polygon (daerah yang dibatasi oleh garis yang berawal dan berakhir pada titik yang sama), titik/point (node yang memiliki label), dan nodes (titik perpotongan antara dua buah garis). Titik disimpan sebagai sepasang angka koordinat dan poligon sebagai rangkaian koordinat yang membentuk garis tertutup. Resolusi dari data vektor tergantung dari jumlah titik yang membentuk garis. Fungsi dari vektor telah berkembang dengan pesatnya sehingga tidak hanya berfungsi sebagai sekedar outline akan tetapi telah menjadi sarana menggambarkan artistik yang sangat powerful. Kalian juga dapat menggunakan fasilitas artistic media, yang mengolah kurva menjadi pen-style yang sangat variatif.

16.1.1 Data Raster

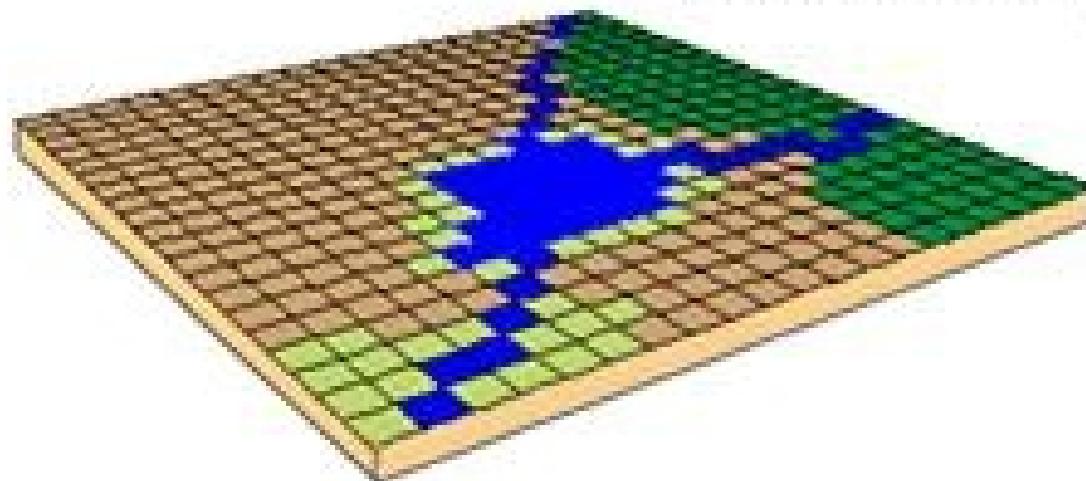
Data raster merupakan data yang disimpan dalam bentuk persegi empat sama sisi (grid) atau sel sehingga terbentuk suatu ruang yang teratur. Pada data raster, objek geografis dipresentasikan sebagai struktur sel grid yang disebut sebagai pixel (picture element). Definisi gambar tergantung pada ukuran pixel-nya, semakin kecil ukuran permukaan bumi yang dipresentasikan oleh sel, semakin tinggi ukuran permukaannya. Pada gambar 16.1 ditampilkan gambar data vektor dan juga data raster yang memiliki perbedaan dalam segi tampilan. Pada data raster gambar akan membentuk seperti potongan bitmap yang apabila kita melihat peta akan tidak begitu jelas di pandang, akan tetapi dalam perhitungan bisa memudahkan kita. Pada data vector gambar yang dihasilkan lebih jelas dan lebih nyata, dan dalam segi perhitungan lebih rumit karena gambar yang dihasilkan lebih jelas dan lebih nyata.

16.1.2 Model Data Vektor

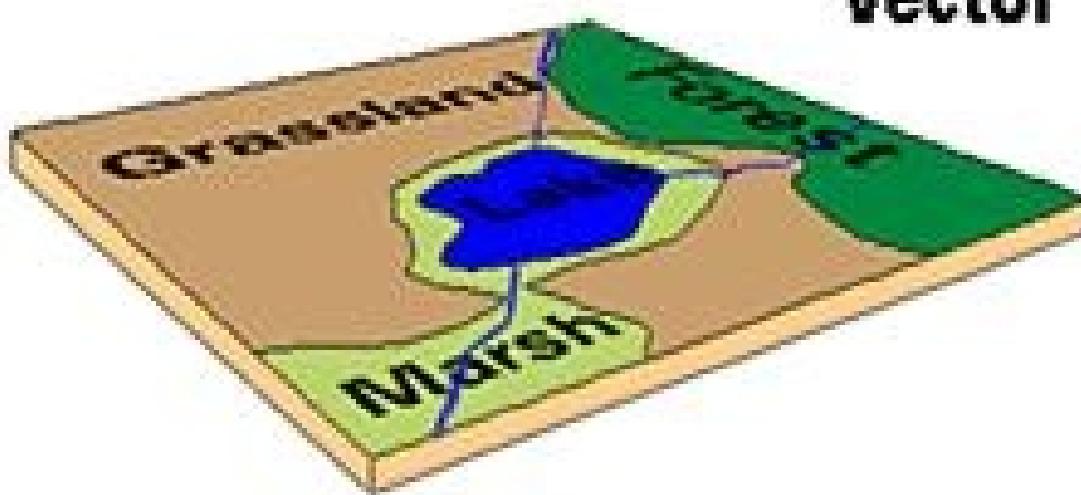
Model data vektor sendiri merupakan model yang banyak digunakan, model ini berbasis pada titik (points) dengan nilai koordinat (x,y) untuk membangun objek spasialnya. Objek yang dibangun terbagi menjadi tiga bagian yaitu berupa titik (point), garis (line) dan area (polygon).

1. Titik (point) merupakan representasi grafis yang paling sederhana pada suatu objek. Titik tidak mempunyai dimensi tetapi dapat ditampilkan dalam bentuk simbol baik pada peta maupun dalam layar monitor. Contoh : lokasi fasilitas kesehatan.
2. Garis (Line) merupakan bentuk linear yang menghubungkan dua atau lebih titik dan merepresentasikan objek dalam satu dimensi. contoh : Jalan dan Sungai.
3. Area (Polygon) merupakan representasi objek dalam dua dimensi. contoh : Danau. Setiap bagian dari data vektor dapat saja mempunyai informasi-informasi yang berasosiasi satu dengan lainnya seperti penggunaan sebuah label untuk menggambarkan informasi pada suatu lokasi.

Raster / Image



Vector



Gambar 16.1 tampilan dari data vektor dan data raster

16.1.3 Perbedaan Data Vektor dan Raster

16.1.4 Objek pada data vektor

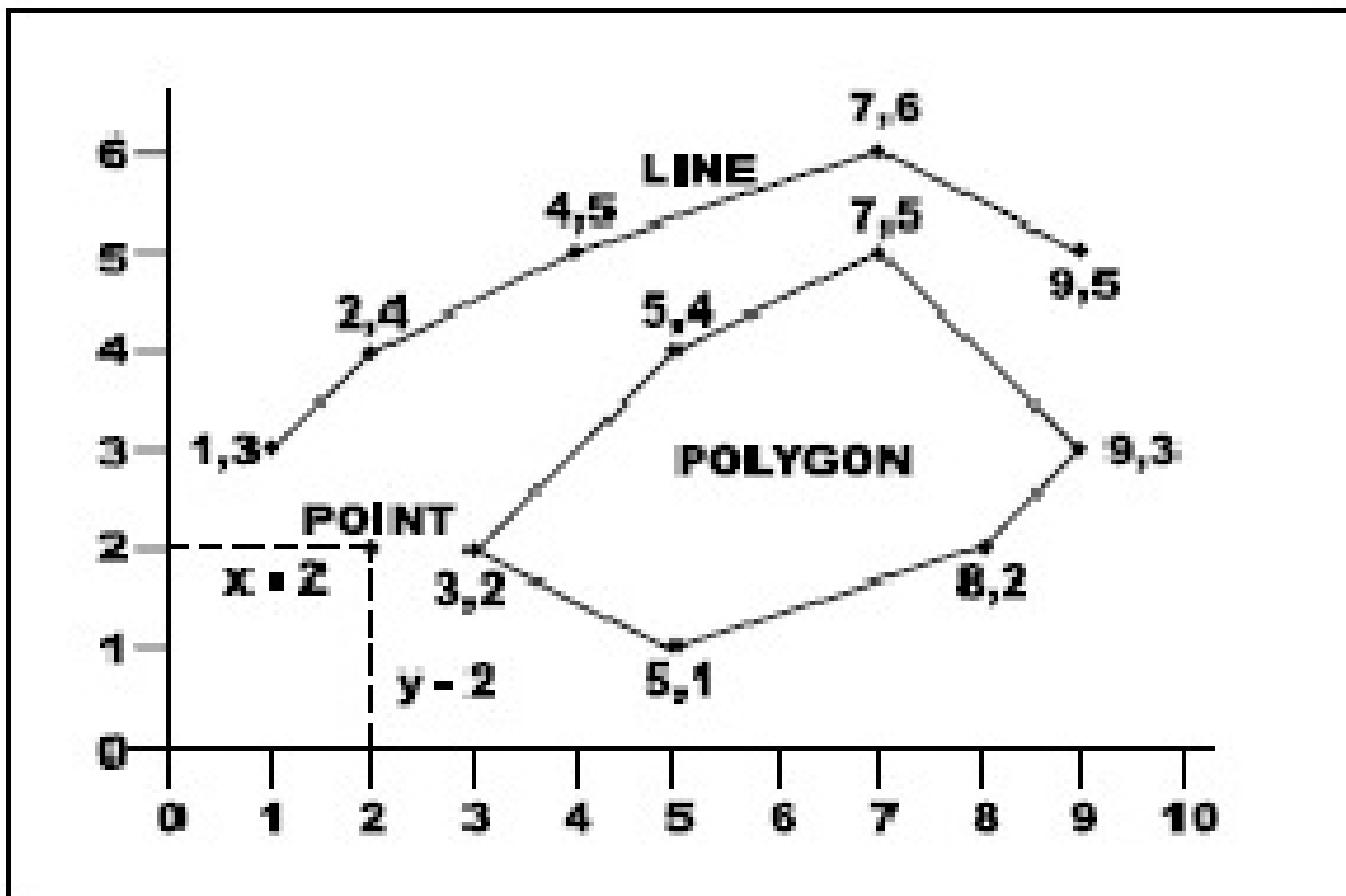
Sesuai pada gambar 16.2 tergambar jelas bahwa pada penggunaan data vektor yang digunakan adalah model yang berbasis titik (point) dengan nilai koordinat (x, y). Objek yang dibangun terbagi menjadi tiga bagian lagi, yaitu berupa titik (point), garis (line), dan area (polygon).

1. titik (point) merupakan representasi grafis yang paling sederhana pada suatu objek. Titik tidak mempunyai dimensi tetapi dapat ditampilkan dalam bentuk symbol baik pada peta maupun dalam layar monitor.
2. garis (line) merupakan bentuk linear yang menghubungkan dua atau lebih titik dan merepresentasikan objek dalam satu dimensi.
3. Area (polygon) merupakan representasi objek dalam dua dimensi.

16.1.5 Cara Menggunaan Data Vektor

Adapun cara menggunakan data vektor ke dalam SIG dapat dilakukan melalui tiga cara yaitu

1. Penyiaman (scanning) adalah proses mengubah data grafis kontinu menjadi data grafis diskrit yang terdiri atas sel-sel penyusun gambar.
2. Digital merupakan proses pengubahan data grafis analog menjadi data grafis digital dalam struktur vektor.
3. Tabulasi adalah proses memasukkan data atribut SIG dengan pembuatan tabel.



Gambar 16.2 tampilan skala data vektor

16.1.6 Aplikasi Yang Memudahkan Pembuatan Data Vektor

16.1.6.1 Microsoft Visio Microsoft Visio atau yang sering kita sebut visio merupakan suatu program aplikasi perangkat komputer yang memang sering digunakan atau dipakai untuk membuat sebuah diagram, misalnya seperti diagram aliran (flowchart), brainstorm, dan juga skema pada jaringan yang dikeluarkan atau dirilis oleh Microsoft Corporation. biasanya aplikasi ini memakai grafik vektor untuk menciptakan atau membuat diagram-diagramnya.

16.1.6.2 Corel Draw CorelDraw merupakan aplikasi perangkat komputer yang juga sering digunakan sebagai pengolahan vektor. coreldraw ini juga merupakan fasilitas yang cukup sangat lengkap di banding aplikasi yang lainnya, coreldraw juga bisa digunakan sebagai pengelolaan kurva bezier. kurva bezier (bezier curve) merupakan garis lengkung yang terbentuk dari rumusan matematis. kurva bezier memiliki sebuah titik pusat (anchor point) dengan garis pengendali arah atau handle. dan dari handle ini dapat kita atur seluruh kelengkungan objek melalui dua titik yang bisa di sebut sebagai titik tangen dari kurva bezier ini.

16.1.7 Kekurangan dan Kelebihan Data Vektor

16.1.7.1 Kelebihan Data Vektor Kelebihan data vektor di bandingkan data raster data vektor relatif lebih ekonomis dalam hal ukuran file dan presisi dalam lokasi, tetapi sangat sulit untuk digunakan dalam komputasi matematik.

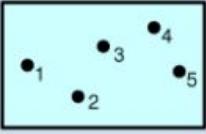
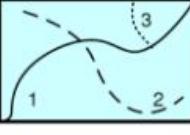
1. Mempunyai struktur data yang sangat sederhana.
2. sangat mudah dimanipulasi dengan menggunakan fungsi-fungsi matematis yang sederhana.
3. pada data vektor teknologi yang digunakan sangat murah dan tidak begitu kompleks sehingga pengguna bisa membaca langsung seperti file teks.
4. bisa membuat sendiri program aplikasi yang menggunakan citra raster.
5. cocok atau sesuai dengan citra-citra satelit penginderaan jauh dan semua image hasil scanning data spasial.
6. memiliki kombinasi data raster dengan data inderaja mudah dilakukan
7. mempunyai beberapa kemampuan-kemampuan permodelan dan analisis spasial tingkat lanjut.
8. pada data vektor metode untuk mendapatkan citra raster lebih mudah.

9. model permukaan bumi di dalam bentuk citra raster yang didapat dari radar atau satelit penginderaan jauh selalu lebih aktual dari pada bentuk vektornya
10. Prosedur untuk memperoleh data dalam bentuk raster lebih mudah, sederhana dan murah.
11. Harga system perangkat lunak aplikasinya cenderung lebih murah.
12. Memiliki resolusi spasial yang tinggi.
13. Transformasi koordinat dan proyeksi tidak sulit dilakukan.
14. Hubungan topologi dan network dapat dilakukan dengan mudah.

16.1.7.2 Kekurangan data Vektor kekurangan data vektor dibandingkan data raster terdapat keterbatasan masalah akurasi dan presisi data terutama dalam menentukan ukuran piksel. Data vektor memiliki keterbatasan dalam ukuran penyimpanan atau kapasitas hasil.

1. Memiliki struktur data yang kompleks.
2. Datanya tidak mudah untuk dimanipulasi.
3. Pengguna tidak mudah berkreasi untuk membuat programnya sendiri untuk memenuhi kebutuhan aplikasinya. Hal ini disebabkan oleh struktur data vektor yang lebih kompleks dan prosedur fungsi dan analisisnya memerlukan kemampuan tinggi karena lebih sulit. Pengguna harus membeli sistem perangkat lunaknya karena teknologinya masih mahal.
4. Prosedurnya pun terkadang lebih sulit. Karena proses keseluruhan untuk mendapatkannya lebih lama, peta vektor seringkali mengalami out of date atau kadaluarsa.
5. Memerlukan perangkat keras dan perangkat lunak yang lebih mahal.
6. Overlay beberapa layers vektor secara simultan memerlukan waktu yang relatif lama.
7. Karena proses keseluruhan untuk mendapatkannya lebih lama, peta vektor seringkali mengalami out of date atau kadaluarsa.
8. data ini adalah ketidakmampuannya dalam mengakomodasi perubahan fenomena yang bersifat gradual.

16.1.8 Model Data Vektor

Jenis	Contoh Representasi	Contoh Atribut																		
Titik		<table border="1"> <thead> <tr> <th>ID</th><th>Nama</th><th>Lokasi</th></tr> </thead> <tbody> <tr> <td>1</td><td>SMU 1</td><td>Kec. A</td></tr> <tr> <td>2</td><td>SDN B</td><td>Kec. A</td></tr> <tr> <td>3</td><td>SMP 5</td><td>Kec. A</td></tr> <tr> <td>4</td><td>SDN A</td><td>Kec. B</td></tr> <tr> <td>5</td><td>SMU 2</td><td>Kec. B</td></tr> </tbody> </table>	ID	Nama	Lokasi	1	SMU 1	Kec. A	2	SDN B	Kec. A	3	SMP 5	Kec. A	4	SDN A	Kec. B	5	SMU 2	Kec. B
ID	Nama	Lokasi																		
1	SMU 1	Kec. A																		
2	SDN B	Kec. A																		
3	SMP 5	Kec. A																		
4	SDN A	Kec. B																		
5	SMU 2	Kec. B																		
Garis		<table border="1"> <thead> <tr> <th>ID</th><th>Status Jalan</th><th>Kondisi</th></tr> </thead> <tbody> <tr> <td>1</td><td>Jalan Nasional</td><td>Baik</td></tr> <tr> <td>2</td><td>Jalan Provinsi</td><td>Sedang</td></tr> <tr> <td>3</td><td>Jalan Kabupaten</td><td>Rusak</td></tr> </tbody> </table>	ID	Status Jalan	Kondisi	1	Jalan Nasional	Baik	2	Jalan Provinsi	Sedang	3	Jalan Kabupaten	Rusak						
ID	Status Jalan	Kondisi																		
1	Jalan Nasional	Baik																		
2	Jalan Provinsi	Sedang																		
3	Jalan Kabupaten	Rusak																		
Poligon		<table border="1"> <thead> <tr> <th>ID</th><th>Guna Lahan</th><th>Luas (Ha)</th></tr> </thead> <tbody> <tr> <td>1</td><td>Sawah</td><td>20</td></tr> <tr> <td>2</td><td>Permukiman</td><td>30</td></tr> <tr> <td>3</td><td>Kebun</td><td>45</td></tr> <tr> <td>4</td><td>Danau</td><td>40</td></tr> </tbody> </table>	ID	Guna Lahan	Luas (Ha)	1	Sawah	20	2	Permukiman	30	3	Kebun	45	4	Danau	40			
ID	Guna Lahan	Luas (Ha)																		
1	Sawah	20																		
2	Permukiman	30																		
3	Kebun	45																		
4	Danau	40																		

Gambar 16.3 representasi data vektor dan atributnya

Sesuai dengan gambar 16.3 model data vektor terbagi menjadi beberapa bagian[38] :

1. Topologi, biasa digunakan dalam analisis spasial dalam SIG. topologi adalah model data vektor yang menunjukkan hubungan spasial di antara objek spasial. Topologi sangat berguna pada saat melakukan seteksi kesalahan ketika proses digitasi, dan berguna pula dalam melakukan proses analisis spasial. Model data vektor dalam topologi dikembangkan dalam dua kategori,yaitu Data Sederhana (Simple Data) yang merupakan representasi data yang mengandung tiga jenis data (titik,garis,polygon)

secara sederhana.Sedangkan Data Tingkat Tinggi (Higher Data Level),dikembangkan lebih jauh dalam melakukan pemodelan secara tiga dimensi (3 Dimensi/3D).

2. Non Topologi, adalah model data yang mempunyai sifat yang lebih cepat dalam menampilkan dan dapat digunakan secara langsung dalam perangkat lunak (software) SIG yang berbeda-beda.
3. Region, adalah sekumpulan polygon, dimana masing-masing polygon dapat atau tidak mempunyai kaitan di antaranya akan tetapi saling bertampalan dalam satu data set.
4. DynamicSegmentation, merupakan model data yang dibangun dengan menggunakan garis dalam rangka membangun model jaringan (network).
5. Model data vektor dikenal pula dengan model data spaghetti.Pada model ini,lembaran kertas peta ditranslasikan garis-garis ke dalam list koordinat (x,y) dalam format digital.Sebuah titik dikodekan sebagai pasangan koordinat (x,y) tunggal.Sementara area atau ulasan dikodekan sebagai poligon dan direkam sebagai pasangan koordinat closed-loop yang didefinisikan batas-batasannya.

16.1.9 Klasifikasi Model Data Spasial

Klasifikasi model data spasial dibagi menjadi dua bagian yaitu data vektor dan model data raster, kemudian data vektor dibagi menjadi dua bagian, yaitu non-topologi dan topologi. Kemudian data topologi tersebut dibagi lagi menjadi dua bagian yaitu data sederhana(simple data) dan data tingkat tinggi(higher data level), lalu data tingkat tinggi di bagi menjadi tiga bagian, yaitu TIN(triangulated irregular network), regions, dan dynamic segmentation. Itulah klasifikasi model data spasial yang di dalamnya banyak sekali bagian yang sangat spesifik, sehingga klasifikasi model data spasial ini bisa dikategorikan sebagai klasifikasi model yang sangat kompleks.

Pada data vektor ini, kita dapat simpulkan bahwa penggunaan suatu aplikasi dapat disesuaikan dengan apa saja hal-hal yang kita perlukan dan kita inginkan agar dapat sesuai pada proses dan juga hasilnya yang kita harapkan. Sehingga data data yang telah dikumpulkan dapat digunakan untuk keperluan sistem informasi geografis yang sangat lengkap.

Pada data vektor yang dijelaskan bahwa data berikut merupakan data yang valid dengan menggunakan model data spasial dengan menggunakan titik, dan menyimpan dataspasial menggunakan garis-garis atau kurva atau poligon beserta atribut-atributnya. bentuk-bentuk dasar sepsertasi data yang ada data spasial ini, di dalam sistem dalam ini dalam sistem kordinat data vektor. jarak antara rumah didesa sistem perhubungannya dan kondisi geografis. [39]

STANDAR OPEN GEOSPATIAL CONSORTIUM

17.1 Open Geospatial Consortium

17.1.1 Definisi

Logo Open Geospatial Consortium 17.1 Open Geospatial Consortium (OGC) Web Services(OWS) adalah layanan yang didefinisikan oleh OGC, yang memungkinkan semua jenis fungsi geospasial. Ini termasuk layanan untuk akses data, tampilan data dan pengolahan data. Permintaan OWS didefinisikan dengan menggunakan protokol Hyper Text Transfer Protocol (HTTP) dan dikodekan menggunakan struktur key value pair (KVP) atau Extensible Markup Language (XML). OWS yang paling banyak dikenal adalah Web Map Service (WMS). [40]



Gambar 17.1 Logo OGC

17.1.1.1 Latar Belakang Sejarah OGC adalah konsorsium industri internasional dari perusahaan, instansi pemerintah, organisasi penelitian, dan universitas yang berpartisipasi dalam proses konsensus untuk mengembangkan spesifikasi antarmuka yang tersedia bagi publik. Standar OpenGIS mendukung solusi interoperabilitas yang ‘mengaktifkan geo’ layanan Web, nirkabel dan

berbasis lokasi, dan arus utama TI. Pada awal 1990an, OGC mendefinisikan sebuah visi untuk komputasi geospasial berbasis jaringan. Baru-baru ini visi ini telah membawa hasil dengan menggunakan layanan web. Bagian ini memberikan penglihatan dari tahun 1990an diikuti oleh bagian selanjutnya yang mendefinisikan arsitektur Layanan OGC Web Services.

Penerapan komputer dan penggunaan sistem informasi geografis (GIS) secara luas telah menyebabkan peningkatan analisis data geografis dalam banyak disiplin ilmu. Berdasarkan kemajuan teknologi informasi, ketergantungan masyarakat terhadap data tersebut semakin meningkat. Kumpulan data geografis semakin banyak dibagi, dipertukarkan, dan digunakan untuk tujuan selain yang diinginkan produsen mereka. GIS, penginderaan jarak jauh, pemetaan otomatis dan manajemen fasilitas (AM / FM), analisis lalu lintas, sistem geopositioning, dan teknologi lainnya untuk Informasi Geografis (GI) memasuki periode integrasi radikal.

Standar untuk interoperabilitas geospasial memberikan kerangka bagi pengembang untuk membuat perangkat lunak yang memungkinkan pengguna mengakses dan memproses data geografis dari berbagai sumber di antarmuka komputer generik dalam lingkungan teknologi terbuka.

- kerangka kerja untuk pengembang berarti bahwa Standar Internasional didasarkan pada rencana umum yang komprehensif, umum (yaitu, dibentuk oleh konsensus untuk penggunaan umum) untuk pengembangan geoprocessing yang dapat dioperasikan.
- akses dan proses berarti orang-orang yang menggunakan database query remote dan mengendalikan sumber daya pemrosesan jarak jauh, dan juga memanfaatkan teknologi komputasi terdistribusi lainnya seperti perangkat lunak yang dikirim ke lingkungan lokal pengguna dari lingkungan terpencil untuk penggunaan sementara.
- dari berbagai sumber berarti bahwa pengguna tidak akan dapat mengakses data yang diperoleh dari berbagai cara dan disimpan dalam berbagai macam database relasional dan nonrelasional.
- di antarmuka komputasi generik berarti bahwa antarmuka standar menyediakan komunikasi yang andal antara sumber daya perangkat lunak yang berbeda yang dilengkapi untuk menggunakan antarmuka ini.
- dalam lingkungan teknologi informasi yang terbuka berarti bahwa standar memungkinkan proses geoprocessing berlangsung di luar lingkungan monolog GIS, penginderaan jarak jauh, dan AM / FM yang tertutup yang mengendalikan dan membatasi basis data, antarmuka pengguna, jaringan, dan fungsi manipulasi data[40].

17.1.2 Dasar-dasar Ilmiah

Prinsip dasar arsitektur OGC Web Services (OWS), meliputi:

- Komponen layanan disusun dalam beberapa tingkatan.
 1. Semua komponen memberikan layanan, kepada klien dan / atau komponen lainnya, dan setiap komponen biasanya disebut layanan (dengan beberapa implementasi) atau server (masing-masing implementasi).
 2. Layanan (atau komponen) diatur secara longgar dalam empat tingkatan, dari Klien sampai Layanan Aplikasi hingga Jasa Pemrosesan ke Layanan Manajemen Informasi, namun tingkat yang tidak diperlukan dapat dilewati.
 3. Layanan dapat menggunakan layanan lain dalam tingkat yang sama, dan ini biasa terjadi pada tingkat Jasa Pengolahan.
 4. Server dapat beroperasi pada (terikat ketat) data yang tersimpan dalam server dan / atau data (longgar terikat) yang diambil dari server lain.
- Kolaborasi layanan menghasilkan hasil yang lebih baik.
 1. Semua layanan menggambarkan diri sendiri, mendukung pengikatan layanan dinamis yang mendukung (just-in-time) yang mendukung publikasi.
 2. Layanan dapat dirantai dengan layanan lain dan sering dirantai, transparan (didefinisikan dan dikendalikan oleh klien), tembus pandang (dideklarasikan tapi terlihat oleh klien), dan tidak jelas (dideklarasikan dan tidak terlihat oleh klien), lihat Sub ayat 7.3. 5 dari [ISO 19119]
 3. Layanan disediakan untuk memudahkan mendefinisikan dan melaksanakan rantai layanan.
- Komunikasi layanan menggunakan standar Internet terbuka.
 1. Komunikasi antar komponen menggunakan protokol World Wide Web (WWW) standar, yaitu HTTP GET, HTTP POST, dan SOAP.
 2. Operasi server khusus ditangani dengan menggunakan Uniform Resource Locators (URL).
 3. Multiguna Internet Mail Extensions (MIME) jenis digunakan untuk mengidentifikasi format transfer data.
 4. Data yang ditransfer sering dikodekan menggunakan Extensible Markup Language (XML), dengan isi dan format yang ditentukan menggunakan Skema XML.
- Antarmuka layanan menggunakan standar terbuka dan relatif sederhana.
 1. Antarmuka webserver OGC digabungkan, hanya memberikan beberapa operasi statis per layanan.
 2. Operasi layanan biasanya tanpa kewarganegaraan, tidak memerlukan server untuk mempertahankan status antarmuka di antara operasi.

3. Satu server dapat mengimplementasikan beberapa antarmuka layanan kapanpun bermanfaat.
 4. Standar pengkodean data berbasis XML XML ditentukan untuk digunakan dalam transfer data.
- Implementasi server dan client tidak dibatasi.
 1. Layanan diimplementasikan oleh perangkat lunak yang dijalankan pada komputer tujuan umum yang terhubung ke Internet. Arsitekturnya adalah perangkat keras dan vendor perangkat lunak yang netral.
 2. Layanan yang sama dan bekerja sama dapat dilakukan oleh server yang dimiliki dan dioperasikan oleh organisasi independen.
 3. Banyak layanan diimplementasikan dengan software Commercial Off The Shelf (COTS) berbasis standar[40].

17.1.3 Pengertian Geospasial

Informasi Geospasial, yang lazim dikenal dengan peta, adalah informasi obyek permukaan bumi yang mencakup aspek waktu dan keruangan. Pengertian geo dalam geospasial, berarti geosfer yang mencakup atmosfer/lapisan udara yang meliputi permukaan bumi, litosfer lapisan kulit bumi, pedosfer tanah beserta pembentukan dan zona-zonanya, sebagai bagian dari kulit bumi, hidrosfer lapisan air yang menutupi permukaan bumi dalam berbagai bentuknya, biosfer segenap unsur di permukaan bumi yang membuat kehidupan dan proses biotik berlangsung dan antroposfer manusia dengan segala aktivitas yang dilakukannya di permukaan bumi.

17.1.3.1 Sejarah OGC adalah konsorsium industri internasional dari perusahaan, instansi pemerintah, organisasi penelitian, dan universitas yang berpartisipasi dalam proses konsensus untuk mengembangkan spesifikasi antarmuka yang tersedia bagi publik. Standar OpenGIS mendukung solusi interoperabilitas yang ‘mengaktifkan geo’ layanan Web, nirkabel dan berbasis lokasi, dan arus utama TI. Pada awal 1990an, OGC mendefinisikan sebuah visi untuk komputasi geospasial berbasis jaringan. Baru-baru ini visi ini telah membawa hasil dengan menggunakan layanan web. Bagian ini memberikan penglihatan dari tahun 1990an diikuti oleh bagian selanjutnya yang mendefinisikan arsitektur Layanan OGC Web Services. Penerapan komputer dan penggunaan sistem informasi geografis (GIS) secara luas telah menyebabkan peningkatan analisis data geografis dalam banyak disiplin ilmu. Berdasarkan kemajuan teknologi informasi, ketergantungan masyarakat terhadap data tersebut semakin meningkat. Kumpulan data geografis semakin banyak dibagi, dipertukarkan, dan digunakan untuk tujuan selain yang diinginkan produsen mereka. GIS, penginderaan jarak jauh, pemetaan otomatis dan manajemen fasilitas (AM / FM), analisis lalu lintas, sistem geopositioning, dan teknologi lainnya untuk Informasi Geografis (GI) memasuki periode integrasi radikal[40].

17.1.4 Definisi

Open Geospatial Consortium (OGC) Web Services(OWS) adalah layanan yang didefinisikan oleh OGC, yang memungkinkan semua jenis fungsi geospasial. Ini termasuk layanan untuk akses data, tampilan data dan pengolahan data. Permintaan OWS didefinisikan dengan menggunakan protokol Hyper Text Transfer Protocol (HTTP) dan dikodekan menggunakan struktur keyvalue-pair (KVP) atau Extensible Markup Language (XML). OWS yang paling banyak dikenal adalah Web Map Service (WMS). [40]

17.1.5 GeospatialWebService

Geospatial Web Service adalah jenis layanan web khusus yang menyediakan akses ke informasi geografis yang heterogen di internet. OGC telah mengembangkan beberapa spesifikasi layanan web untuk menstandardisasi layanan web geospasial untuk mengakses data dan aplikasi geospasial. Layanan web geospasial yang penting meliputi Web Feature Service (WFS), Web Map Service (WMS), Web Coverage Service (WCS), Layanan Katalog (CS), dan Web Processing Service (WPS) seperti pada gambar 17.2 [40].

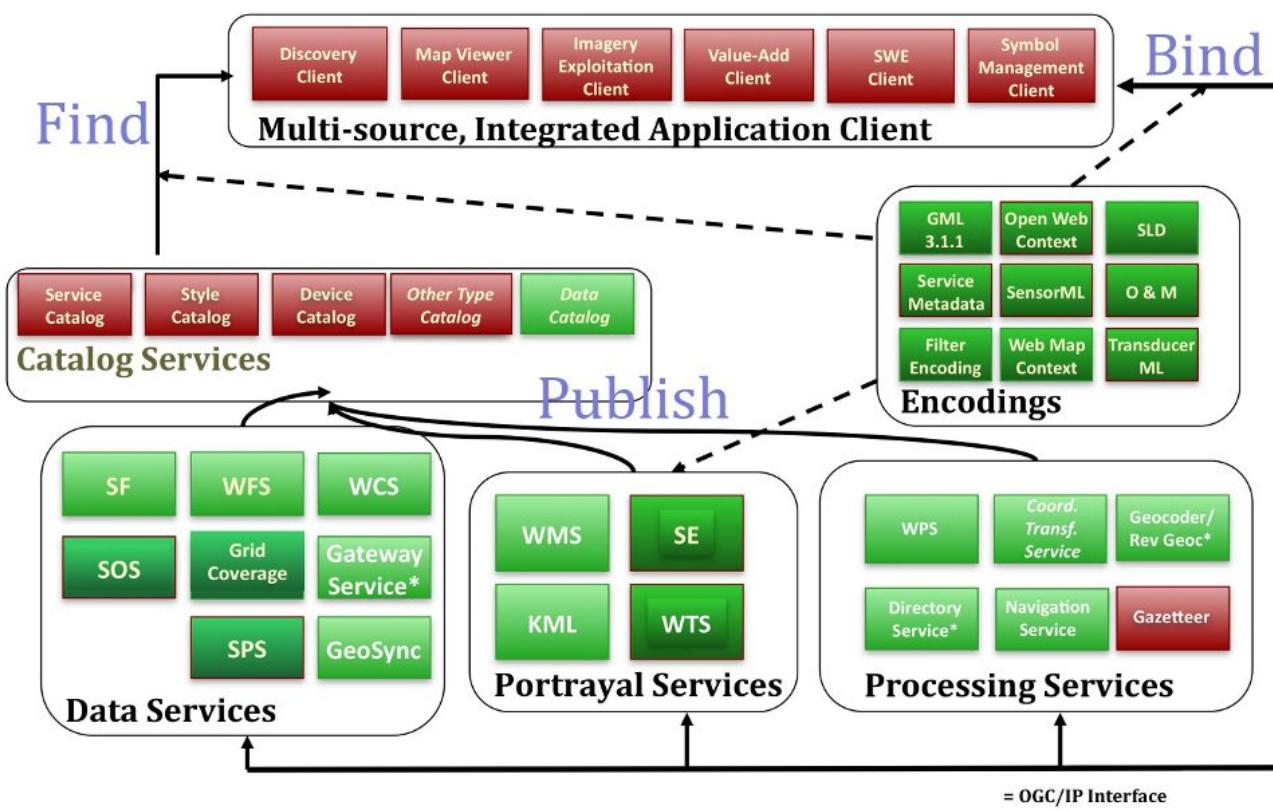
17.1.6 GeospatialSemanticWeb

Geospatial Semantic Web berkaitan dengan informasi geografis bahwa penelitian semantik web dasar tidak ditujukan untuk memperbaiki hasil query yang mencari informasi yang tersimpan dalam database geografis. Geospatial Semantic Web Service alamat semantik masalah heterogen ditemukan dalam layanan web geospasial. Layanan web semantik geospasial mendefinisikan data geospasial dan layanan pemrosesan dalam hal semantik dengan membangun pada entologi dan kemudian memberikan makna spesifik pada entologi tersebut. [40]

17.1.7 OGC Standar

Standar OGC adalah dokumen teknis yang detail antarmuka atau pengkodean. Pengembang perangkat lunak menggunakan dokumen-dokumen ini untuk membangun antarmuka dan pengkodean yang terbuka ke dalam produk dan layanan mereka. Standar ini merupakan “produk” utama dari Konsorsium Geospasial Terbuka dan telah dikembangkan oleh anggotanya untuk mengatasi tantangan interoperabilitas yang spesifik. Idealnya, ketika standar OGC diterapkan pada produk atau layanan online oleh dua insinyur perangkat lunak yang berbeda yang bekerja secara independen, komponen dan plug and play yang dihasilkan, artinya, mereka bekerja sama tanpa melakukan debug lebih jauh. OGC mempertahankan dua jalur standar: Jalur standar penuh dan jalur standar Komunitas. Masing-masing diringkas di bawah ini. Jalur standar penuh: Jalur standar penuh adalah proses konsensus untuk mengembangkan dan menyetujui standar di dalam Komite Teknis OGC. Dalam jalur ini, Kelompok Kerja Standar dibuat dan kelompok tersebut menulis standar dan mendukung proses persetujuan di Panitia Teknis. Ada dua level dalam track ini:

Web Services Framework Of OGC Geoprocessing Standards



3

Gambar 17.2 OGC Geoprocessing Standards

1. Standar OGC: ini adalah standar OGC tradisional yang menghasilkan standar yang dapat diterapkan dan dapat diuji atau model konseptual dari mana standar implementasi dapat dikembangkan; dan
2. Standar OGC dengan Compliance Suite: ini adalah standar OGC dengan kemampuan yang telah terbukti untuk diterapkan. Untuk mencapai tingkat ini, standar OGC harus memiliki setidaknya tiga implementasi referensi dan harus ada paket uji kepatuhan Program OGC Compliance untuk semua fitur wajib standar.

Jalur standar penuh dapat menggunakan spesifikasi yang ada untuk membentuk dasar standar baru. Namun, dalam proses ini, keanggotaan OGC telah berkomitmen untuk mendukung dan mempertahankan standar melalui siklus hidupnya Standar komunitas : standar Komunitas adalah posisi resmi OGC yang mendukung spesifikasi atau standar yang dikembangkan di luar OGC. Standar Komunitas dianggap sebagai standar normatif oleh keanggotaan OGC dan bagian dari Baseline Standar OGC. Pertimbangan utama untuk standar Komunitas adalah bahwa harus ada bukti pelaksanaan yang kuat. OGC tidak mengambil alih pemeliharaan pekerjaan, namun standar Komunitas adalah “cuplikan” dari standar matang dimana penggas tersebut telah membagikan Hak Kekayaan Intelektual dengan OGC atau memberikan penggunaan tak terbatas atas Kekayaan Intelektual kepada semua pelaksana . Standar masyarakat dapat melayani dua tujuan:

1. untuk membawa standar de facto dari komunitas geospasial yang lebih besar menjadi titik acuan yang stabil yang dapat secara normatif dirujuk oleh pemerintah dan organisasi lainnya; dan
2. untuk membawa standar baru, namun diimplementasikan, ke OGC untuk membentuk dasar bagi penyempurnaan dan pengembangan interoperabilitas lebih lanjut antara standar OGC lainnya.

Standar OGC dan dokumen pendukung tersedia untuk umum tanpa biaya apapun. OGC Web Services (OWS) adalah standar OGC yang dibuat untuk digunakan dalam aplikasi World Wide Web. [40]

DATA GEOSPASIAL WEB MAP TILE SERVICE

18.1 Web Map Tile Service

Web Map Tile Service merupakan peta ubin yang dikembangkan pertama kali oleh Open Geospatial Consortium (OGC). Terdapat potensi di dalam Web Map Tile Services yaitu gambar peta ubin dapat di cache pada lokasi antara klien dan server, mengurangi latensi yang terkait dengan proses pembuatan gambar. Lapisan ubin biasanya di pasang di sisi server, menyajikan ubin gambar peta secara bersamaan ke beberapa pengguna. Selain itu, banyak pemetaan klien, seperti Google Earth atau Nasa World Wind, telah menyematkan cache, yang juga dapat mengurangi kemacetan jaringan dan penundaan jaringan [41].

WMTS telah menjadi populer untuk visualisasi data geografis multi-dimensi di Internet. Dalam sistem tile, data disusun pada sejumlah skala yang telah ditentukan. Untuk setiap skala, area yang dipetakan dibagi menjadi banyak ubin persegi dengan ukuran 256 x 256 piksel. Setiap ubin disimpan sebagai gambar di server Internet. Menggunakan ubin untuk memungkinkan visualisasi yang cepat di peta interaktif pada banyak skala ditunjukkan oleh Badan Penelusuran Angkasa dan Angkutan Udara Nasional (NASA) Global Cycle (Browse) 2013 di <http://earthdata.nasa.gov/labs/worldview/> yang memberikan akses yang sangat responsif dan terukur terhadap citra real-time (Cechini et al., 2013). Adopsi standar WMTS yang luas untuk penerbitan data salju diilustrasikan oleh katalog penelusuran ArcGIS Online (<http://arcgis.com>), di mana lebih dari 50 kumpulan data WMTS yang terkait salju dapat ditemukan. WMTS dapat diakses menggunakan browser web dan sistem informasi geografis (SIG) [42].

18.1.1 Skema Ubin

Peta sudah lama dikenal hanya seperti yang tercetak di atas kertas. Peta kartografi tercetak tersebut merupakan representasi statis yang terbatas pada skala visualisasi tetap dengan Tingkat Detil tertentu (LOD). Namun, dengan perkembangan peta digital, pengguna dapat memperbesar atau mengurangi area yang divisualisasikan dengan melakukan pembesaran operasi, dan LOD diharapkan dapat diperbarui sesuai dengan itu.

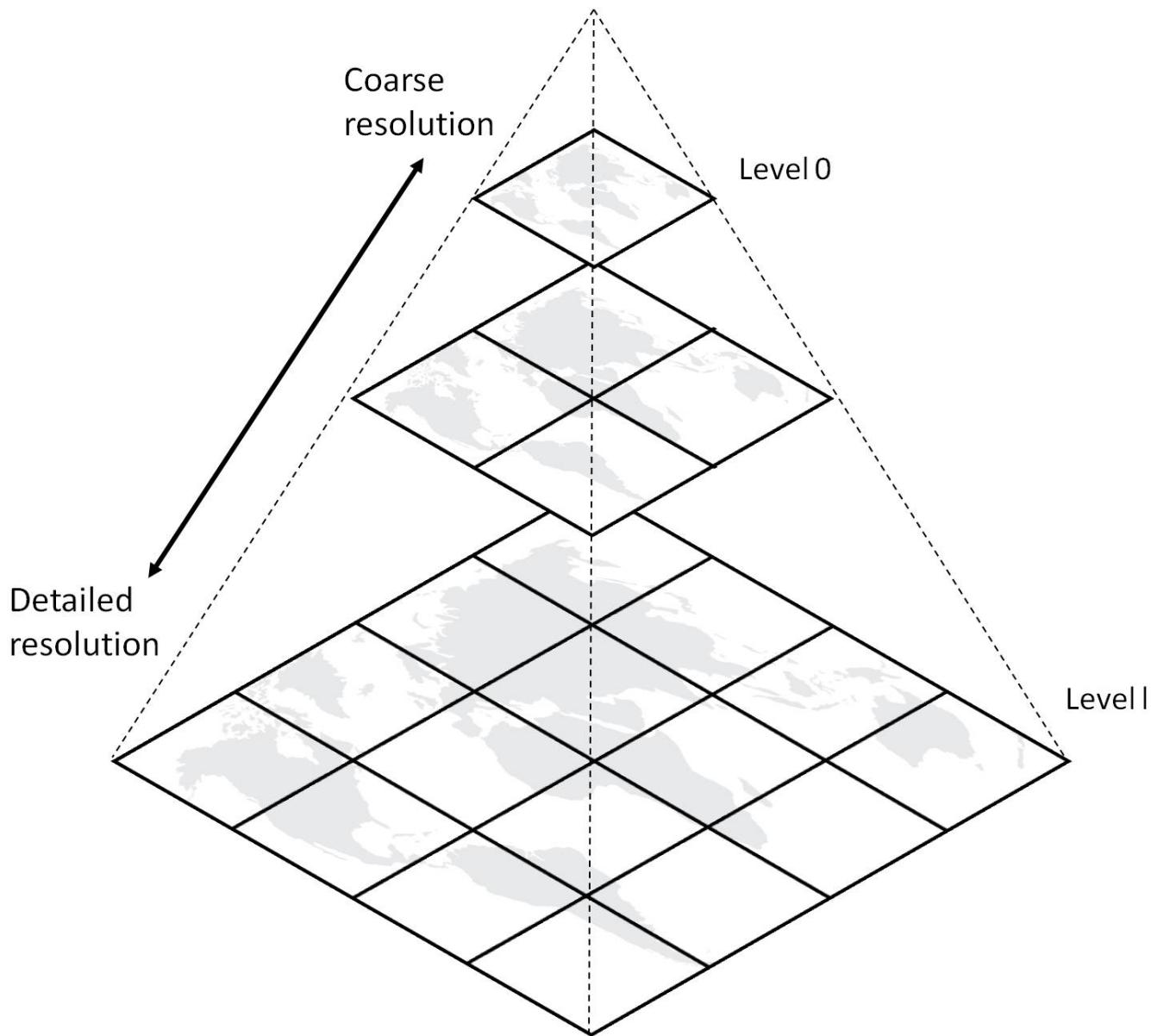
Adaptasi konten peta sangat bergantung pada skala: Peta skala kecil berisi informasi yang kurang rinci daripada peta skala besar di wilayah yang sama. Proses mengurangi jumlah data dan menyesuaikan informasi dengan skala yang diberikan disebut generalisasi kartografi, dan biasanya dilakukan oleh server peta web.

Untuk menawarkan layanan peta web ubin, server peta web membuat peta melintasi serangkaian skala tetap melalui generalisasi yang progresif. Gambar peta yang dirender kemudian dibagi menjadi ubin, menggambarkan piramida ubin seperti yang digambarkan pada gambar 18.1.

gambar 18.1 merupakan contoh skema ubin dari Microsoft Bing Maps dimana tingkat pertama memungkinkan mewakili seluruh dunia dalam 4 ubin (2x2) dari 256x256 piksel. Tingkat berikutnya mewakili seluruh dunia dalam 16 ubin (4x4) dari 256x256 piksel dan seterusnya sampai tingkat 4.

18.1.2 Metode Pemotongan Web Map Tiles Service

Web GIS standar dibangun menggunakan WMS (Web Map Services). Ciri khas Web Gis dalam menampilkan data spasial atau map adalah menarik langsung dari server dengan tidak memperhitungkan banyaknya lapisan yang diminta. Padahal proses menampilkan data spasial atau map ini tidak bisa dipaksakan menarik dalam jumlah besar karena akan memperlambat waktu



Gambar 18.1 Representasi piramida ubin

respon web. Hal ini berbanding terbalik dengan harapan pengguna akan Web GIS yang mudah dioperasikan, memiliki tampilan yang ramah, pengaksesan yang halus dan data cukup update atau mendekati near real time[7] (Yang, 2011).

Pendekatan umum untuk meningkatkan waktu respon saat permintaan data spasial adalah dengan memotong map menjadi beberapa bagian kecil atau tiles. Metode pemotongan ini secara standar bisa dibuat dengan WMTS (Web Map Tile Service) atau dikenal dengan tiles tradisional. Pemotongan standar cukup bisa mengurangi beban server namun waktu respon web masih lambat. Hal itu menjadi masalah yang saat ini dihadapi oleh Web Gis E-Government Sistem Pemantauan Bumi Nasional (SPBN). Pengguna harus menunggu lama waktu respon web. Padahal banyak informasi penting yang disampaikan khususnya terkait tanggap bencana titik panas kebakaran hutan yang terjadi saat ini. Dengan melihat fakta-fakta di atas, Penelitian ini ingin menganalisis penggunaan tiling pada open source web GIS, metode ini mempertimbangkan tile static dan dynamic map. Untuk menganalisa digunakan metode matematika, data hasil testing, statistik hasil pengujian dan membandingkan beberapa metode yang telah berkembang untuk meningkatkan performa web gis. Penulis berharap analisis ini dapat meminimalkan waktu respon dan meningkatkan pelayanan E-Government Sistem Pemantauan Bumi Nasional (SPBN). Oleh karena itu penulis melakukan studi ‘Analisis Optimalisasi Web Gis Dengan Metode Tiling’. Penelitian ini diharapkan dapat membantu memecahkan masalah diatas [43].

18.1.3 Peta Jalan

proses pemetaan bumi secara akurat sampai saat ini, melestarikan yang sangat terampil, well-equipped dan individu terorganisir perkelompok. Bertahun-tahun biasanya peran surveyor, kartografer, dan ahli geografi untuk memetakan dunia dengan tulisan di atas kertas atau, sejak tahun 1960an, masuk komputer. Ekspedisi Lewis dan Clark ke peta Amerika Utara Barat, dan Lambton dan Ekspedisi Everest’s Great Arc untuk India, hanya dua episode terkenal dalam sejarah peta dan pembuatan peta. Setiap negara memiliki lembaga pemetaan nasional yang didirikan bermuatan dengan menjaga agar peta nasional tetap akurat saat ini (misalnya, Survei Geologi AS dan Survei Ordnance Inggris).

Pada tanggal 1 Mei 2000. Presiden AS Bill Clinton mengumumkan penghapusan ketersediaan selektif dari sinyal GPS1 dan dengan demikian, disediakan akurasi yang jauh lebih baik untuk biaya sederhana dan murah,yaitu penerima GPS Secara praktis, ini berhasil mungkin untuk mendapatkan posisi penerima dengan akurasi 6 sampai 10 meter dalam kondisi normal,berbeda dengan kira-kira 100 meter sebelumnya. Upaya untuk mengembangkan lokasi berbasis layanan mendahului pemberitahuan dan berdasarkan informasi dari mobile tiang telepon atau beacon lainnya. Namun, metode ini tidak mendapatkan banyak pangsa pasar karena untuk kompleksitas teknis dan ketidakmampuan mereka untuk memberikan cakupan universal. Sebaliknya, GPS memungkinkan pengembangan receiver murah dengan akurasi posisi yang baik, dan, pada pertengahan-2001, adalah mungkin untuk membeli unit penerima sekitar US \$ 100,3 penerima ini membantu lebih banyak orang daripada sebelumnya mengumpulkan informasi tentang lokasi yang berbeda dan upload ke komputer mereka.

Namun sampai tahun 2002, ketika standar pertukaran (format eXchange GPS atau GPX) diterbitkan, dimanipulasi dan berbagi Informasi ini merupakan tugas yang rumit diperlukan komputasi dan manipulasi data pengetahuan. Untungnya, kebanyakan penerima GPS pengembang dengan cepat mengadopsi standar GPX,dan, pada tahun 2004, sudah menjadi hal biasa ketersediaan lokasi berkualitas tinggi informasi telah memungkinkan pemetaan mass market berdasarkan penerima GPS terjangkau, komputer rumah dan internet [44].

18.1.4 Desain Ekstraksi Data WMTS

Mengekstraksi data deret waktu dari WMTS terdiri dari empat langkah utama:

18.1.4.1 langkah-langkah

1. (1) membangun alamat pengeluaran sumber daya seragam (URL) dari gambar ubin yang tumpang tindih pada titik waktu tertentu,
2. (2) mendownload gambar ubin,
3. (3) temukan koordinat piksel yang sesuai dengan titik kepentingan di dalam gambar yang didownload,
4. (4) ubah warna piksel atau indeks warna piksel ke unit variabel yang teramat.

Keempat langkah ini diulang untuk setiap langkah waktu dalam deret waktu (Gambar 1).

Spesifikasi WMTS, yang juga dikenal dengan nama “TMS” (layanan peta ubin), adalah protokol layanan web untuk mengambil gambar peta ubin yang telah didefinisikan oleh OGC (Maso et al., 2010). Permintaan WMTS ada dalam format:

$$[server] = [layer] = [time] = [x] = [y] = [zoom].png \quad (18.1)$$

- (1)
dimana x adalah indeks x (horizontal) ubin dan y adalah indeks genteng (vertikal).

Zoom adalah angka (biasanya antara 1 dan 20) yang menunjukkan skala peta. Nomor ini tidak boleh lebih tinggi dari tingkat zoom yang didukung maksimum dari layanan web WMTS. Mengingat garis lintang, bujur, dan zoom, indeks x ubin dan genteng Indeks y dapat dihitung sebagai:

$$xtile = \frac{lon + 180}{360} 2^z oom \quad (18.2)$$

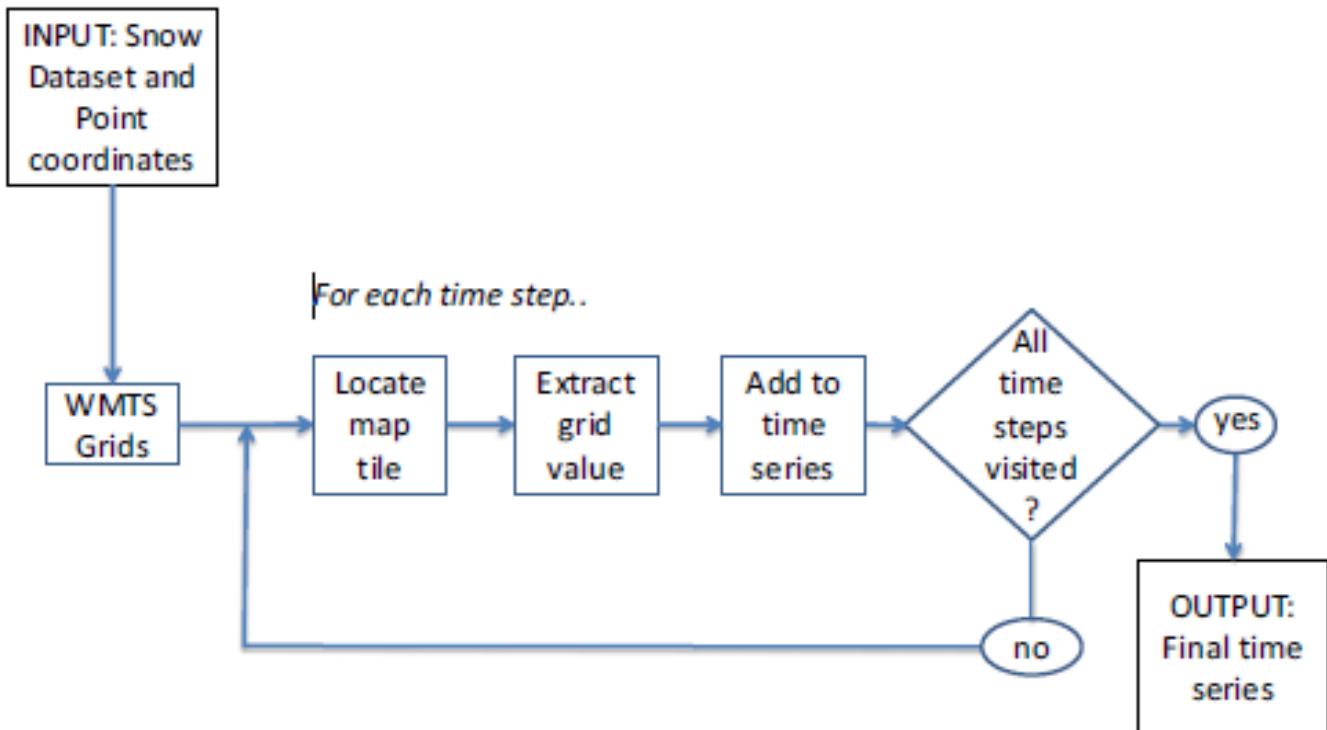
$$ytile = \frac{1 - log(tan(lat_{rad}) + \frac{1}{cos(lat_{rad})})}{2\pi} 2^z oom \quad (18.3)$$

18.1.5 Kerangka Tethys

Untuk membuat fungsi ekstraksi data Python tersedia bagi pengguna akhir, kami merancang aplikasi web interaktif dalam kerangka kerja Tethys. Tethys (Jones et al., 2014) adalah platform yang dapat digunakan untuk mengembangkan dan meng-host aplikasi web sumber daya air. Ini mencakup perangkat lunak open source dan open source: PostgreSQL, GeoServer, dan Web Processing Service. Salah satu keuntungan dari Tethys atas aplikasi web kustom adalah menyediakan arsitektur plug-in (Heinemann dan Councill, 2001) untuk penerapan aplikasi baru yang relatif mudah. Gaya grafis, tata letak, panel navigasi, pengelolaan pengguna, dan pengelolaan penyimpanan data web semuanya ditangani oleh Tethys. Tethys plugin dikenal sebagai “app” dan memiliki empat komponen: Persistent Storage, Controller, Template, dan Gizmo. Penyimpanan persisten digunakan untuk menyimpan pengaturan pengguna, misalnya lokasi favorit. Pengendali menangani perhitungan dan logika bisnis aplikasi. Tethys menyediakan pengendali dasar dengan fungsi built-in untuk menghubungkan ke model. Template mendefinisikan posisi teks, tombol, grafik, tabel, dan peta di antarmuka pengguna. Gizmo adalah komponen antarmuka pengguna yang dapat disesuaikan. Bisa berupa button, date time picker, daftar drop-down, chart, table, atau map. Peta dan bagan template bertanggung jawab untuk menampilkan peta dan bagan interaktif di halaman arahan aplikasi [42].

18.1.6 Desain Antarmuka Pengguna Inspektor Salju

Antarmuka pengguna web Snow Inspector terdiri dari dua tampilan: ‘peta salju’ dan ‘grafik salju’ . Peta menggunakan kontrol peta interaktif OpenLayers versi 3 (openlayers.org). Pengguna dapat menambahkan titik di manapun pada peta dengan klik mouse.

**Gambar 18.2** Snow Retrieval Function Loop for Point Location

Saat pengguna selesai bekerja dengan peta, koordinat bentuk yang ditentukan pengguna dilewaskan ke pengontrol peta. Pengontrol peta melewati koordinat dan parameter masukan lainnya (rentang waktu, layanan WMTS) ke pengendali salju. Pengendali salju meluncurkan skrip pengambilan data. Saat skrip pengambilan data berfungsi, ia memiliki opsi untuk menyimpan data yang diunduh dalam cache waktu penyimpanan yang terus-menerus. Dengan menggunakan metode Asynchronous JavaScript (AJAX), peta dan bagan template secara berkala memeriksa pengendali salju untuk status kemajuan dan memperbarui diagram dan bilah kemajuan di halaman web. Setelah pemrosesan selesai, pengendali mengambil rangkaian waktu dan meneruskannya ke bagan tempat grafik interaktif diperbarui [42].

18.1.7 Salju Inspektor Data Application Programmer Interface Design

Template lainnya digunakan untuk menampilkan rangkaian waktu hasil dalam empat format yang berbeda: comma-separated values (CSV), JavaScript Object Notification System (JSON), WaterML 1.1, dan WaterML 2.0. WaterML adalah standar OGC yang disetujui secara internasional untuk pertukaran data rangkaian waktu hidrologi (OGC, 2012a). Ini tidak hanya berisi data nilai tetapi juga metadata terkait termasuk lokasi situs, organisasi sumber data, dan unit pengukuran. Keseluruhan desain aplikasi web Snow Inspector termasuk pengontrol dan template. API time series dapat diakses dengan menggunakan parameter string [42].

DATA GEOSPASIAL WEB MAP SERVICE

19.1 Deskripsi WMS

Web Map Service adalah salah satu jenis penggambaran OGC Layanan model layanan web dan ia menyediakan platform multi-interoperability. Karya ini menghadirkan sebuah metode untuk menerapkan layanan peta web OGC berdasarkan teknik Layanan Web dan memperkenalkan proses terperinci.

Web Map Service (wms) memberikan informasi kepada pengguna internet oleh tata ruang peta gambar. Umumnya, yang tersimpan di dalam tata ruang data tersebut data vektor itu adalah panjang untuk menciptakan data vektor peta gambar. setiap sub-rectangle dikirim ke sebuah wms sub-maps node untuk menghasilkan sekumpulan peta yang dihasilkan merekonstruksi dengan semuanya sub-maps. Semua sub-maps dihasilkan di paralel, jadi makin sedikit waktu seluruh yang habis memproduksi sebuah peta.

Web Map Service (WMS) memberikan informasi kepada pengguna internet oleh tata ruang peta gambar. Umumnya, yang tersimpan di dalam tata ruang data tersebut data vektor. Itu adalah panjang ayub untuk menciptakan data vektor peta gambar dari Biaya untuk mengurangi waktu, maka kami memanfaatkan linux cluster. Ketika peta diminta, itu adalah suatu koordinat lingkup didefinisikan dengan xmin persegi panjang, ymin, xmax, ymax harus dispesifikasi. Kami merancang beban keseimbangan menurut algoritma untuk membagi permintaan ke dalam beberapa sub-rectangles.

Setiap sub-rectangle dikirim ke sebuah wms sub-maps node untuk menghasilkan sekumpulan . Peta yang dihasilkan akan merekonstruksi dengan semuanya ini sub-maps . Semua ini sub-maps dihasilkan di paralel, jadi makin sedikit waktu yang seluruh habis di memproduksi sebuah peta . Bagaimana untuk membagi menurut adalah kunci masalah permintaan . Pertama , kami menghadirkan metode untuk menghitung tingkat 2d bobot beban distribusi peta lingkup . Kedua , node beban kemampuan yang dievaluasi . Kemudian , penulis hadirmetode menurut untuk membelah seluruh. Kertas ini juga membahas algoritma kinerja pelaksanaan .

19.1.1 Pengenalan peta map service

Pengenalan Peta map Service (WM)s menghasilkan peta secara spasial direferensikan data dari informasi geografis secara dinamis. Standar internasional ini mendefinisikan sebuah ‘peta’ untuk menjadi penggambaran tentang informasi geografis sebagai file gambar digital cocok untuk ditampilkan pada layar komputer. Peta tidak data itu sendiri. Wms-dihadirkan maps umumnya diterjemahkan dalam format bergambar seperti PNG,GIF atau JPEG,atau kadang-kadang sebagai elemen-elemen grafis berbasis vektor dalam Scalable Vector Grafis (SVG) atau Komputer Web Metafile Grafis (WebCGM format).

WMS pelaksanaannya dengan Membuka Geospatial Konsorsium Peta Web Service (WMS menentukan spesifikasi) untuk grid-based multidimensi data lingkungan. WMS dapat membaca data dalam jumlah besar format data ilmiah umum - khususnya format NetCDF dengan perubahan iklim dan konvensi Perkiraan kemudian efisien menghasilkan gambar peta di ribuan mengkoordinir sistem referensi. Ia dirancang untuk memerlukan konfigurasi minimal dari administrator sistem pada saat digunakan dengan alat bantu klien yang sesuai, menyediakan pengguna akhir dengan cara interaktif untuk memvisualisasikan data tanpa perlu mendownload file besar atau menafsirkan meta data kompleks.

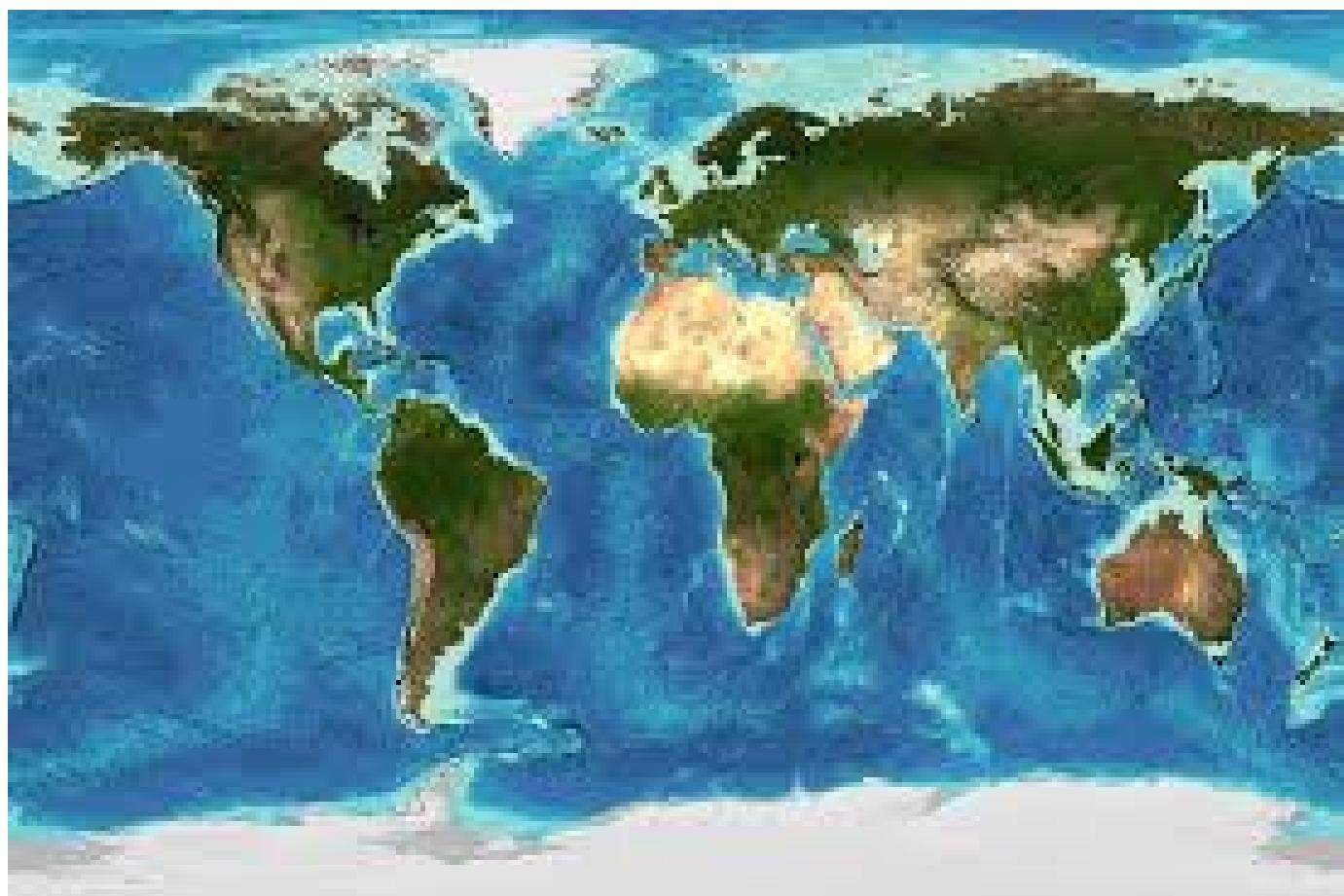
Data WMS biasanya digunakan untuk membandingkan data yang sudah kita buat dengan data yang dibuat oleh pihak lain, atau juga sering kali digunakan sebagai pelengkap dalam pembuatan peta.

19.1.2 Interoperability antara WM dan grid data lingkungan

Dalam WMS, unit penting informasi merupakan lapisan. Setiap lapisan dapat ditampilkan dalam jumlah Gaya, setiap dikaitkan dengan sebuah legenda. Lapisan mungkin yang dapat ditampilkan atau non-yang dapat ditampilkan dan bisa diatur hierarchically. Tiga operasi utama dapat dilakukan oleh klien WM menentukan standar: Permintaan GetCapabilities dokumen XML berisi metadata tersedia pada lapisan dan kemampuan layanan lain; permintaan GetMap gambar peta animasi atau sesuai dengan pilihan pengguna Layer, Style, sejauh geografis dan resolusi dan permintaan GetFeatureInfo informasi lebih lanjut tentang lokasi geografis tertentu, yang diwakili oleh piksel tertentu dalam gambar peta.

Disini menjelaskan ncWMS, sebuah implementasi dari spesifikasi Web Map Service (WMS) geospasial untuk data lingkungan gridded multidimensional. ncWMS dapat membaca data dalam sejumlah besar format data ilmiah - terutama format CDS Bersih dengan konvensi iklim dan perkiraan - kemudian secara efisien menghasilkan citra peta dalam ribuan sistem referensi koordinat yang berbeda.

Tujuan ncWMS ia adalah untuk dapat menghasilkan beberapa jenis visualisasi berbeda (maps, transects bagian vertikal, dsb.) secara efisien data dari diselenggarakan dalam format file yang berbeda dan mengkoordinir sistem referensi (CRSs).



Gambar 19.1 Menjeleskan tentang Interoperability antara WMS dan grid data lingkungan

19.2 Mengimplementasikan NcWMS sebagai aplikasi Web Java

NcWMS diimplementasikan sebagai aplikasi web Java, dikemas sebagai arsip web (WAR) untuk file yang standar penempatan wadah servlet seperti Apache Tomcat. Java telah dipilih sebagai bahasa menerapkan karena banyak penyedia data lingkungan sudah menggunakan teknologi server-side Java dan karena ketersediaan berkualitas tinggi dan perpustakaan yang kuat seperti Java-NetCDF.

Banyak WMS client, termasuk Godiva2 client yang dibundel dengan ncWMS, adalah tidak hanya berasal dari client yang membuat permintaan GetMap menggunakan nomor yang terbatas dari kotak pengikat yang telah diperbaiki. Hal ini akan meningkatkan skalabilitas dari sistem secara keseluruhan dengan mengizinkan permintaan diulangi untuk gambar yang sama untuk dilayani dari cache level-aplikasi. Karena itu ncWMS menerapkan cache seperti (lihat diagram arsitektur dalam pohon ara. Catatan bahwa cache ini berpendapat array data, tidak gambar akhir: ini memungkinkan pengguna untuk mengubah parameter penata dari suatu gambar (misalnya Palet deep warna) tanpa re-ekstrak data dari file sumber, mempercepat visualisasi interaktif. Banyak administrator sistem memilih juga untuk menerapkan cache ubin di atas ncWMS, yang lebih jauh dapat meningkatkan performa dan skalabilitas.

ncwms biasanya menyajikan data spasial peta atau citra, yang akan menjadi layer saat visualisasi di browser. biasanya dengan melakukan request getmap, Getmap merupakan salah satu operasi wms, akan dihasilkan tampilan peta berupa layer yang melapisi

(overlay) peta pada dasar google maps. peta yang berferensi geografis dihasilkan dari sebuah web map service, dan peta ini biasanya disajikan dalam format gambar seperti PNG, GIF dll.

19.2.1 Menjelaskan tentang layanan WMS

Layanan web map service meningkatkan dengan adanya fisheye pandangan metode untuk cartographic data .Sementara banyak studi difokuskan pada pandangan fisheye,masalah yang banyak besar distorsi yang memetakan mereka semuanya memiliki lebih dari wilayah dan / atau high density jalan di perbatasan peta belum diatasi .Untuk sepenuhnya menghapus distorsi di kedua fokus dan konteks daerah.

19.3 Beberapa operasi wms pada tahap implementasi

WMS memiliki tiga buah operasi wms pada saat tahap implementasi yaitu : GetCapabilities merupakan deskripsi informasi yang dimiliki wms dan parameter permintaan yang dapat diterima, Getmap yaitu peta dengan parameter dimensi, Getfeature info yaitu meminta informasi mengenai fitur tertentu yang ditampilkan pada peta, Web Map Service (WMS) memberikan informasi kepada pengguna internet oleh tata ruang peta gambar.

19.3.1 WMS

WMS (Web Map Services) tidak hanya dapat menghasilkan gambar raster tetapi juga gambar vektor. Contohnya adalah SVGT (Scalable Vector Graphics Tiny) yang merupakan bagian dari SVG (Scalable Vector Graphics) yang digunakan pada perangkat mobile device. Mahalnya biaya komunikasi antara handheld device dengan jaringan internet melalui GPRS (Global Pocket Radio System) menimbulkan sebuah masalah dalam penerapan aplikasi mobile mapping. Untuk mengeliminasi masalah tersebut, harus dibuat sebuah aplikasi mobile mapping yang dapat mentransfer data sekecil mungkin.

Dengan bantuan teknologi web map service sejumlah besar layanan wms dikembangkan dan dirilis ke network. tidak untuk organik di antara peta layanan web, pengguna sangat kesulitan mendapatkan dan memanfaatkan services ini. satu jenis layanan web peta mesin pencari frame yang diajukan, termasuk url mesin pencari, kemampuan basis data dan kemampuan dokumen respon analyzer. Web map service merupakan suatu komponen software yang merupakan suatu komponen software yang merupakan self-containing.wms yang paling banyak diadopsi dan populer yang digunakan peta web service (WMS),spesifikasi yang menguraikan sebuah mekanisme komunikasi memungkinkan menguraikan produk perangkat lunak untuk meminta dan memberikan peta preassembled citra dikompilir gambar peta, yang mungkin berisi kedua kordinat dan data raster yang meminta klien

BAB 20

DATA GEOSPASIAL

20.1 Geospasial

20.1.1 Pengertian Geospasial

Geospasial terdiri dari dua kata, yaitu geo dan spasial. Geo berarti bumi sedangkan spasial berarti ruang. UU No 4 tahun 2011 tentang geospasial menyebutkan, spasial adalah aspek keruangan dari suatu objek, atau yang mencakup lokasi, letak, dan posisinya. Data geospasial dipecah menjadi dua, yaitu yang pertama; Data grafis atau geometri. Data ini terdiri dari tiga elemen : titik, garis, dan luasan. data data ini berbentuk dalam vektor maupun raster. yang kedua adalah data attribut atau data tematik.

20.2 Data Spasial

20.2.1 Definisi Data Spasial

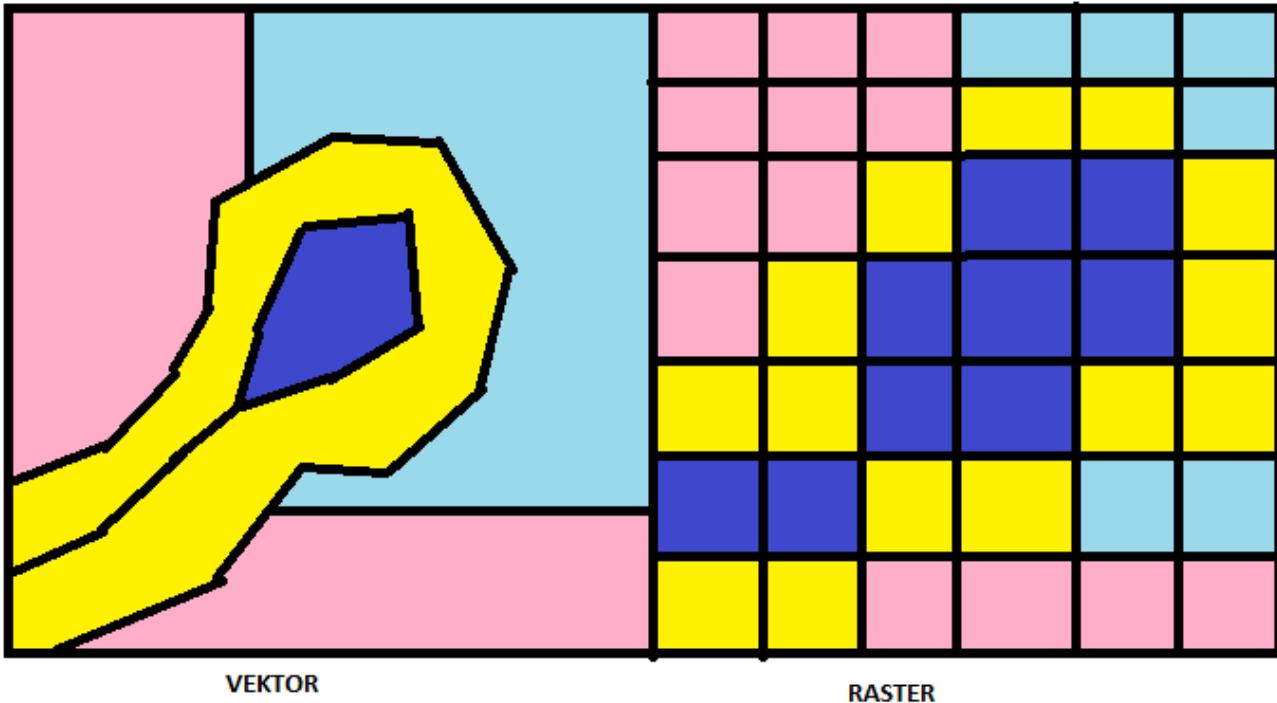
Data spasial adalah data yang berreferensi dari representasi objek objek yang ada di bumi. Data spasial umumnya berbentuk peta yang isinya interpretasi dan atau proyeksi seluruh fenomena yang ada di muka bumi. Fenomena tersebut berupa fenomena alami dan fenomena buatan manusia. Semua data yang ada dipeta adalah representasi obyek bumi. Data spasial di bagi menjadi dua tipe, yaitu model data vektor dan model data raster. Model data vektor menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan titik, garis, kurva atau poligon beserta elemennya. Model data raster menampilkan, dan menyimpan data spasial dengan menggunakan struktur matrikx atau pixel yang membentuk grid.

Pada gambar 20.1 merupakan contoh perbedaan antara vektor dengan raster.

20.3 Tipe Data Vektor

20.3.1 Definisi Tipe Data Vektor

Data vektor adalah data yang disimpan dalam bentuk koordinat titik yang menampilkan, menempatkan, dan menyimpan data spacial dengan menggunakan titik, garis atau polygon. Terdapat tiga jenis tipe data vektor yaitu titik, garis, dan polygon. Tipe data ini biasanya terdapat pada peta. Setiap bagian dari data vektor bisa saja mempunyai informasi yang berasosiasi satu sama lain. Model data vektor diwakili oleh simbol simbol yang terdiri atas interkoneksi garis dan titik yang merepresentasikan lokasi dan garis batas dari entitas geografi. Dalam model data vektor , data dapat direpresentasikan sebagai Lines (garis), Polylines (polygon), Points (titik), Area (daerah), Nodes (titik potong) Pada model data vektor ini, suatu objek dinyatakan dalam bentuk koordinat(x,y) yang berhubungan satu sama lainnya kecuali objek titik. Pada gambar 20.2 merupakan salah satu contoh data vektor line pada peta yaitu sungai.



Gambar 20.1 Perbedaan Vektor dengan Raster.



Gambar 20.2 Sungai merupakan contoh vektor line pada peta.

20.4 Data Atribut

20.4.1 pengertian Data Atribut

Data atribut merupakan data yang mempresentasikan aspek-aspek deskripsi/penjelasan dari suatu fenomena di permukaan bumi dalam bentuk kata-kata, angka, atau tabel. Data atribut berfungsi untuk menggambarkan gejala topografi karena memiliki aspek deskriptif dan kualitatif. Oleh karena itu, data atribut sangat penting dalam menjelaskan seluruh objek geografi. Contohnya, atribut kualitas tanah terdiri atas status kepemilikan lahan, luas lahan, tingkat kesuburan tanah dan kandungan mineral dalam tanah. Data atribut bisa berupa data kuantitatif (angka) seperti data jumlah penduduk dan dapat berupa data kualitatif (mutu) seperti data tingkat kesuburan tanah. Bentuk-bentuk data atribut:

1. Data kuantitatif (angka-angka/statistik), contoh: jumlah penduduk

2. Data kualitatif (kualitas/mutu), contoh: tingkat kesuburan tanah

Kelebihan dan Kekurangan Data Atribut :

Kelebihan :

1. Data dapat dimanipulasi

2. Dapat mengetahui jenis lokasi peta

Kekurangan :

1. Sulit membedakan gambar dan warna pada peta

2. Sulit menentukan lokasi

20.5 Data Vektor Line

20.5.1 Pengertian

Line merupakan bahasa Inggris dari garis. Garis adalah bentuk geometri liniar yang menghubungkan dua titik atau lebih dan biasanya digunakan untuk mempresentasikan object berdimensi satu. Batas object geometri polygon juga merupakan sebuah garis-garis, begitu pula dengan jaringan listrik, jaringan komunikasi, jaringan air minum, saluran buangan, dan utility lain yang dapat dipresentasikan sebagai object dengan bentuk geometri garis. Hal itu pula yang akan bergantung pada skala peta yang menjadi sumbernya atau skala representasi akhirnya. Garis bisa digunakan untuk menunjukkan route suatu perjalanan atau menggambarkan boundary. Poligon bisa digunakan untuk menggambarkan sebuah danau atau sebuah Negara pada peta dunia. Dalam format vektor, bumi direpresentasikan sebagai suatu mosaik dari garis (arc/line), poligon (daerah yang dibatasi oleh garis yang berawal dan berakhir pada titik yang sama), titik/ point (node yang mempunyai label), dan nodes (merupakan titik perpotongan antara dua baris). Seperti telah diuraikan sebelumnya, data vektor terbentuk dari tiga jenis geometri yakni titik (point), garis (line), dan area (polygon). Oleh karena itu, objek-objek di permukaan bumi perlu divisualisasikan dalam ketiga geometri tersebut agar bisa diproses dengan GIS. Contoh visualisasi dunia nyata menjadi elemen gambar ketiga geometri tersebut antara lain landmark dan fasilitas sebagai titik, jalan dan sungai sebagai garis, dan daerah administrasi tertentu sebagai area. Berikut ini penjelasan lebih dalam mengenai ketiga entitas geometri tersebut.

20.5.1.1 titik (point) meliputi semua objek grafis atau geografis yang dikaitkan dengan pasangan koordinat (x,y). Selain memuat informasi koordinat, data titik juga bisa saja merupakan suatu simbol yang memiliki keterkaitan dengan informasi lain. Satu buah objek titik memiliki satu baris dalam tabel atribut. Karakteristik-karakteristik dari titik ini dijelaskan oleh kolom-kolom yang dibentuk pada tabel atribut.

20.5.1.2 Garis (line) merupakan semua unsur-unsur linier yang dibangun dengan menggunakan segmen-segmen garis lurus yang dibentuk oleh dua titik koordinat atau lebih (Burrough, 1994). Entitas garis yang paling sederhana memerlukan ruang untuk menyimpan titik awal dan titik akhir (dua pasangan koordinat x,y) berserta informasi lain mengenai simbol yang digunakan untuk merepresentasikannya. Garis tunggal yang terbentuk dari titik awal dan titik akhir saja disebut sebagai line. Sedangkan garis bersegmen banyak yang terbentuk dari banyak titik (vertex) disebut polyline. Dalam GIS, baik line maupun polyline dianggap sebagai suatu entitas yang sama yakni polyline. Setiap satu entitas polyline memiliki satu baris dalam tabel atribut. Karakteristik dari entitas ini disimpan dalam kolom-kolom tabel atribut.

20.5.1.3 Area (polygon) merupakan suatu objek tertutup yang memiliki luasan. Polygon dapat direpresentasikan dengan berbagai cara di dalam model data vektor. Karena kebanyakan peta tematik yang digunakan dalam GIS berurusan dengan polygon, metode-metode representasi dan pemanipulasi entity ini banyak mendapat perhatian. Seperti halnya titik dan polyline, satu objek poligon juga diwakili oleh satu baris pada tabel atribut. Poligon biasanya digunakan untuk merepresentasikan objek dunia nyata yang memiliki luasan seperti wilayah administrasi, danau, guna lahan, jenis tanah, dan sebagainya

20.5.1.4 Kelebihan dan Kekurangan Data Vektor Line

Kelebihan:

1. Memerlukan ruang atau tempat menyimpan yang lebih sedikit di computer.

2. Satu layer dapat dikaitkan dengan atau menggunakan atribut sehingga dapat menghemat ruang penyimpanan secara keseluruhan.

3. Dengan banyak atribut yang banyak dikandung oleh satu layer, banyak peta tematik lain yang dapat dihasilkan sebagai peta turunannya.

4. Hubungan topologi dan network dapat dilakukan dengan mudah.

5. Memiliki resolusi spasial yang tinggi.

6. Representasi grafis data spasialnya sangat mirip dengan peta garis buatan tangan manusia.

7. Memiliki batas-batas yang teliti, tegas dan jelas sehingga sangat baik untuk pembuatan peta-peta administrasi dan persil tanah milik.

8. Transformasi koordinat dan proyeksi tidak sulit dilakukan.

Kekurangan:

1. Memiliki struktur data yang kompleks.
2. Datanya tidak mudah untuk dimanipulasi.
3. Pengguna tidak mudah berkreasi untuk membuat programnya sendiri untuk memenuhi kebutuhan aplikasinya. Hal ini disebabkan oleh struktur data vektor yang lebih kompleks dan prosedur fungsi dan analisisnya memerlukan kemampuan tinggi karena lebih sulit. Pengguna harus membeli sistem perangkat lunaknya karena teknologinya masih mahal. Prosedurnya pun terkadang lebih sulit.
4. Karena proses keseluruhan untuk mendapatkannya lebih lama, peta vektor sering kali mengalami out of date atau kadaluarsa.
5. Memerlukan perangkat keras dan perangkat lunak yang lebih mahal.
6. Overlay beberapa layers vektor secara simultan memerlukan waktu yang relatif lama.

20.6 Raster

20.6.1 Pengertian Raster

Data raster (juga dikenal sebagai data grid) mewakili tipe keempat dari fitur: permukaan. Data raster berbasis sel dan kategori data ini juga mencakup citra udara dan satelit. Ada dua jenis data raster: kontinu dan diskrit. Contoh data raster diskrit adalah kepadatan penduduk. Contoh data kontinyu adalah pengukuran suhu dan elevasi. Ada juga tiga jenis dataset raster: data tematik, data spektral, dan gambar. Raster dataset adalah intrinsik untuk analisis spasial yang paling. Analisis data seperti ekstraksi kemiringan dan aspek dari Digital Elevation Models terjadi dengan dataset raster. Pemodelan hidrologi spasial seperti ekstraksi daerah aliran sungai dan jalur aliran juga menggunakan sistem berbasis raster. Data spektral menyajikan citra udara atau satelit yang kemudian sering digunakan untuk memperoleh informasi geologi vegetasi dengan mengklasifikasikan tanda tangan spektral dari setiap jenis fitur.

20.6.1.1 Karakteristik Raster Resolusi suatu data raster akan merujuk pada ukuran permukaan bumi yang direpresentasikan oleh setiap piksel. Makin kecil ukuran atau luas permukaan bumi yang dapat direpresentasikan oleh setiap pikselnya, makin tinggi resolusi spasialnya. Piksel-piksel di dalam zone atau area yang sejenis memiliki nilai (isi piksel atau ID number) yang sama. Pada umumnya, lokasi di dalam model data raster, diidentifikasi dengan menggunakan pasangan koordinat kolom dan baris (x,y). Nilai yang merepresentasikan suatu piksel dapat dihasilkan dengan cara sampling yang berlainan:

1. Nilai suatu piksel merupakan nilai rata-rata sampling untuk wilayah yang direpresentasikannya.
2. Nilai suatu piksel adalah nilai sampling yang berposisi di pusat (atau di tengah) piksel yang bersangkutan.
3. Nilai suatu piksel adalah nilai sample yang tertetap di sudut-sudut grid.

20.6.1.2 Kelebihan dan Kekurangan Raster

Kelebihan :

1. Memiliki struktur data yang sederhana
2. Mudah dimanipulasi dengan menggunakan fungsi-fungsi matematis sederhana
3. Teknologi yang digunakan cukup murah dan tidak begitu kompleks sehingga pengguna dapat membuat sendiri program aplikasi yang menggunakan citra raster.
4. Compatible dengan citra-citra satelit penginderaan jauh dan semua image hasil scanning data spasial.
5. Overlay dan kombinasi data raster dengan data inderaja mudah dilakukan
6. Memiliki kemampuan-kemampuan permodelan dan analisis spasial tingkat lanjut
7. Metode untuk mendapatkan citra raster lebih mudah
8. Gambaran permukaan bumi dalam bentuk citra raster yang didapat dari radar atau satelit penginderaan jauh selalu lebih aktual dari pada bentuk vektornya
9. Prosedur untuk memperoleh data dalam bentuk raster lebih mudah, sederhana dan murah.
10. Harga system perangkat lunak aplikasinya cenderung lebih murah.

Kekurangan:

1. Secara umum memerlukan ruang atau tempat menyimpan (disk) yang besar dalam computer, banyak terjadi redundancy data baik untuk setiap layer-nya maupun secara keseluruhan.

2. Penggunaan sel atau ukuran grid yang lebih besar untuk menghemat ruang penyimpanan akan menyebabkan kehilangan informasi dan ketelitian.
3. Sebuah citra raster hanya mengandung satu tematik saja sehingga sulit digabungkan dengan atribut-atribut lainnya dalam satu layer.
4. Tampilan atau representasi dan akurasi posisi sangat bergantung pada ukuran pikselnya (resolusi spasial).
5. Sering mengalami kesalahan dalam menggambarkan bentuk dan garis batas suatu objek, sangat bergantung pada resolusi spasial dan toleransi yang diberikan.
6. Transformasi koordinat dan proyeksi lebih sulit dilakukan
7. Sangat sulit untuk merepresentasikan hubungan topologi (juga network).
8. Metode untuk mendapatkan format data vektor melalui proses yang lama, cukup melelahkan dan relatif mahal.

DATA GEOSPASIAL SHAPEFILE

21.1 Shapefile

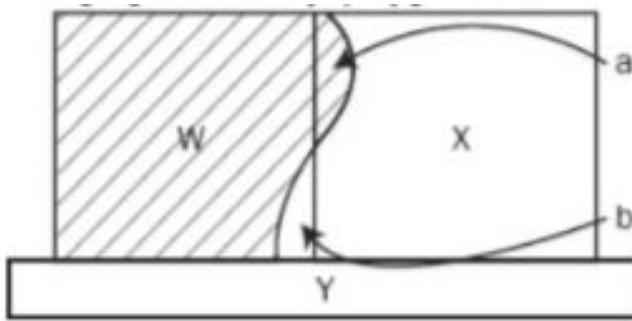
21.1.1 Pengertian Shapefile

Shapefile ArcView memiliki format data tersendiri yang disebut dengan shapefiles. Shapefiles adalah format data yang menyimpan lokasi geometrik dan informasi atribut dari suatu feature geografis. Pada umumnya kita hanya butuh satu file kerja seperti file Microsoft Word dengan extension file .doc, akan tetapi shapefile memiliki perbedaan, yaitu bahwa satu shapefile memiliki beberapa file yang saling berkaitan satu sama lainnya. Beberapa file ini memiliki extension yang 42 berbeda-beda yang disimpan dalam workspace yang sama. Catatan : tiga file extension pertama adalah bagian file extension yang harus ada dalam sebuah shapefile, file extension berikutnya sifatnya optional. [45] Fitur geografis di shapefile dapat ditunjukkan oleh titik, garis, atau poligon (area). Ruang kerja yang berisi shapefile mungkin juga berisi table dBase, yang dapat menyimpan atribut tambahan yang dapat digabungkan ke fitur shapefile. Semua file yang memiliki ekstensi seperti file .txt, .asc, .csv atau tab muncul di ArcCatalog sebagai file text secara default. Akan tetapi, pada kotak dialog Opsi Kita dapat memilih tipe file mana yang harus direpresentasikan sebagai file teks dan seharusnya tidak ditampilkan di pohon Catalog. Ketika file teks berisi nilai koma dan tab-delimited, kita bisa melihat isi file di tampilan table ArcCatalog dan menggabungkannya ke dalam fitur geografis. file teks bisa juga kita hapus, tetapi isinya hanya bisa dibaca di ArcCatalog. Shapefile adalah seperangkat file komputer yang digunakan untuk menyimpan informasi geografis (mis., Batas saluran sensus) dan tabel atribut yang terkait dengan informasi geografis (mis., Perumahan sensus dan karakteristik demografis). Shapefiles dapat dimanipulasi menggunakan sistem informasi geografis (SIG); ArcView 8.3 (ESRI, Redlands, CA) digunakan dalam proyek ini. Paket pajak shapefile diperoleh dari kantor penilai pajak Fulton dan Gwinnett County. Shapefile ini mengandung poligon yang sesuai dengan lokasi dan dimensi setiap paket tanah kena pajak di county. Alamat setiap paket disimpan dalam tabel atributnya. Shapefile ESRI atau biasa disebut shapefile adalah format data geospasial yang umum untuk perangkat lunak sistem informasi geografis, dengan pengertian bahwa shape merupakan properti intrinsik utama untuk sistem visual manusia. manusia lebih sering mengasosialisasikan objek dengan bentuknya ketimbang elemen lainnya (warna misalnya), pada umumnya, citra yang dibentuk oleh mata merupakan citra dwimatra (2 dimensi), sedangkan objek yang dilihat umumnya berbentuk trimatra (3 dimensi). informasi bentuk objek dapat diekstansi dari citra pada permulaan pra-pengolahan dan segmentasi citra. salah satu tantangan utama pada komputer vision adalah merepresentasikan bentuk, atau aspek-aspek penting dari bentuk.

21.1.2 Struktur Data Shapefile

Geodatabase adalah struktur data yang kuat dan canggih. selain topologi yang anda dapatkan secara gratis, area dan sekeliling area yang digambarkan dengan fitur linier. Namun, ESRI juga mendukung struktur data yang jauh lebih rumit: shapefile. sebuah shapefile dapat menggabungkan dua elemen penting yang dimiliki oleh geodatabase(komponen geografis dan database atribut) Perangkat lunak basis data adalah sistem manajemen basis data yang dinamai dBASE. Shapefile tertentu dibatasi hanya untuk mewakili satu dari jenis berikut: titik, multipoint, polyLines, atau poligon dengan titik. masing-masing titik memiliki catatan database relasional. Jika sejumlah titik dianggap objek yang sama, maka objek tersebut hanya memiliki satu record di tabel atribut. seperti pada geodatabases, polylines dapat disusun dari satu atau lebih jalur terhubung ataupun terputus-putus. Namun, jalur diperbolehkan untuk disusun hanya dari segmen garis lurus. Poligon dalam shapefile memiliki kemiripan dengan poligon

basis geodata, namun tidak ada topologi yang ada dan tidak ada yang dapat diciptakan. setiap poligon berdiri sendiri. Hal ini digambarkan secara lengkap oleh satu entitas linier: urutan segmen yang dimulai di satu lokasi geografis dan kembali ke lokasi tersebut. mungkin ada poligon yang berdekatan atau tidak. poligon lain mungkin tumpang tindih. Lihat gambar berikut : Gambar



Gambar 21.1 Gambar Shapefile Poligon

silabus shapefile menggambarkan masalah dengan tumpang tindih dan kesenjangan. (Poligon W memiliki poligon batas kanan melengkung, X memiliki batas kiri lurus)21.1 Masalah dimana shapefiles sangat rentan adalah mungkin ada potongan tumpang tindih atau kekosongan-kekosongan (gap) antara poligon yang dimaksud, dan tidak ada yang dapat anda lakukan jika data tetap dalam format shapefile. Pada gambar diatas sliver ádiklaim oleh W dan X, sedangkan sliver btidak ada pada keduanya. Area geografis yang dipartisi menjadi poligon shapefile yang saling menggairahkan akan memiliki informasi duplikat. karena batas total masing-masing poligon didefinisikan untuk poligon itu, setiap garis umum adalah digital ganda. Selanjutnya, ada dua batasan independen, dan anda tidak memiliki jaminan bahwa mereka kongruen. Dengan geodatabases, anda dapat membuat peraturan topologi untuk memastikan bahwa poligon tidak tumpang tindih atau memiliki celah, namun tidak dengan shapefile, anda tidak mendapatkan area dari perimeter sebagai atribut. Keuntungan dari representasi shapefile adalah kesederhanaan, kecepatan pemrosesan, kecepatan menggambar, dan biasanya, ekonomi penyimpanan. shapefiles berguna bila anda tidak memerlukan geoprocessing yang canggih. Sadarilah bahwa banyak kumpulan data GIS telah dimasukkan ke dalam format shapefile. Mungkin ada banyak konversi ke format geodatabase di masa depan anda jika anda ingin menggunakan kumpulan data tersebut di geoprocessing.[46]

21.1.3 Daftar beberapa file extension

1. shp - File yang menyimpan feature geometri (diperlukan dalam sebuah shapefile)
2. shx - File yang menyimpan index dari feature geometri (diperlukan dalam sebuah shapefile)
3. dbf - File dBASE yang menyimpan informasi atribut dari suatu feature (diperlukan dalam sebuah shapefile)
4. sbn dan sbx File yang menyimpan spatial index dari feature (optional)
5. fbn dan fbx File yang menyimpan spatial index dari feature shapefile yang read-only (optional)
6. ain dan aih File yang menyimpan index atribut dari field yang aktif dalam sebuah tabel (optional)
7. prj - File yang menyimpan informasi koordinat dari sebuah shapefile, file ini dapat muncul jika kita menggunakan ArcView Projection Utility (optional). [45]

21.1.3.1 Contoh penggunaan extension shapefile

1. read.shapefile (form.name)
2. read.shp (shp.name)
3. read.shx (shx.name)
4. read.dbf (dbf.name, sundulan = FALSE)
5. write.shapefile (shapefile, out.name, arcgis = FALSE)
6. write.shp (shp, out.name)
7. write.shx (SHX, out.name)
8. write.dbf (DBF, out.name, arcgis = FALSE)
9. calc.header (shapefile)
10. add.xy (shapefile)

21.1.3.2 Bentuk file

- scaleXY (shapefile, scale.factor)
- convert.to.shapefile (shpTable, attTable, bidang, jenis)
- convert.to.simple (shp)
- change.id (shpTable, newFieldAsVector)
- dp (poin, toleransi)

21.1.4 Komponen Teknis

Komponen yang ada pada sebuah aplikasi GIS mempunyai fungsi utama untuk membaca dan menulis data spasial, baik yang tersimpan dalam sebuah shapefile (*.shp) atau tersimpan ke dalam sebuah database. Dalam MapServer yang sudah berjalan ada beberapa Komponen utama yang digunakan secara penuh untuk menjalankan Aplikasi GIS untuk menangani data spasial baik yang tersimpan dalam sebuah flat file atau juga dalam DBMS yaitu : 1. SHAPELIB Shapelib merupakan library yang ditulis menggunakan bahasa pemrograman C yang digunakan untuk melakukan proses read terhadap Shapefile (*.shp) yang sudah didefinisikan ESRI (Environmental System Research Institute). Format dalam shapefile umum digunakan untuk menyimpan data vektor simple (tanpa topologi) dengan atribut, shapefile merupakan format data default yang digunakan dalam GIS.

21.1.5 Deskripsi Teknis Shapefile

SDE, ARC / INFO, PC ARC / INFO, Data Otomasi Kit (DAK), dan ArcCAD Perangkat lunak menyediakan penerjemah data form-to-coverage, dan ARC / INFO juga menyediakan a penerjemah coverage-to-shape. Untuk pertukaran dengan format data lainnya, shapefile spesifikasi diterbitkan dalam makalah ini. Aliran data lainnya, seperti yang berasal dari global receiver positioning system (GPS), juga dapat disimpan sebagai shapefile atau X, Y event tables.

21.1.6 Mengapa Harus menggunakan Shapefile

Sebuah shapefile menyimpan geometri nontopologis dan informasi atribut untuk ruang fitur dalam kumpulan data Geometri untuk fitur disimpan sebagai bentuk yang terdiri dari satu set koordinat vektor Karena shapefile tidak memiliki overhead pengolahan dari struktur data topologi, mereka memiliki kelebihan dibandingkan sumber data lain seperti kecepatan dan edit gambar yang lebih cepat kemampuan. Shapefiles menangani fitur tunggal yang tumpang tindih atau tidak bersebelahan. Mereka Juga biasanya membutuhkan lebih sedikit ruang disk dan lebih mudah dibaca dan ditulis. Shapefiles dapat mendukung fitur titik, garis, dan area. Fitur area direpresentasikan sebagai loop tertutup, poligon digital ganda. Atribut disimpan dalam file format dBASE. Setiap record atribut memiliki hubungan satu lawan satu dengan catatan bentuk yang terkait

21.1.7 Shapefile dapat dibuat dengan 4 Metode Umum

Ekspor Shapefiles dapat dibuat dengan mengekspor sumber data ke shapefile ARC / INFO, PC ARC / INFO, Spasial Database Engine (SDE), ArcView GIS, atau perangkat lunak Business MAP. Digitize Shapefiles dapat dibuat secara langsung dengan mendigitalkan bentuk menggunakan ArcView GIS alat pembuatan fitur Pemrograman Menggunakan Avenue (ArcView GIS), MapObjects, ARC Macro Bahasa (AML) (ARC / INFO), atau Simple Macro Language (SML) (PC ARC / INFO) perangkat lunak, Anda dapat membuat shapefile dalam program Anda. Menulis langsung ke spesifikasi shapefile dengan membuat sebuah program.

21.1.8 Jenis Numerik pada Shapefile

Sebuah shapefile menyimpan bilangan bulat dan bilangan presisi ganda. Sisa dokumen ini akan mengacu pada jenis berikut: Integer: Signed 32-bit integer (4 byte) Ganda: Signed 64-bit IEEE presisi ganda floating point nomor (8 byte) Nomor titik terapung harus berupa nilai numerik. Ketidakterbatasan positif, tak terhingga negatif, dan Nilai No-a-Number (NaN) tidak diperbolehkan dalam shapefile. Meski begitu, shapefiles dukung konsep nilai tidak ada data, namun saat ini hanya digunakan untuk pengukuran. Setiap bilangan floating point lebih kecil dari -10³⁸ dianggap oleh shapefile reader mewakili nilai tidak ada data.

21.1.9 Pengertian Shapefile

Shapefile adalah format non topologi yang secara efisien dan sederhana serta berfungsi sebagai wadah untuk penyimpanan lokasi geometric dan juga atribut informasi dari data geografis. Dan untuk melakukan suatu penambahan shapefile, maka kita dapat menggunakan python dan plugin pyshp. Shapefile merupakan file yang dapat menyimpan data vektor dalam arcview. Yang mana, Shapefile inilah yang kemudian diolah serta dianalisis dalam berbagai pekerjaan spasial dengan sebuah arcview. Maka saat ditampilkan dalam suatu lembar view, maka shapefile akan menjadi sebagai sebuah theme.

21.1.9.1 Cara Mengcreate Shapefile Menggunakan Python

1. Import shapefile
2. Memasukan variable dengan sebuah inisialisasi Projection Utility (optional).

21.1.9.2 Format yang Digunakan Dalam Mengcreate Data

1. SHP = dalam sebuah format SHP, terdapat 3 tipe shape file yaitu Point, Polyline dan Polygon.
2. DBF = dalam sebuah format DBF, ada 3 field yang digunakan. Yang mana, Field pertama berisikan atribut tabel, dan field kedua yang berisikan method yang dapat digunakan, serta field yang ketiga berfungsi untuk dapat menyimpan nama sebuah shapefile yang sebelumnya telah diinputkan. Dan data geospasial tersebut kemudian disimpan dengan menggunakan method a.save file.shp.

21.1.9.3 Cara Menambahkan Record

- Pada Point = `a.point(x, y)` ataupun `a.point(x, y, 0, 0)` dan dengan domain x dan y merupakan koordinat.
- Pada Polyline = `a.poly(shapefile = 3, parts = [[[x1, y1, z1, w1], [x2, y2, z2, w2], []]])`
- Pada Polygon = `a.poly(shapefile = 5, parts = [[[.], [.]]])`

21.1.9.4 Tipe Field Pada Atribut Shapefile

Berikut ini merupakan tipe-tipe FIELD yang dapat disematkan atau digunakan pada Shapefile, diantaranya yaitu:

1. TEXT Field ini dapat digunakan pada semua karakter dan berfungsi sebagai teks dengan rentang jumlah karakter yaitu 1 s.d. 255. Dan tipe field ini juga digunakan untuk sebuah nilai teks, contohnya seperti nama tempat/anotasi ataupun label. Selain itu, text dapat juga berisi angka, namun angka tersebut harus tetap dianggap sebagai teks yang tidak bisa dilakukan operasi aritmatika (tambah, kurang, bagi, kali, dsb).
2. FLOAT 4 bytes; Angka pecahan dengan sebuah rentang luas antara +/-3,438. Float merupakan tipe data angka dengan presisi tunggal yang juga memiliki pecahan. Yang mana, jumlah presisi atau biasa disebut jumlah angka, dapat juga ditentukan dengan konstan.
3. DOUBLE 8 bytes; Double yaitu tipe data angka yaitu dengan presisi ganda yang memiliki pecahan. Tipe Double mempunyai kesamaan dengan tipe float, yaitu hanya memiliki presisi yang lebih tinggi. Dan untuk data yang berisi angka-angka desimal sangat penting, misalnya angka luasan ataupun rupiah, dengan tipe double yang paling sesuai.
4. SHORT 2 bytes; Short integer dapat digunakan untuk angka tanpa pecahan. Dan Short integer juga biasa digunakan untuk sebuah data berupa ID, nomor, urutan, dan kode.
5. LONG 4 bytes; Long integer juga mempunyai kesamaan dengan short integer, yang mana dapat digunakan untuk angka tanpa pecahan namun juga dengan kemungkinan digit yang lebih panjang lagi.
6. DATE 8 bytes; Date berfungsi untuk dapat menyimpan tanggal, waktu ataupun tanggal-waktu dengan contoh format mm/d-d/yyyy hh:mm:ss

21.1.10 sistem informasi geografis shapefile

dalam sebuah sistem informasi ditampilkan berbagai data seperti shapefile dengan berbagai skala, shapefile adalah format data yang menyimpan lokasi geometrik dan informasi geografis. Bentuk geometri yang tersimpan adalah dalam bentuk koordinat vektor, Format ini adalah format yang dikeluarkan oleh Environmental System Resource Institute (ESRI) yang merupakan salah satu vendor SIG terkemuka. Format data pada shapefile merupakan format data vektor yang terkenal untuk software Sistem Informasi Geografis (GIS). Shapefile adalah format data vektor yang digunakan untuk menyimpan lokasi , bentuk, dan atribut dari fitur geografis. Format data pada SHP disimpan dalam satu set file terkait dan berisi dalam satu kelas fitur Format data pada vektor ini berisi tentang data referensi geografis yang didefinisikan sebagai objek tunggal seperti jalan, sungai, landmark, kode pos. Data fitur dan atribut akan disimpan dalam satu SHP. Banyak aplikasi yang berbentuk GIS yang bersifat opensource ataupun proprietary dapat bekerja dengan shapefile.

21.1.11 spesifikasi shapefile

Sesungguhnya shapefile merupakan kumpulan beberapa file dengan tiga ekstensi utama yang mandatory/wajib yaitu shp, shx, dbf serta beberapa tambahan atau optional file yang lain. Satu shapefile dianjurkan dengan nama file yang sama dengan ekstensi yang berbeda, shp (shape format, menyimpan data gambar geometry), shx (shape index format, index dari gambar geometry sehingga memudahkan atau mempercepat proses pencarian), dbf (attribute format, berisi table attribute dari tiap gambar dalam dBas).

21.1.12 format data shapefile

Ukuran data pada SHP dan file komponen DBF tidak dapat melebihi 2 GB (231 bit), sekitar 70 juta fitur titik yang terbaik. Jumlah maksimum fitur untuk jenis geometri lainnya bervariasi tergantung pada jumlah yang digunakan, panjang maksimum nama field adalah 10 karakter, dan jumlah maksimum dari field adalah 255.

21.1.13 pembuatan shapefile

Shapefile adalah format data vektor geospasial untuk software GIS yang dikembangkan oleh ESRI (Environmental System Research Institute) dengan spesifikasi yang terbuka untuk kepentingan software GIS atau software yang bisa mengolah (input) format data pada SHP antara lain ArcGIS, ArcView, MapInfo, ERDAS, Global Mapper.

DATA GEOSPASIAL POINT

22.1 Definisi GIS(GEOGRAPHICS INFORMATION SYSTEM)

Sistem Informasi Geografis (SIG) adalah sebuah sistem yang dirancang untuk menangkap, menyimpan, memanipulasi, menganalisa, mengatur, dan menampilkan seluruh jenis data geografi. SIG tidak lepas dari data spasial, yang merupakan sebuah data yang mengacu pada posisi, obyek, dan hubungan di antaranya dalam ruang bumi. Data spasial dalam SIG terbagi menjadi dua model data yaitu model data vektor dan model data raster. Model data vektor merepresentasikan bumi sebagai suatu mosaik yang terdiri atas garis (arc/line), polygon (daerah yang dibatasi oleh garis yang berawal dan berhenti pada titik yang sama), titik/point (node yang mempunyai label), dan nodes (merupakan titik perpotongan antara dua buah garis). Model data raster atau sel grid merepresentasikan obyek geografis sebagai struktur sel grid yang diwakili oleh setiap pixel pada citra. Model data raster sangat baik untuk merepresentasikan batas-batas yang berubah secara gradual seperti jenis tanah, vegetasi, dan lain-lain

22.2 Definisi Data Spasial (GEOGRAPHICS INFORMATION SYSTEM)

Data Spasial Sistem Informasi Geografis (SIG) model data yang akan digunakan dari bentuk dunia nyata harus dapat diimplementasikan ke dalam basisdata. Data ini dimasukkan ke dalam komputer yang nantinya memanipulasi objek dasar yang memiliki atribut geometri (entity spasial/entity geografis) (Prahasta, 2002a). Data spasial pada dasarnya dapat disimpulkan bahwa data spasial merupakan suatu entitas data dalam Sistem Informasi Geografis (SIG) yang dapat dikelola, dianalisa dan dapat memetakan informasi objek keruangan beserta data-data atributnya serta dapat disimpan di dalam database yang dapat ditampilkan kedalam suatu sistem tertentu sehingga dapat mendukung dalam pengambilan keputusan.

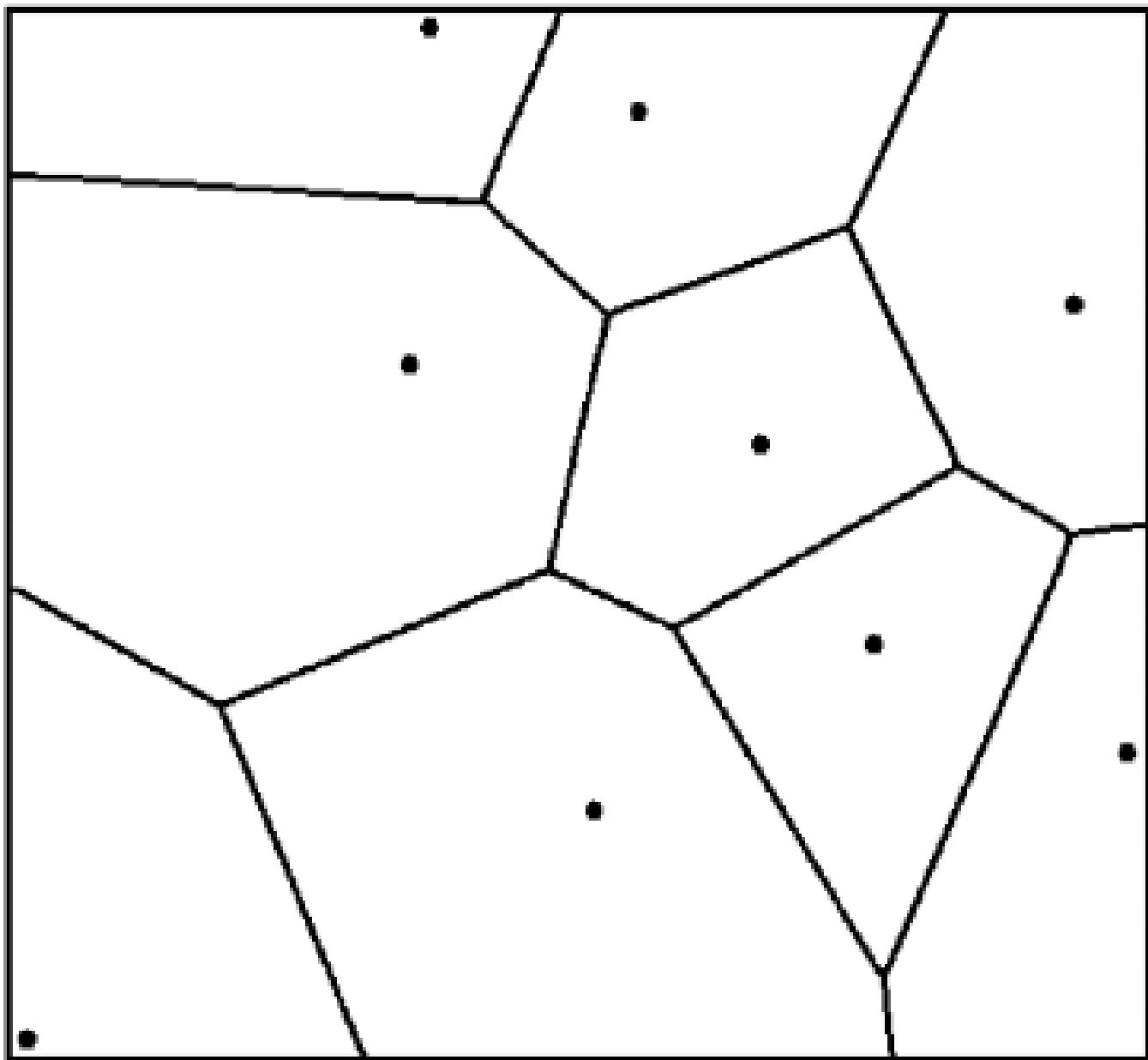
22.2.1 ABSTRAK

Diagram Voronoi merupakan konsep interdisipliner yang telah diterapkan pada berbagai bidang. Dalam sistem informasi geografis (GIS), kemampuan yang ada untuk menghasilkan diagram Voronoi biasanya fokus pada hal biasa (tidak berbobot) titik fitur/point (tidak linear atau area). Untuk integrasi yang lebih baik dari model diagram Voronoi dan GIS, pendekatan berbasis raster dikembangkan, dan dilaksanakan mulus sebagai perpanjangan ArcGIS menggunakan ArcObjects. Dalam tulisan ini, metodologi dan pelaksanaan ekstensi dijelaskan, dan contoh yang disediakan untuk fitur titik(point), line, dan poligon. Keuntungan dan keterbatasan ekstensi juga dibahas. Ekstensi memiliki beberapa fitur berikut:

1. bekerja untuk titik, garis, dan fitur vektor poligon;
2. dapat menghasilkan kedua diagram Voronoi biasa dan multiplicatively tertimbang dalam format vektor;
3. dapat menetapkan atribut non-spasial fitur input ke Voronoi sel melalui bergabung spasial;
4. dapat menghasilkan biasa atau Euclidean dataset raster jarak tertimbang untuk aplikasi pemodelan spasial. Hasil dapat dengan mudah dikombinasikan dengan GIS dataset lain untuk mendukung analisis spasial berbasis vektor dan pemodelan spasial berbasis raster.

22.2.2 Diagram Voronoi untuk Point, Line dan Poligon dalam fitur GIS

Mengingat satu set jumlah terbatas titik berbeda di ruang 2-D Euclidean, diagram Voronoi titik set adalah kumpulan dari daerah yang membagi pesawat, dan semua lokasi di satu wilayah (kecuali pada batas wilayah) lebih dekat ke titik sponding corredaripada titik lain (Gbr. 1). Diagram Voronoi dinamai matematikawan Rusia Georgy Fedosevich Voronoi yang didefinisikan dan mempelajari kasus n-dimensi umum pada tahun 1908. Diskusi tentang konsep diagram Voronoi dari titik pandangan-sejarah dan geometris dapat ditemukan di Okabe et al. (1992, 2000). Diagram Voronoi telah banyak diterapkan untuk masalah partisi ruang dalam berbagai disiplin ilmu, dari astronomi untuk geografi untuk zoologi. Okabe et al. (2000) tercatat 22 bidang di mana aplikasi diagram Voronoi dapat ditemukan, dan diyakini bahwa bidang aplikasi tidak terbatas pada daftar. Contoh aplikasi terbaru dari diagram Voronoi termasuk berbasis Voronoi selular automata (Shi dan Pang, 2000), tomografi adaptif 3-D di (. Bo Hm et al, 2000) geofisika prospeksi, analisis distribusi gempa bumi-dan peringatan dini. Gambar 22.2 adalah ilustrasi dari Voronoi Diagram Point.

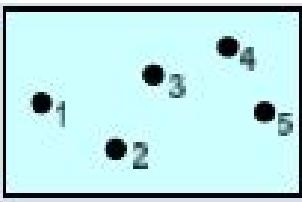


Gambar 22.1 voronoi.

22.2.3 Model data Vektor pada Geographics Information System GIS

Vektor pada GIS mampu melakukan penempatan, menampilkan data spasial bahkan menyimpan datanya yang menggunakan titik-titik, garis-garis dan juga poligon yang dilengkapi dengan artibut-artibutnya. Bentuk-bentuk dasar representasi dari data spasial ini di dalam sistem model data vektor dapat didefinisikan oleh sistem koordinat kartesian dua dimensi (X,Y). Dimana di dalam model data spasial vektor, garis-garis atau kurva (busur atau arcs) adalah berupa sekumpulan titik-titik terurut yang saling berhubungan (Prahasta, 2002a).

22.2.3.1 Model data Vektor dengan bentuk point/titik pada Geographics Information System GIS Point/titik merupakan representasi grafis yang paling sederhana pada suatu objek. titik pada data vektor tidak mempunyai dimensi tetapi bisa ditampilkan ke dalam bentuk simbol baik pada peta maupun dalam layar monitor. contoh lokasi fasilitas kesehatan, kantor pemerintah dan lain-lain. Pada gambar 22.2 dijelaskan bahwa gambar point/titik data vektor GIS. Dalam sebuah artikel dari husein yang menyebutkan bahwa GIS merupakan pemahaman dari Geography, Information dan System [48].

Jenis	Contoh Representasi	Contoh Atribut																		
Titik		<table border="1"> <thead> <tr> <th>ID</th><th>Nama</th><th>Lokasi</th></tr> </thead> <tbody> <tr> <td>1</td><td>SMU 1</td><td>Kec. A</td></tr> <tr> <td>2</td><td>SDN B</td><td>Kec. A</td></tr> <tr> <td>3</td><td>SMP 5</td><td>Kec. A</td></tr> <tr> <td>4</td><td>SDN A</td><td>Kec. B</td></tr> <tr> <td>5</td><td>SMU 2</td><td>Kec. B</td></tr> </tbody> </table>	ID	Nama	Lokasi	1	SMU 1	Kec. A	2	SDN B	Kec. A	3	SMP 5	Kec. A	4	SDN A	Kec. B	5	SMU 2	Kec. B
ID	Nama	Lokasi																		
1	SMU 1	Kec. A																		
2	SDN B	Kec. A																		
3	SMP 5	Kec. A																		
4	SDN A	Kec. B																		
5	SMU 2	Kec. B																		

Gambar 22.2 point.

22.3 contoh perancangan sistem (GEOGRAPHICS INFORMATION SYSTEM) dengan menggunakan Point

Pada Point dapat menampung SHP yang bertipe Point. Pada gambar 22.3 dijelaskan bahwa gambar point/titik data vektor GIS sebagai berikut. Dimana di dalam kelas terdapat atribut berupa :

1. X bertipe double, adalah titik koordinat dalam garis X dari sebuah point.
2. Y bertipe double, adalah titik koordinat dalam garis Y dari sebuah point.

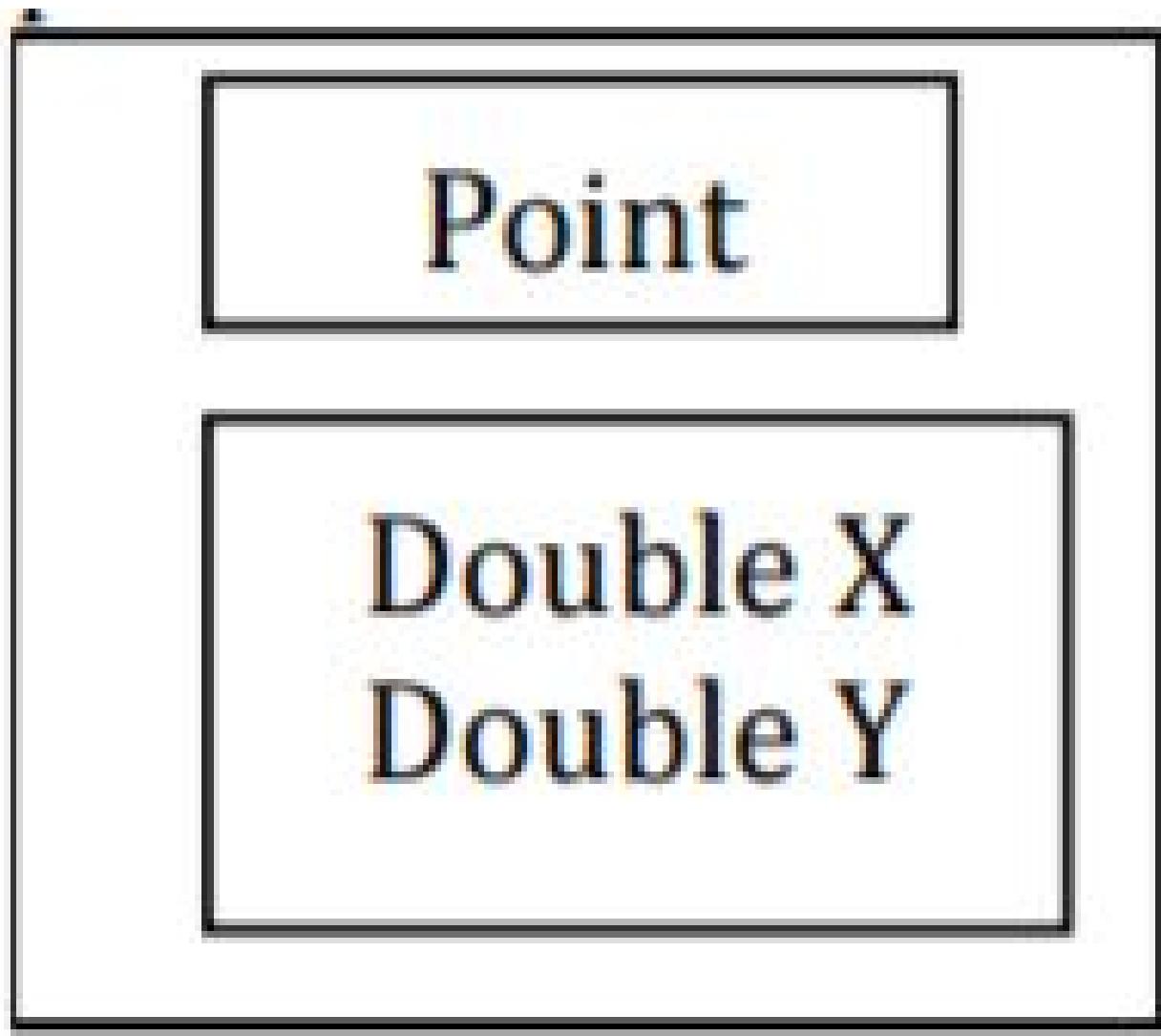
22.3.1 Layer objek/point

Dimana Layer dari sebuah point dapat dilihat dari gambar 22.4 tipe data yang digunakan merupakan tipe data point yang terdapat koordinat real, yaitu berupa latitude dan longitude. dimana layer ini mengkonfirmasi letak sekolah dan perumahan.

22.3.2 Pola Titik

Pola titik (point pattern) adalah pola yang muncul dari sebuah variabel yang dianalisis pada daerah yang tersampel (Cressie, 1993). Sampel yang digunakan merupakan sebuah sampel yang memiliki bentuk tidak beraturan atau juga sampel yang memiliki jarak berbeda. Daerah tersebut dapat diperoleh dari sebuah data koordinat kartesius (x,y) dengan berdasarkan titik yang diamati. Data pada pola titik spasial yang sudah ada dapat diperoleh dari informasi apakah pola yang tadi diperoleh dapat menggambarkan keteracakannya data spasial, clustering, ataupun keteraturan. Seperti contohnya yaitu : Penentuan sebuah posisi pohon-pohon yang memiliki ukuran tertentu. Apakah pohon-pohon tersebut dapat membentuk sebuah pola keteracakannya spasial, clustering, ataupun keteraturan. Analisis dari data pola titik yang dilakukan karena agar dapat mengetahui apakah daerah titik yang akan menjadi objek penelitian tersebut membentuk daerah beraturan atau tidak , Sehingga nantinya dapat diketahui apakah terjadi ketergantungan antar titik atau tidak. Pada estimasi dari data spasial, adalah teknik analisis data geostatistika bertujuan untuk mengetahui serta untuk melakukan estimasi nilai dari variabel teregional pada lokasi s. Nilai dari suatu variabel yang diamati nantinya dapat dinyatakan sebagai variabel random spasial $Z(s)$ dengan s adalah vektor lokasi DR^d . Dan menggunakan metode kriging ini merupakan metode untuk melakukan estimasi nilai dari variabel yang teregional $Z(s)$ pada suatu lokasi A. berdasarkan variabel random spasial $Z(s)$. Variabel random spasial $Z(s)$ pada data geostatistika merupakan suatu variabel random Z di lokasi s . Pada data spasial variabel random X dapat didefinisikan sebagai variabel random spasial $Z(s)$ di lokasi s dan dari variabel random Y didefinisikan sebagai variabel random spasial $Z(s+h)$ di lokasi $s+h$. Pada analisis data geostatistika variansi ini digunakan untuk menentukan korelasi antara variabel random spasial $Z(s)$ dan $Z(s+h)$. Nilai variansi dari variabel random spasial pada lokasi $Z(s)$ dan $Z(s+h)$ dapat ditentukan dengan rumus sebagai berikut:

$$\begin{aligned} varz(s)z(s+h) &= varz9s0 + varz(s+h)2covz(s).z(s+h) \\ &= EZ(s) - z(s+h) - E(z(s)) - E(Z(s+h))] \end{aligned} \quad (22.1)$$



Gambar 22.3 kelaspoint.

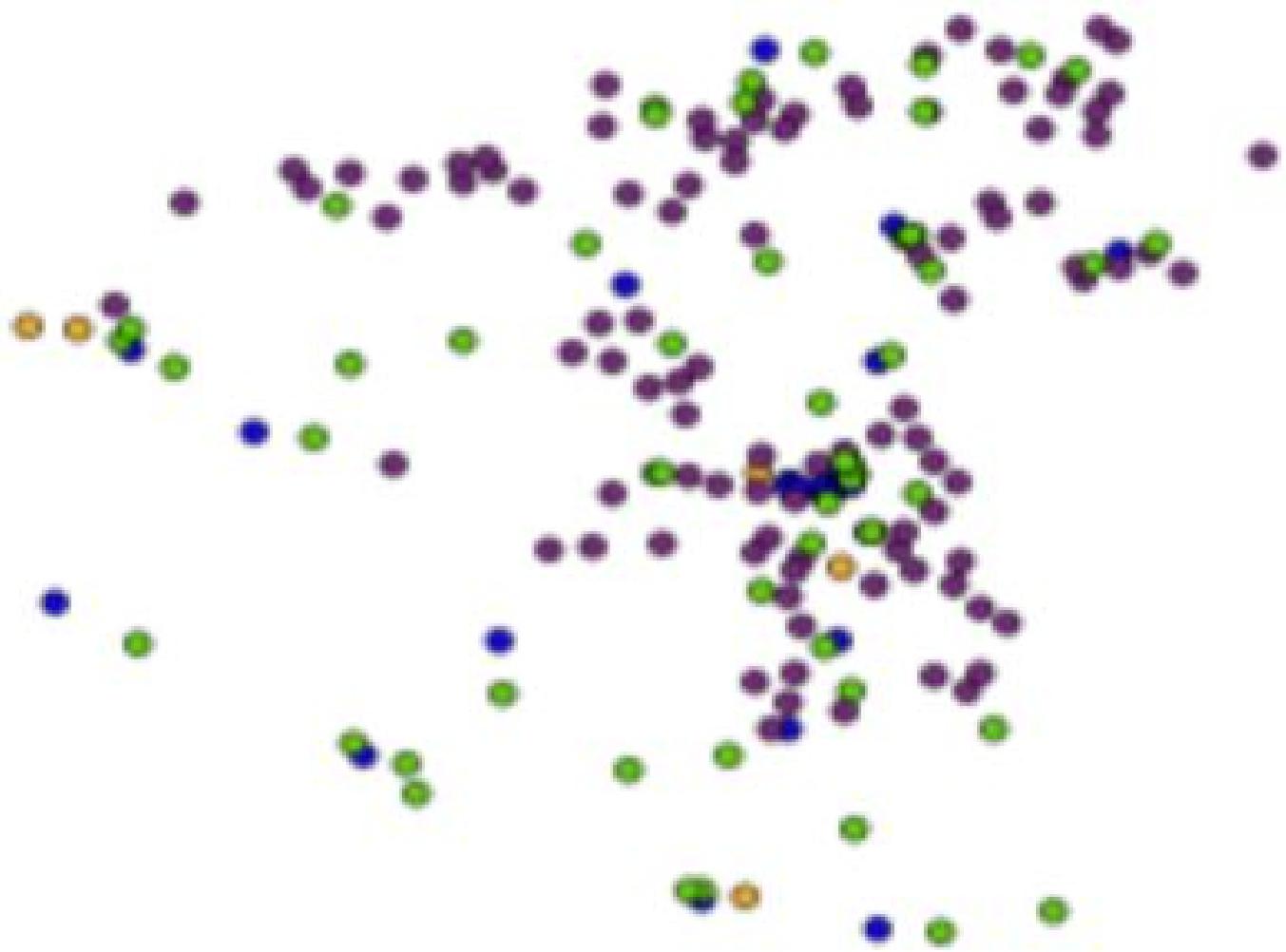
22.3.3 Point dalam MongoDB

MongoDB merupakan sebuah DBMS non relasional yang berorientasi dokumen dan bersifat open source yang ditulis dalam C++. MongoDB dikembangkan oleh perusahaan 10gen. Sebuah basis data pada MongoDB berisi satu atau lebih collections dari documents. Documents pada MongoDB dapat berisi nilai yang nilai tersebut dapat berisi sebuah document lain, kumpulan document, atau tipe data dasar seperti Double, String, dan Date. MongoDB menyiapkan tipe pengindeksan geospasial yang berguna mendukung keperluan kueri geospasial seperti 2D dan 2DSphere. Indeks 2D digunakan untuk menyimpan data sebagai titik pada bidang datar dua dimensi. Sedangkan apabila titik tersebut disimpan pada bidang lengkung maka indeks 2DSphere dapat digunakan. Secara default, sistem referensi koordinat yang digunakan MongoDB pada objek lengkung seperti bumi yaitu WGS84. MongoDB mendukung tipe data geospasial seperti Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, Feature, FeatureCollection, dan GeometryCollection (de Souza et al. 2014).

Membuat Data Geospasial Import shapefile Masukan variable, misalkan variable a untuk shapefile.writer() a = shapefile.writer() Jadi, format membuat data geospasial ada 2, yaitu : .shp =_i a.point(x,y) a.poly [(x,y),(v,w)] .dbf =_i a.field (name.field,c,40) a.record (bdg) Data geospasial tersebut disimpan menggunakan method a.save(file.shp).

Arti dari method pada writer : Point (x,y) : memasukkan data berbentuk point ke dalam .shp dan seluruh data harus berformat ESRI.1 Poly [(a,b),(c,d)] : memasukkan data geospasial berbentuk polygon (kembali ke titik awal) atau polyline (tidak kembali ketitik awal). Field (nama,c,40) : artinya membuat atribut polygon dengan table nama dengan tipe data varchar dengan panjang 40. Method ini dapat diulang dan dapat dilakukan untuk kruthan field baru lagi. Record(Bandung) : Mengisi table dimana yang hanya 1 field dengan value atau nilai Bandung. Save (file name) : menyimpan file dengan save file

Titik(dimensi nol - point) adalah representasi grafis atau geometri yang paling sederhana bagi objek spasial. Representasi ini tidak mempunyai dimensi, namun di atas peta dapat diidentifikasi lalu dapat ditampilkan pada layar monitor dengan menggunakan simbol-simbol tertentu. Perlu dipahami juga bahwa skala akan menentukan apakah suatu objek akan ditampilkan sebagai titik atau polygon. Pada peta yang berskala besar, unsur-unsur bangunan akan ditampilkan sebagai polygon, sedangkan pada skala kecil akan ditampilkan sebagai unsur-unsur titik. Format titik : koordinat tunggal, tanpa panjang, tanpa luasan. Contoh : lokasi kecelakaan, letak pohon



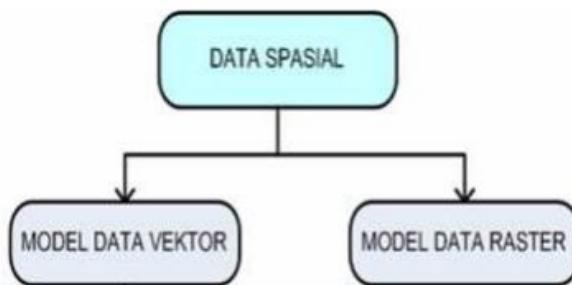
Gambar 22.4 layerpoint.

DATA GEOSPASIAL POLIGON

23.1 Data Polygon

23.1.1 Data Geospasial

Istilah ini digunakan karena GIS dibangun berdasarkan pada geografi atau spasial. Objek ini mengarah pada spesifikasi lokasi dalam suatu space. Objek bisa berupa fisik, budaya atau ekonomi alamiah. Penampakan tersebut ditampilkan pada suatu peta untuk memberikan gambaran yang representatif dari spasial suatu objek. Terdapat dua model data geospasial, yaitu model data raster dan model data vektor. Pada gambar 23.1 dijelaskan tentang klasifikasi model data geospasial.



Gambar 23.1 Klasifikasi Model Data Geospasial.

23.1.1.1 Data Vektor Model data vektor merupakan model data yang paling banyak digunakan dan dikenal pula sebagai model data spaghetti. Lembaran kertas peta ditranslasi garis demi garis ke dalam list koordinat (x,y) dalam format digital. Sebuah titik dikodekkan sebagai pasangan koordinat (x,y) tunggal. Sebuah garis dikodekkan sebagai listatau string pasangan koordinat (x,y). Sementara area atau luasan dikodekan sebagai polygon. Data vektor juga merupakan data yang direkam dalam bentuk koordinat titik yang menampilkan, menempatkan dan menyimpan data spesial dengan menggunakan titik, garis atau area (poligon). Ada tiga tipe data vektor (titik, garis, dan poligon) yang bisa digunakan untuk menampilkan informasi pada peta. Titik bisa digunakan sebagai lokasi sebuah kota atau posisi towerradio. Garis bisa digunakan untuk menunjukkan route suatu perjalanan atau menggambarkan boundary. Poligon bisa digunakan untuk menggambarkan sebuah danau atau sebuah negara pada peta dunia. Dalam format vektor, bumi direpresentasikan sebagai suatu mosaik dari garis (arc/line), poligon (daerah yang dibatasi oleh garis yang berawal dan berakhir pada titik yang sama), titik/ point (node yang mempunyai label), nodes (merupakan titik perpotongan antara dua garis). Setiap bagian dari data vektor dapat saja mempunyai informasi yang berasosiasi satu dengan lainnya seperti penggunaan sebuah label untuk menggambarkan informasi pada suatu lokasi. Peta vektor terdiri dari titik, garis, dan area poligon. Bentuknya dapat berupa peta lokal jalan. Kelebihan data vektor yaitu struktur datanya lebih rumit, efisiensi untuk analisis, sebagai sarana representasi yang baik, transformasi proyeksi lebih efisien, ketelitian, akurat dan lebih presisi serta relasi atribut langsung dengan DBMS (database). Adapun kekurangan data vektor adalah sulit dalam melakukan proses overlay,

Tidak bisa menampilkan data image/foto udara, struktur data yang terlalu banyak tidak efektif dalam menampilkan banyak spasial, memerlukan algoritma dan proses yang sangat kompleks, kualitas (output) sangat bergantung dengan printer dan kartografi dan sulit dilakukan simulasi. Model ini berbasiskan pada titik dengan nilai koordinat (x,y) untuk membangun objek spasialnya. Objek yang dibangun terbagi menjadi tiga bagian lagi yaitu berupa titik (point), garis (line), dan area (polygon).

23.2 Pengertian Polygon

Dalam artikel Mahendra menjelaskan Polygon merupakan representasi objek dalam dua dimensi. Contoh : danau, persil tanah, dan lain-lain. Entity polygon dapat direpresentasikan dengan berbagai cara di dalam model data vektor. Polygon berasal dari kata Poly yaitu banyak dan gon(gone) yaitu titik. Yang dimaksud adalah polygon yang digunakan sebagai kerangka dasar pemetaan yang memiliki titik-titik dimana titik tersebut mempunyai sebuah koordinat X dan Y.

23.2.1 Jenis - jenis Polygon

1. Polygon tertutup Polygon Tertutup adalah suatu polygon dimana titik awal dan titik akhir mempunyai posisi yang sama atau berhimpit, sehingga polygon ini adalah suatu rangkaian tertutup.
2. Polygon tertutup(koordinat lokal) Polygon tertutup yang terikat oleh azimuth dan koordinat.
3. Polygon terbuka tidak terikat/lepas(koordinat lokal) Polygon terbuka tanpa ikatan sama sekali (polygon lepas), pengukuran seperti ini akan terjadi pada daerah-daerah yang tidak ada titik tetapnya dan sulit melakukan pengukuran baik dengan cara astronomis maupun dengan satelit. Polygon semacam ini dihitung dengan orientasi lokal artinya koordinat dan azimuth awalnya dimisalkan sembarang.
4. Polygon terbuka tidak terikat sempurna Polygon terbuka yang salah satu ujungnya terikat oleh azimuth saja, sedangkan ujung yang lain tidak terikat sama sekali. Polygon semacam ini dapat dihitung dari azimuth awal dan yang diketahui dan sudut-sudut polygon yang diukur, sedangkan koordinat dari masing-masing titiknya masih lokal. Polygon terbuka yang salah satu ujungnya terikat oleh koordinat saja, sedangkan ujung yang lain tidak terikat sama sekali. Polygon semacam ini dapat dihitung dengan cara memisalkan azimuth awal sehingga masing-masing azimuth sisi polygon dapat dihitung, sedangkan koordinat masing-masing titik dihitung berdasarkan koordinat yang diketahui. Oleh karena itu pada polygon bentuk ini koordinat yang dianggap betul hanyalah pada koordinat titik yang diketahui (awal) sehingga polygon ini tidak ada orientasinya.
5. Polygon terbuka terikat sempurna. Suatu polygon yang terikat sempurna dapat terjadi pada polygon tertutup ataupun polygon terbuka, suatu titik dikatakan sempurna sebagai titik ikat apabila diketahui koordinat dan jurusannya minimum 2 buah titik ikat dan tingkatnya berada diatas titik yang akan dihasilkan. Polygon terbuka yang masing-masing ujungnya terikat azimuth dan koordinat.

Serangkaian titik-titik yang dihubungkan dengan garis lurus sehingga titik-titik tersebut membentuk sebuah rangkaian(jaringan) titik atau polygon. Pada pekerjaan pembuatan peta, rangkaian titik polygon digunakan sebagai kerangka peta, yaitu merupakan jaringan titik-titik yang telah tertentu letakknya di tanah yang sudah ditandai dengan patok, dimana semua benda buatan manusia seperti jembatan, jalan raya, gedung maupun benda-benda alam seperti danau, bukit, dan sungai akan berorientasikan. Kedudukan benda pada pekerjaan pemetaan biasanya dinyatakan dengan sistem koordinat kartesius tegak lurus (X,Y) dibidang datar(peta), dengan sumbu X menyatakan arah timur-barat dan sumbu Y menyatakan arah utara - selatan. Koordinat titik-titik polygon harus cukup teliti mengingat ketelitian letak dan ukuran benda - benda yang akan dipetakan sangat tergantung pada ketelitian dan kerangka peta.

23.2.2 Bentuk-bentuk Polygon

Menurut bentuknya, polygon dibedakan menjadi dua bentuk :

1. Polygon Terbuka Polygon terbuka adalah suatu polygon dimana titik awal dan titik akhirnya berbeda. Polygon yang titik awal dan titik akhirnya merupakan titik yang berlainan (bukan satu titik yang sama). Polygon terbuka ini dapat dibagi lebih lanjut berdasarkan peningkatan pada titik - titik(kedua titik ujungnya). Ada peningkatan untuk polygon terbuka yaitu peningkatan koordinat. Peningkatan koordinat pada polygon terbagi menjadi beberapa yaitu, tanpa ikatan sama sekali dan pada salah satu ujung yang lain tanpa ikatan sama sekali. Jenis - jenis polygon terbuka adalah Polygon terbuka terikat sempurna, Polygon terbuka terikat sepihak dan Polygon terbuka tidak terikat.
2. Polygon Tertutup Polygon Tertutup adalah suatu polygon dimana titik awal dan titik akhir mempunyai posisi yang sama atau berhimpit, sehingga polygon ini adalah suatu rangkaian tertutup. Pada polygon tertutup garis - garis kembali ke titik awal, jadi membentuk segi banyak. Berakhir di stasiun lain yang mempunyai ketelitian letak sama atau lebih besar daripada ketelitian letak titik awal. Berdasarkan fungsinya, polygon dibedakan menjadi Polygon untuk keperluan kerangka peta, syaratnya harus memiliki titik - titik yang cukup baik, dalam arti menjangkau semua wilayah dan Polygon yang berfungsi sebagai titik - titik pertolongan untuk mengambil detail lapangan. Kelemahan polygon tertutup yaitu, bila ada kesalahan yang profesional dengan jarak (salah satu sistematis) tidak akan ketahuan. Dengan kata lain, walaupun ada kesalahan, namun polygon tertutup kelihatan baik juga. Jarak - jarak yang diukur secara elektronis sangat mudah menggapi kesalahan seperti kesalahan frekuensi gelombang.

23.2.3 Karakteristik Polygon

1. Titik distrukturisasi dan disimpan (direcord) sebagai satu pasang koordinat(x,y).
2. Garis distrukturisasi dan disimpan sebagai suatu susunan pasangan koordinat(x,y) yang beraturan.
3. Luasan distrukturisasi dan disimpan sebagai satuan susunan pasangan koordinat(x,y) yang berurutan yang menyatakan segmen-segmen garis yang menutup menjadi suatu poligon.
4. Data dalam bentuk poligon (area), meliputi daerah administrasi, geologi, geomorfologi, jenis tanah, dan penggunaan tanah.
5. Data dalam bentuk pixel (grid) meliputi citra satelit dan foto udara. Data dasar yang dimasukan dalam SIG diperoleh dari tiga sumber, yaitu data lapangan (terestris), data peta, dan data penginderaan jauh.

Struktur data polygon bertujuan untuk mendeskripsikan properties yang bersifat topologi dari suatu area sedemikian rupa sehingga properties yang dimiliki oleh blok-blok bangunan spasial dasar dapat ditampilkan dan dimanipulasi sebagai data peta tematik. Seperti halnya titik dan garis, area juga dapat menggambarkan objek yang berbeda menurut skalanya. Area dapat menggambarkan wilayah hutan atau sawah pada peta skala besar. Poligon-poligon didefinisikan dengan menggunakan arcs: dengan melakukan tracing batas-batasnya searah dengan perputaran jarum jam (clockwise), merekam komponen arc beserta orientasinya, memberikan tanda negative pada arcs yang mendefinisikan batas-batas internal. Dalam GIS istilah poligon adalah kumpulan pasangan koordinat yang menghubungkan paling sedikit tiga titik (vertex) dan titik awal bertemu dengan titik yang paling akhir dan menutup. Misalnya : Batas Administrasi Region, merupakan sekumpulan poligon, di mana masing-masing poligon tersebut dapat atau tidak mempunyai keterikatan di antaranya akan tetapi saling bertampalan dalam satu data set. Cara yang paling sederhana untuk merepresentasikan suatu poligon adalah pengembangan dari cara yang digunakan untuk merepresentasikan arc yang sederhana yaitu merepresentasikan setiap poligon sebagai sekumpulan koordinat (x,y) yang membentuk segmen garis, dimana mempunyai titik awal dan titik akhir segmen garis yang sama (memiliki nilai koordinat yang sama). Bentuk-bentuk dasar representasi data spasial ini, di dalam sistem model data vektor, didefinisikan oleh sistem koordinat kartesian dua dimensi (x,y). Di dalam model data spasial vektor, garis-garis atau kurva merupakan sekumpulan titik-titik terurut yang dihubungkan. Sedangkan luasan atau poligon juga disimpan sebagai sekumpulan list titik-titik, tetapi dengan catatan bahwa titik awal dan titik akhir poligon memiliki nilai koordinat yang sama dengan syarat poligon tersebut tertutup. Representasi vektor suatu objek merupakan suatu usaha di dalam menyajikan objek yang bersangkutan sesempurna mungkin. Untuk itu, ruang atau dimensi koordinat diasumsikan bersifat kontinyu yang memungkinkan semua posisi, panjang dan dimensi didefinisikan dengan presisi. Fitur poligon adalah area tertutup seperti bendungan, pulau, batas negara dan sebagainya. Seperti fitur polyline, poligon diciptakan dari rangkaian simpul yang terhubung dengan garis kontinyu. Namun karena poligon selalu menggambarkan area tertutup, simpul pertama dan terakhir harus selalu berada di tempat yang sama! Poligon sering memiliki batas geometri bersama yang sama dengan poligon tetangga. Banyak aplikasi GIS memiliki kemampuan untuk memastikan bahwa batas-batas poligon tetangga persis sama. Kita akan membahasnya di topik Topology nanti di tutorial ini. Seperti halnya titik dan polyline, poligon memiliki atribut. Atribut menggambarkan masing-masing poligon. Misalnya bendungan mungkin memiliki atribut untuk kedalaman dan kualitas air. Format : Koordinat dengan titik awal dan akhir sama, mempunyai panjang dan luasan. Contoh : Tanah persil, bangunan. [49].

Bagian IV

PEMROGRAMAN SIG PYTHON

PEMROGRAMAN SIG PYTHON

24.1 Pengertian python

Menurut beberapa programmer yang menggunakan bahasa pemrograman python ini, Python merupakan salah satu bahasa pemrograman yang dinamis dan mempunyai sistem manajemen memori yang otomatis Seperti bahasa pemrograman yang dinamis lainnya, Python biasanya digunakan melalui script atau kode-kode meskipun bahasa pemrograman ini lebih banyak dimanfaatkan untuk yang umumnya tidak banyak yang menggunakan script. Bahasa Pemrograman Python ini bisa dipakai dan digunakan untuk segala macam kebutuhan dari pengembang-pengembang software atau perangkat lunak dan juga Bahasa Pemrograman Python ini dapat digunakan di berbagai sistem operasi.

24.2 Sejarah python

Dalam sebuah artikel oleh Guido van Rossum (berbahasa Inggris) yang di terjemahkan menyebutkan bahwa sejarah bahasa pemrograman Python dimulai pada awal 1990 dan diciptakan oleh Guido Van Rossum di Stichting Mathematisch Centrum (CWI) Belanda yang merupakan kelanjutan dari bahasa pemrograman ABC. Rilis terakhir Python dari CWI adalah versi 1.2 pada tahun 1995. Kemudian Guido melanjutkan Python pada Corporation for National Research (CNRI) di Virginia. Python versi 1.6 merupakan versi terakhir yang dikembangkan oleh CNRI. Pada tahun 2000, Guido dan pengembang Python berpindah ke BeOpen.com dan hanya merilis satu versi yakni 2.0. Kemudian Guido pindah lagi ke Digital Creations.

Gambar 24.1 merupakan foto dari Guido van Rossum yang merupakan pencipta dari python.

Python 3.0 yang juga disebut Python 3000 atau py3k dirilis untuk pertama kali dengan kode yang tidak efisien, dengan dirilisnya pada tanggal 3 Desember 2008 dalam periode pengujian yang panjang. Maka, dengan banyak fitur utamanya yang telah dikirim kembali ke Python 2.6 yang cocok ke untuk kembali belakang dan menjadi versi 2.7.

24.3 Indentation

Dalam penulisan bahasa pemrograman Python, setiap perintah yang masih dalam satu kesatuan diberikan tambahan spasi dari perintah yang ada diatasnya, bukan kurung kurawal atau kata kunci, untuk membatasi blok. Hal ini disebut juga aturan off-side. Peningkatan indentasi datang setelah perintah tertentu. penurunan indentasi menandakan akhir blok perintah sebelumnya.

Contoh perintah bahasa pemrograman python yaitu : Perintah perulangan

```
while x < 10:  
    while y < 10:  
        print y,  
        y = y + 1  
        print x,  
        x = x + 1
```



Gambar 24.1 Guido van Rossum pencipta Python

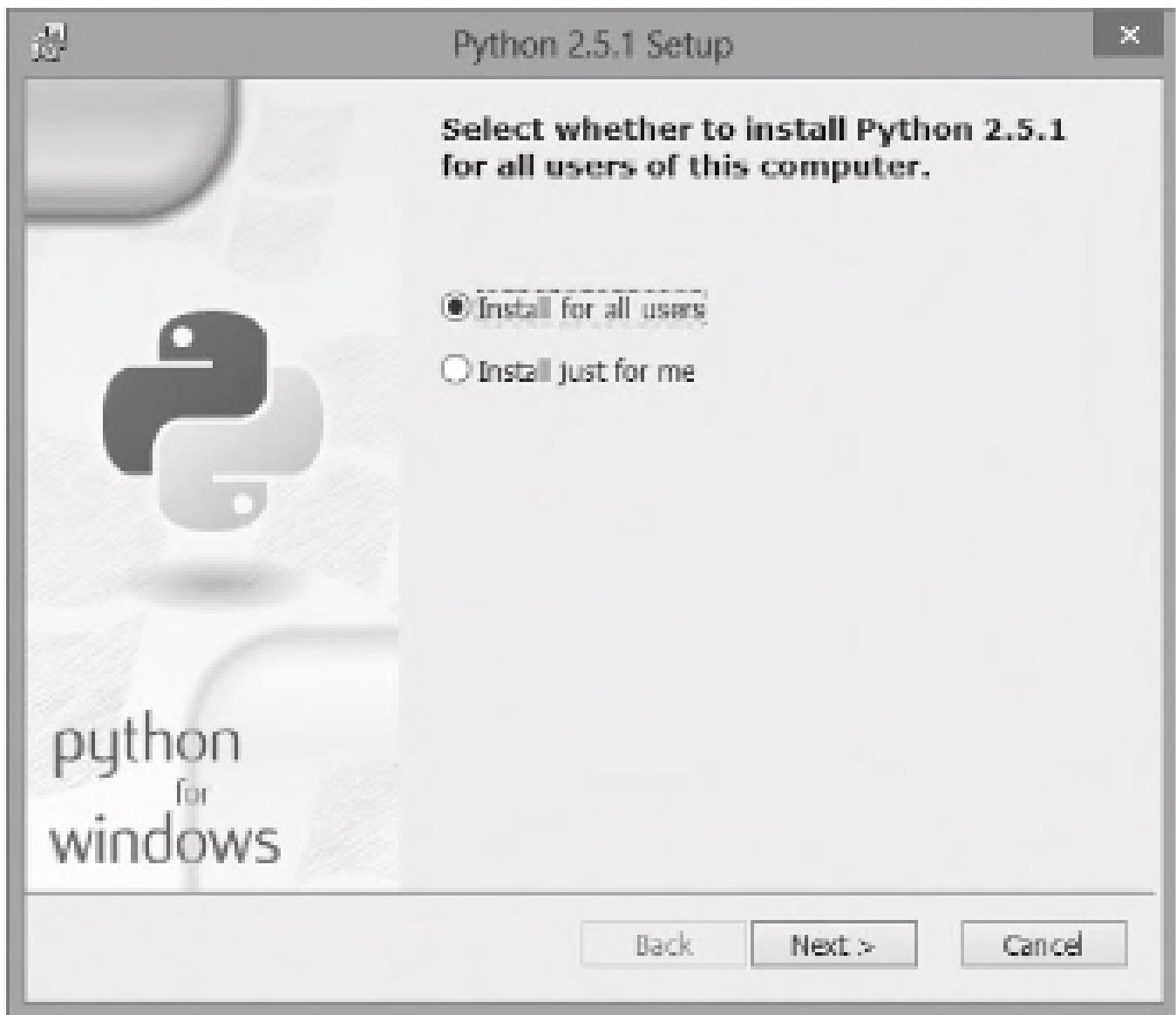
24.3.1 Fitur dan filosofi

Python adalah bahasa pemrograman multi-paradigma: maksudnya pemrograman berorientasi objek dan pemrograman terstruktur yang didukung sepenuhnya, dan ada berbagai fitur bahasa yang mendukung pemrograman fungsional dan pemrograman berorientasi-aspek (termasuk metaprogramming dan metode sihir). Banyak model lain yang didukung dengan menggunakan ekstensi, termasuk disain oleh kontrak dan pemrograman logika. Desain Python hanya menawarkan dukungan yang terbatas untuk pemrograman fungsional dalam tradisi Lisp. Bahasa memiliki fungsi peta (), reduce () dan filter (), lengkap untuk daftar, kamus, dan set, serta ekspresi generator. Perpustakaan standar memiliki dua modul (itertools dan functools) yang mengimplementasikan alat fungsional yang dipinjam dari Haskell dan Standard ML[50].

24.4 instalasi python untuk windows

Dari sebuah artikel oleh Guido Van Rossum yang telah diterjemahkan, menyatakan bahwa Python Interpreter dan standar library yang tersebar luas tersedia secara bebas dalam berbagai bentuk source atau bentuk biner untuk semua platform utama dari situs Python (www.python.org) dan dapat didownload secara gratis. Situs yang sama juga berisi distribusi dan petunjuk ke berbagai modul, program dan tools Python secara gratis, dan dokumentasi sebagai tambahan. [51] di sini penulis artikel menunjukkan instalasi software python yang akan digunakan untuk sql injection. Pada dasarnya proses instalasi python tidak jauh berbeda dengan instalasi software pada umumnya berikut adalah langkah-langkah instalasi python:

1. Langkah pertama adalah klik ganda pada file instalasi python.exe kemudian run
2. pilih install for all user kemudian tekan Next. karena tujuannya agar aplikasi bisa digunakan oleh setiap user yang ada di pc. termasuk guest sekalipun.
- gambar 24.2 merupakan tampilan awal instalasi app python
3. pilih folder tempat instalasi python
- gambar 24.3 tampilan untuk memilih folder instalasi
4. klik next lagi untuk melanjutkan pemasangan
- gambar 24.4 merupakan tampilan komponen instalasi python
5. tunggu proses instalasi python beberapa saat



Gambar 24.2 tampilan awal instalasi

gambar 24.5 merupakan tampilan proses instalasi

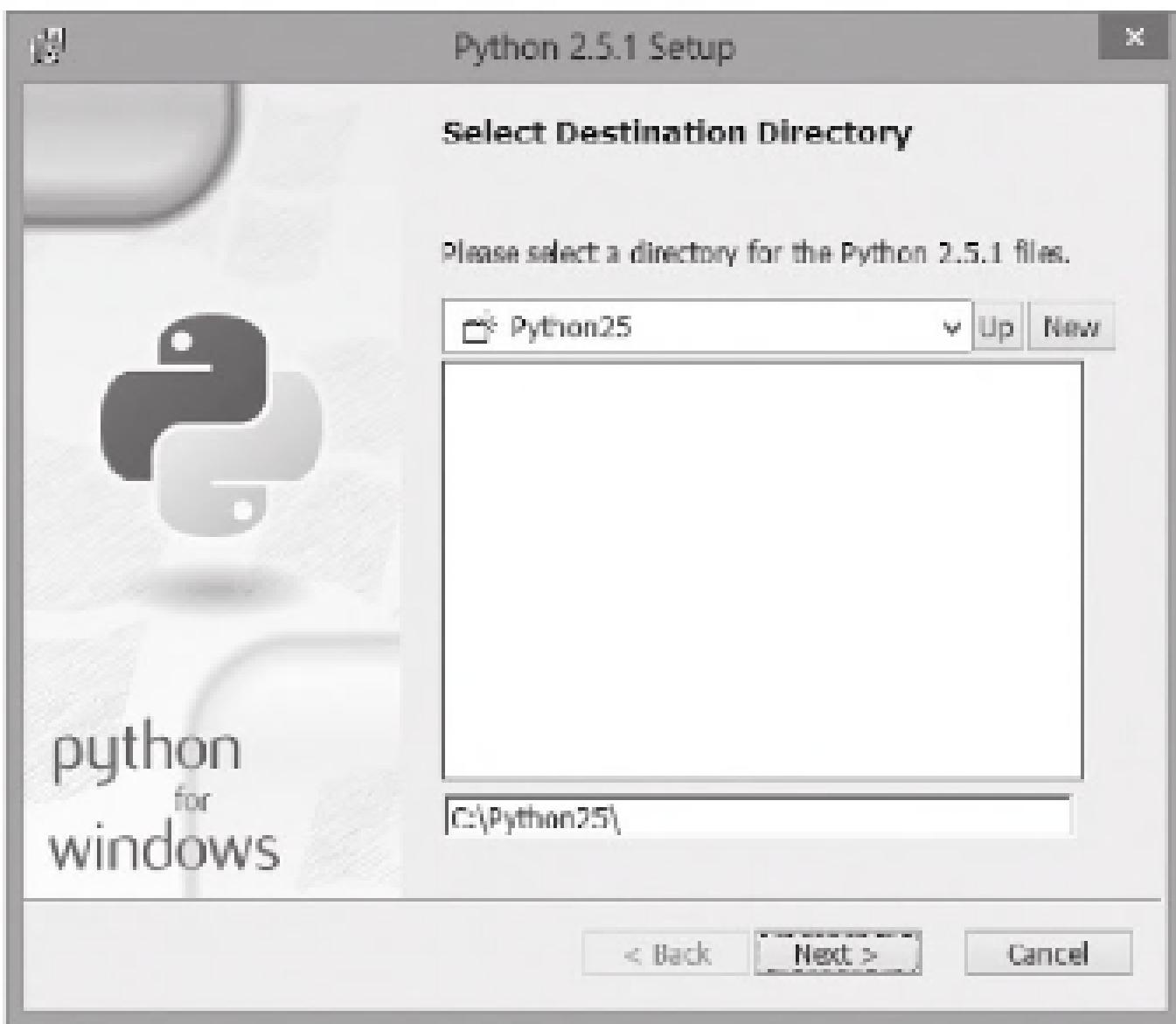
6. jika proses instalasi selesai, klik tombol finish
- instalasi 24.6 software telah selesai dan software python siap digunakan

24.4.1 Pemrograman Dalam Bahasa Python

Program adalah urut-urutan instruksi untuk menjalankan suatu komputasi. Komputasi dapat berupa matematis, seperti pencarian bilangan prima, persamaan kuadrat, atau yang lainnya. Akan tetapi juga dapat berupa pencarian dan penggantian teks dalam dokumen. Instruksi atau perintah atau statement pada masing-masing bahasa pemrograman dapat berbeda, namun beberapa instruksi dasar secara prinsip hampir disemua bahasa pemrograman sama, seperti :

1. Input mengambil data dari keyboard, mouse, file, atau dari device lain.
2. Output menampilkan data pada tampilan monitor atau pada device lain.
3. Math melakukan operasi dasar matematika seperti penambahan dan perkalian.
4. Conditional pemilihan suatu ondisi dan mengeksekusi sesuatu dengan statement selanjutnya.
5. Reptisi Operasi perulangan.

Masih banyak hal lain yang belum tercakup diatas, namun program-program bagaimanapun kompleksnya pasti akan terdiri kumpulan intruksi-intruksi di atas. Pemrograman merupakan proses yang kompleks yang memungkinkan terjadi kesalahan. Berbagai macam kesalahan dapat terjadi dalam pemrograman dinamakan bug. Sedangkan proses pencarian kesalahan dinamakan



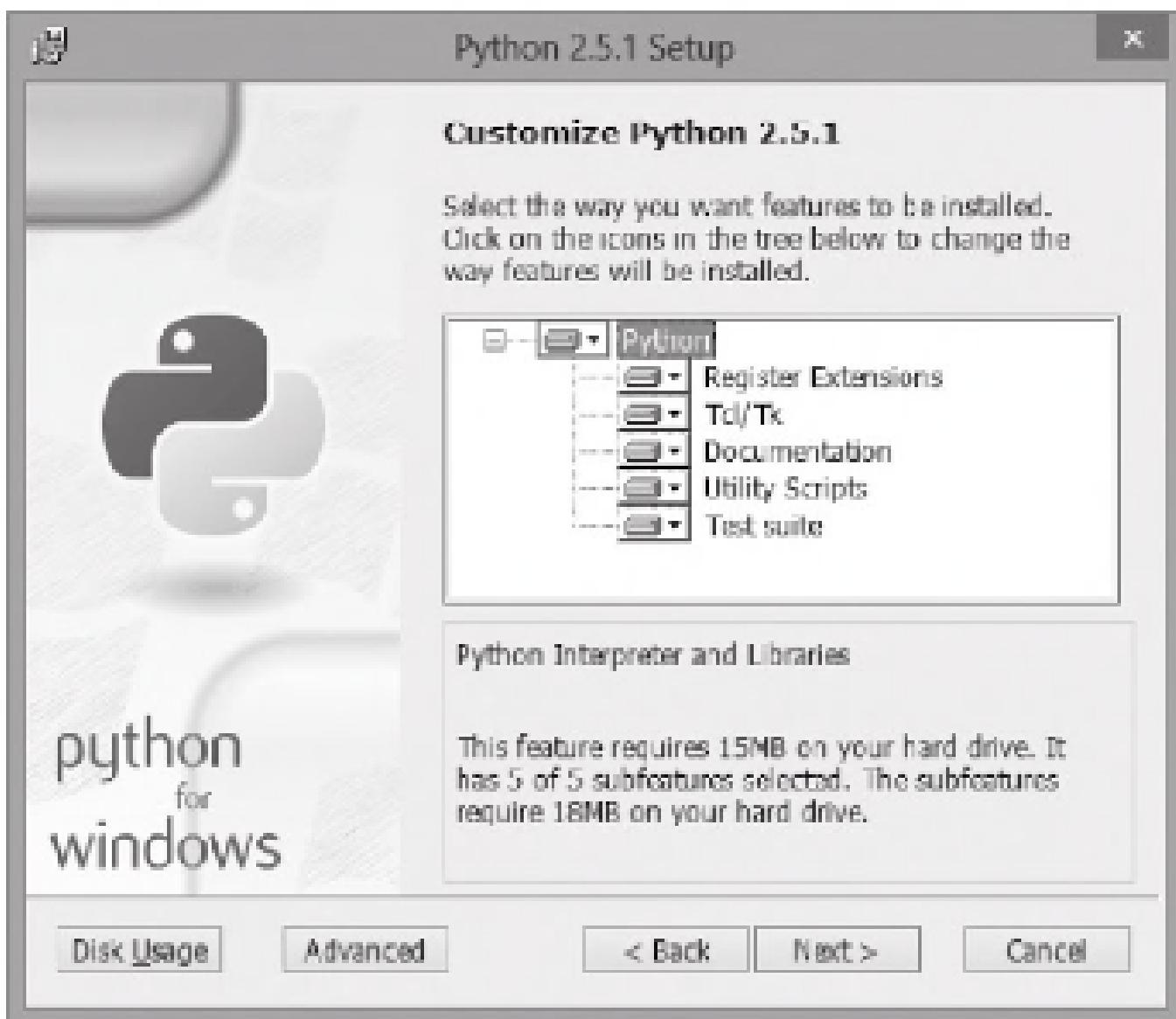
Gambar 24.3 pilih folder tempat instalasi

debugging. Dalam pemrograman, kesalahan dapat dibagi menjadi tiga macam, yakni kesalahan sintaks (syntax error), kesalahan run-time (run-time error) dan kesalahan dalam pemrograman dapat menjadikan pencarian kesalahan lebih cepat.

1. Kesalahan Sintaks Python hanya dapat dieksekusi jika dan hanya jika program tersebut sintaksnya telah sepenuhnya benar. jika tidak, maka proses akan berhenti dan memberi pesan kesalahan. Sintaks menunjukkan struktur program dan aturannya. Sintaks dalam bahasa Indonesia bisa diumpamakan sebuah kalimat yang harus diawali dengan huruf besar dan diakhiri dengan titik. Bila terjadi kesalahan sintaks dalam bahasa, maka beberapa pembaca tidak akan begitu mempermasalahkan, tetapi Python tidak demikian. Python harus ditulis dengan benar tanpa ada satupun kesalahan sintaks.
2. Kesalahan Run-time Kesalahan tipe kedua adalah kesalahan run-time. Disebut demikian karena kesalahan ini tidak akan muncul sebelum program dijalankan. Kesalahan ini juga sering disebut exception karena kesalahan ini biasanya menunjukkan sesuatu yang ganjil (dan tidak benar) terjadi.
3. Kesalahan Logika Kesalahan tipe ketiga adalah kesalahan logika atau semantik. Jika terjadi kesalahan tipe ini, program akan tetap berjalan dengan sukses tanpa pesan kesalahan, namun tidak menjalankan program dengan benar atau tidak menjalankan program sesuai dengan maksud yang diinginkan pemrogram.

24.4.2 methods

Metode pada objek adalah fungsi yang melekat pada kelas objek; Contoh sintaks Metode (argument) adalah untuk metode dan fungsi normal, yang sintaksis untuk Class. Method (contoh, argument). Metode Python memiliki selfparameter yang akurat untuk mengakses data contoh, berbeda dengan diri tersirat pada beberapa bahasa pemrograman berorientasi objek lainnya (misalnya Java, C ++ atau Ruby).



Gambar 24.4 komponen yang akan di install

24.4.3 typing

Python menggunakan bebek mengetik dan telah mengetikkan benda tapi nama variabel untyped. Ketik kendala tidak diperiksa pada waktu kompilasi; Sebaliknya, operasi pada objek mungkin gagal, menandakan bahwa Benda yang diberikan bukan tipe yang sesuai. Meski dengan ketahasiaan mengetik, Python sangat kuat diketik, melarang operasi yang tidak didefinisikan dengan baik (misalnya, menambahkan nomor ke a string) daripada diam-diam mencoba untuk memahami mereka. Python memungkinkan pemrogram untuk menentukan jenis mereka sendiri menggunakan kelas, yang paling sering digunakan untuk pemrograman berorientasi objek. Contoh kelas baru dibuat dengan memanggil kelas (misalnya, SpamClass () atau EggsClass ()), dan kelasnya sendiri contoh tipe metaclass (sendiri merupakan contoh dari dirinya sendiri), memungkinkan metaprogramming and reflection. Sebelum versi 3.0, Python memiliki dua jenis kelas: old-style dan gaya baru. Gaya lama kelas dieliminasi dengan Python 3.0, membuat semua kelas bergaya baru. Dalam versi antara 2.2 dan 3.0, kedua jenis kelas bisa digunakan. Sintaks dari kedua gaya adalah sama, yaitu Perbedaannya adalah apakah objek kelas diwarisi dari, secara langsung atau tidak langsung (semua gaya baru kelas mewarisi dari objek dan merupakan contoh tipe).

24.4.4 mathematics

Berbeda dengan beberapa bahasa pemrograman lainnya, pembagian bilangan bulat didefinisikan sebagai istilah bulat (round) menuju minus tak terhingga. Oleh karena itu $7 // 3$ adalah 2, tapi $(-7) // 3$ adalah -3. Ini seragam dan tetap: misalnya, ini berarti bahwa persamaan $(a + b) // b == a // b + 1$ selalu benar, sedangkan dalam bahasa seperti C, $(-6 + 7) / 7 == -6 / 7$. Ini juga berarti bahwa persamaan $b * (a // b) + a \% b == a$ berlaku untuk nilai positif dan negatif dari a. Namun, menjaga keabsahan persamaan ini berarti bahwa sementara hasil dari $\% b$ seperti yang diharapkan, pada interval setengah terbuka $[0, b)$, di mana b adalah bilangan bulat positif, ia harus berbaring dalam interval $(b, 0]$ ketika b negatif. Python menyediakan fungsi bulat untuk pembulatan pelampung ke bilangan bulat. Versi sebelum 3 digunakan round-away-from-zero: round (0.5) adalah 1.0, round (-0.5) adalah -1.0. [45] Python 3 menggunakan round-to-even: round (1.5) adalah 2.0, round (2.5) adalah 2.0. [46] Desimaltype / class



Gambar 24.5 tampilan proses instalasi

dalam module decimal (sejak versi 2.4) memberikan representasi numerik yang tepat dan beberapa mode pembulatan. Python memungkinkan ekspresi boolean dengan beberapa hubungan kesetaraan dengan cara yang konsisten dengan penggunaan umum dalam matematika. Misalnya, ekspresi $a < b \neq c$ menguji apakah a kurang dari b dan b kurang dari c . Bahasa yang diturunkan dari C menafsirkan ungkapan ini secara berbeda: di C, ungkapan pertama akan mengevaluasi sebuah \neq , menghasilkan 0 atau 1, dan hasilnya kemudian akan dibandingkan dengan c .

24.4.5 Pernyataan dan Arus Kontrol

Pernyataan Python antara lain :

1. Pernyataan if, yang secara kondisional mengeksekusi satu blok kode, bersama dengan yang lain dan elif (kontraksi yang lain-jika).
2. Untuk pernyataan, yang mengulangi objek yang berulang-ulang, menangkap setiap elemen ke variabel lokal untuk digunakan oleh blok terlampir.
3. Pernyataan sementara, yang mengeksekusi satu blok kode selama kondisinya benar.
4. Pernyataan percobaan, yang memungkinkan pengecualian diajukan di blok kode terlampir untuk ditangkap dan ditangani oleh pengecualian; Ini juga memastikan bahwa kode pembersihan di blok akhirnya akan selalu dijalankan terlepas dari bagaimana blok tersebut keluar.
5. Pernyataan kelas, yang mengeksekusi satu blok kode dan menempelkan namespace lokal ke kelas, untuk digunakan dalam pemrograman berorientasi objek.
6. Pernyataan def, yang mendefinisikan suatu fungsi atau metode.



Gambar 24.6 tampilan akhir instalasi

7. Pernyataan dengan (dari Python 2.5), yang menyertakan blok kode di dalam manajer konteks (misalnya, mengakuisisi alock sebelum blok kode dijalankan, dan melepaskan kunci setelahnya).
8. Pernyataan kelulusan, yang berfungsi sebagai PDN. Secara sintaktis diperlukan untuk membuat blok kode kosong.
9. Pernyataan tegas, digunakan saat melakukan debug untuk memeriksa kondisi yang seharusnya diterapkan.
10. Pernyataan hasil, yang mengembalikan nilai dari fungsi generator. Dari Python 2.5, yield juga merupakan operator. Formulir ini digunakan untuk menerapkan coroutines.
11. Pernyataan impor, yang digunakan untuk mengimpor modul yang fungsi atau variabelnya dapat digunakan dalam program saat ini. Python tidak mendukung pengoptimalan panggilan ekor atau kelanjutan kelas satu, dan menurut Guido van Rossum, tidak akan pernah.

Namun, dukungan yang lebih baik untuk fungsionalitas mirip coroutine disediakan di 2.5, dengan memperluas generator Python. Sebelum 2.5, generator adalah pemicu malas; Informasi dilewatkan secara tidak sadar dari generator. Seperti Python 2.5, adalah mungkin untuk menyampaikan informasi kembali ke fungsi generator, dan seperti pada Python 3.3, informasi dapat melewati beberapa tingkat tumpukan.

BAB 25

PEMROGRAMAN SIG VARIABEL PYTHON

25.1 Variabel

Pada sebagian besar dari bahasa pemrograman, nama suatu variabel menjelaskan suatu nilai dengan tipe data tertentu dan mempati alamat memory yang pasti. Variabel menyimpan data atau nilai yang dilakukan selama program dieksekusi, Nilai suatu variabel tersebut dapat diganti-ganti, namun pada tipe data selalu tetap ada. Tidak demikian dengan sebuah python dimana tipe datanya dapat diubah-ubah secara dinamis[52].

Variabel merupakan entitas yang memiliki nilai dan nilai tersebut berbeda satu sama yang lain. Variabel mengalokasikan memori untuk menyimpan nilai - nilai nya. Hal ini berarti ketika saat anda membuat variabel maka anda memesan beberapa atau sebuah ruang di memori tersebut. Variabel dapat digunakan untuk menyimpan bilangan bulat, desimal juga karakter. Python sangat mementingkan indentasi, sehingga kita perlu melakukan indentasi secara konsisten. Indentitas tersebut dipermudah dengan menggunakan tombol Tab dan dimulai dari kolom pertama untuk setiap blok baru.

Variab membutuhkan pengenal atau bisa disebut dengan identifier yg didefinisikan oleh pemrograman. Tanda pengenal atau identifier memiliki aturan-aturan dalam penulisannya seperti:

1. harus diawali huruf atau di garis bawah
2. tidak menggunakan aritmatika
3. jika terdiri dari dua kata atau lebih,makan antar katanya tidak boleh menggunakan spasi
4. tidak boleh menggunakan kata-kata yang sudah ada arti khususnya dalam bahasa Python
5. penggunaan huruf kecil dan besar harus dibedakan
6. panjang maksimal yaitu 32 karakter

Variabel merupakan nama yang menunjukkan nilai tertentu. Dalam Pyton, untuk membentuk variabel cukup memberi nama pada nlai yang kita inginkan. Statement yang melakukan hal tersebut disebut assignment.

```
>>> pesan = Hi, Apa kabar D43A
>>> n = 20
>>> pi = 3.14159
```

Contoh diatas menunjukkan tiga buah assigment. Contoh pertama memberi nilai “Hi, Apa kabar D43A” pada variabel pesan, kedua memberi nilai 20 pada variabel n, dan ketiga memberi sebuah nilai 3.14159 pada variabel pi. [53]

Variabel akan memiliki tipe sesuai dengan nilainya.

```
>>>type (pesan)
<Type string>
>>> type (n)
<type int>
>>> type (pi)
```

```
<type float>
```

Contoh lainnya:

```
>>>menit=59
>>>menit/60
0
```

Nilai yang telah diberikan oleh si variabel menit itu adalah 59, jika kita hitung, maka pembagiannya 59 oleh 60 akan menghasilkan angka 0,9833333. Tapi, dalam Python memberi nilai 0. Hal ini terjadi karena bahasa pemrograman Python menggunakan pembagian integer. Jika kedua operan adalah integer, maka hasilnya itu harus berupa integer juga, dan pembagian integer itu selalu dibulatkan ke bawah

Nilai variabel dapat tampil dengan perintah print:

```
>>> print pesan
Hi, Apa kabar D43A
>>> print n
20
>>>print pi
3.14159
```

Akan tetapi, jika pada command-line, maka dapat langsung dipanggil nama variabel tersebut :

```
>>> pesan
Hi, Apa kabar D43A
```

Perintah dari print juga dapat memberi lebih dari satu nilai dalam satu baris:

```
>>> print Nilai dari pi adalah ,pi
Nilai dari pi adalah 3.14159
```

koma untuk memisahkan nilai dan variabel.[53]

Ada 2 cara untuk membuat variabel. Cara yang pertama yaitu variabel langsung dengan nilai disebut dengan inisialisasi. Sedangkan cara yang kedua dengan memasukkan nilai pada variabel yang biasa disebut penempatan.[54]

Variabel python tidak perlu untuk melakukan deklarasi eksplisit untuk memesan ruang memori. Deklarasi tersebut terjadi secara otomatis ketika menetapkan nilai ke variabel nya. Tanda sama (=) dapat digunakan untuk memberikan nilai pada variabel.

Operan di sebelah kiri operator = nama dari variabel dan operan di sebelah kanan operator = nilai yang disimpan dalam variabel Python memiliki lima tipe data standar - Nomor Tali Daftar tupel Kamus

Variabel pada Python memiliki beberapa aturan seperti :

- Case Sensitive : penggunaan huruf besar dan huruf kecil yang dibedakan.
- Harus dimulai dengan underscore (_) atau huruf biasa, setelah itu dapat diikuti dengan huruf, angka atau underscore (_).
- Tidak boleh mengandung karakter special seperti !,@,#,\$ dan lainnya.
- Hanya dapat menggunakan suatu variable setelah kita memberikan nilai ke dalamnya atau telah dilakukan assignment.
- Setiap variable akan menyimpan referensi ke suatu objek dalam memory.[54]

Variabel adalah sebuah nama yang selalu menunjukkan nilai tertentu. Dalam bahasa pemrograman Python, untuk membentuk sebuah variabel itu cukup hanya memberi nama pada nilai yang akan dibuat. Ini disebut dengan Assignment. Contoh :

```
>>>pesan="Halo, Semuanya"
```

Contoh di atas membuktikan bahwa itu adalah sebuah assignment, yang memberi kan sebuah nilai Halo, Semuanya:[53]
contoh lainnya:

```
>>> c = 2 # c bilangan bertipe integer
>>> print c
2
>>> c = c * 2.0 # Sekarang c bilangan bertipe float
>>> print c
4.0
```

ke suatu object dalam memory. Untuk lebih jelasnya dapat kita lihat contoh berikut ini :

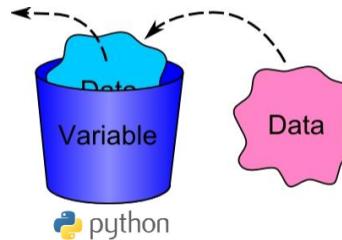
```
>>> a = 2
>>> b = 3
>>> a,b = b,a
>>> a
3
```

```
>>> b
2
>>>
```

Beberapa bentuk if adalah sebagai berikut :

```
a. if tunggal
if x == 1:
print x bernilai 1
b. if dengan else
if x == 1:
print x bernilai 1
else:
print x tidak bernilai 1
c. if dengan pilihan if lainnya
if x == 1:
print x bernilai 1
elif x == 2:
print x bernilai 2
else:
print x tidak bernilai 1 atau 2
d. if di dalam if
if x == 1:
if y == 1:
print x dan y bernilai 1
```

Tulisan `a = 2` artinya kita memberi nilai pada variabel a dengan angka 2 yang bertipe integer (bilangan bulat). Statemen berikutnya adalah melakukan operasi perkalian `a * 2.0` lalu hasilnya disimpan pada variabel yang sama yaitu variabel a. Dengan demikian nilai a yang lama akan diganti dengan nilai a yang baru, yaitu yang berasal dari operasi yang terakhir. Sebagai Konsekuensi atau hasil dari operasi yang dirancang atau dibuat tersebut, sekarang variabel a memiliki tipe data float, suatu tipe dapat berkaitan dengan bilangan pecahan atau desimal. Nilai variabel b menghasilkan nilai 4.0 atau empat.



Gambar 25.1 gambar yang menggambarkan keadaan variabel pada python

Tanda dari pagar (#) menyatakan awal dari suatu komentar. Komentar adalah bagian dari script program python yang tidak akan dieksekusi oleh interpreter. Python mementingkan indentasi, sehingga perlu indentasi yang konsisten. Indentasi dipermudah sesuai penggunaan tombol Tab dan dimulai dari kolom pertama untuk setiap blok baru. Bahasa pemrograman Python suatu fasilitas shell di Linux, sehingga kita untuk mencoba penggunaan Python secara leluasa. Lokasi instalasi Python default pada distribusi Linux. Keunggulan Python adalah :

1. Python is powerful and fast = suatu kumpulan modul-modul yang sangat baik dan dapat menangani secara praktis setiap domain masalah
2. Python plays well with others = Python bisa berintegrasi dengan Component Object Model (COM)
3. Python runs everywhere = versi Python berjalan di .NET, Java Virtual Machine dapat melihat bahwa sumber yang sama dapat berjalan tanpa perubahan berarti pada setiap sistem operasi tersebut.
4. Python is Open = Python dilisensi open source membuat Python digunakan dan disebarluaskan secara free,
5. Python is friendly and easy to learn = Dokumentasi yang lengkap merupakan salah satu fasilitas Python yang disenangi penggunanya. Apabila pembaca melakukan instalasi Python

Dalam menulis program, akan menggunakan code yang pernah kita buat atau ditulis sebelumnya, pasti kita gunakan kembali, dengan beberapa nilai berbeda.

Tentu saja tidak mungkin menuliskan kembali kode yang ingin dipanggil ulang tersebut. Solusinya, supaya dapat dikelompokan kode-kode yang sering dipanggil ulang dalam suatu kelompok.

Selain itu dapat memecah masalah-masalah yang besar menjadi masalah-masalah yang lebih kecil. Dalam C atau bahasa pemrograman lain, biasanya digunakan istilah function.

Kemampuan dari python dalam mengelola tipe data sangat baik. Untuk melakukan pendeklarasian suatu variabel dapat dilakukan secara langsung tanpa menyebutkan tipe - tipe data, ini yang membedakan sebuah python dengan bahasa lain. Python akan menentukan tipe-tipe data tersebut secara otomatis. Python juga dapat mendukung konversi dan perhitungan antar semua tipe data tersebut dengan keakuratan yang tinggi. Python membagi sebuah tipe data ke dalam 2 jenis bilangan (semua tipe yang berhubungan dengan angka murni) dan string. Untuk tipe data dalam rumpun bilangan termasuk di dalamnya adalah integer, long, float, oktal, hexadimal dan bilangan kompleks. Hal-hal yang harus diperhatikan :

- Untuk suatu bilangan oktal dan hexa masing-masing diawali dengan 0 dan 0x
- Untuk bilangan panjang diakhiri menggunakan karakter dari l ataupun L
- Untuk bilangan float dapat menggunakan e atau E pada eksponensial
- Untuk bilangan kompleks dapat dibagi menjadi bagian real dan imajiner, dan diakhiri dengan j atau J Operator untuk tipe dalam rumpun bilangan.[55]

25.1.1 Membuat Perubah Variabel

Variabel atau perubah memiliki suatu symbol yang dapat dimuati oleh sembarang himpunan dari bilangan tersebut. Dalam pengertian dari komputasi sebuah nama dapat digunakan untuk menyimpan suatu nilai dengan kapasitas tertentu dan alamat tertentu dalam sebuah memori komputer tersebut. Variabel merupakan pendaftaran dari tipe data bagi variabel itu, konstanta dan parameter sering digunakan untuk sebuah program agar dapat mempunyai alamat penyimpanannya dan kapasitas dari data nya dalam memori komputer. Dalam membuat variabel kita hindari spasi dan dapat menggunakan sebuah karakter khusus, selain itu nama dalam sebuah kata cadangan Python (seperti input, eval, if, elif, for, def, dan lain-lain) tidak dapat menjadi variabel.[56]

Nama variabel atau disebut juga dengan identifier dalam bahasa pemrograman Python juga dapat berupa kumpulan dari huruf (letter) maupun angka (digit) yang dengan cara membedakan huruf kecil dan juga huruf besar, Karakter pertama pada identifier harus berupa huruf dan juga perlu diketahui bahwa penggunaan karakter garis bawah itu juga dapat digunakan. Kesalahan sintax akan muncul apabila dalam penamaan sebuah variabel itu tidak sesuai dengan aturan yang ada. Contoh :

```
>>>876swafm="Radio Swa"
SyntaxError: invalid syntax
>>>more$=1000000
```

SyntaxError: invalid syntax Baris pertama itu salah, karena pada 876swafm karakter pertamanya itu bukanlah huruf. Pada baris ketiga juga salah karena pada more\$ berisi karakter dolar.

Tidak seperti pemrograman pada lainnya, variabel pada Python tidak harus dideklarasikan secara eksplisit. Pendeklarasian variabel terjadi secara otomatis ketika kita memberikan sebuah nilai pada suatu variabel. Tanda sama dengan (=) dapat digunakan untuk memberikan value-nilai pada suatu variabel. Operan di sebelah kiri dari tanda (=) adalah nama variabel, sedangkan operan yang sebelah kanan dari tanda (=) adalah nilai yang diberikan pada variabel.[55]

```
>>>harga = 100
>>>diskon = 30
>>>harga - diskon
70
```

Pada contoh di atas tersebut, 100 dan 30 adalah suatu nilai yang diberikan untuk variabel harga dan variabel diskon itu. Sedangkan pernyataan dari harga-diskon akan menghitung selisih antara harga dengan diskon. Variabel juga dapat menyimpan suatu nilai berupa teks (dibaca string).

```
>>>a = 'sekolah'
>>>b = 'dasar'
>>>a + b
'sekolahdasar'
```

Variabel juga bisa menyimpan dua nilai string atau lebih dengan menggunakan operator (+).

```
>>>c = 'Py' + 'thon'
>>>c
'Python'
```

Jika kita telah melakukan penilaian pada variabel, kita dapat menggunakan variabel tersebut pada yang lainnya.

```
>>>a = 2
>>>a = a + 3
>>>a
5
```

Selain itu, juga dapat memberikan sebuah nilai untuk beberapa variabel.

```
>>>p=q=r=1
>>>p
1
>>>q
1
>>>r
1
```

Selain itu, kita dapat memberikan beberapa nilai untuk beberapa variabel (disebut multiple assignment).

```
>>>x, y, z = 1, 2, 'belajar Python'
>>>x
1
>>>y
2
>>>z
'belajar Python'
```

Bentuk lain dari sebuah contoh di atas, kita juga dapat menggunakan tanda kurung-buka atau kurung-tutup.

```
>>>(x, y, z) = (1, 2, 'Belajar Sebuah Python')
```

Cara di atas, dapat juga kita gunakan untuk pertukaran nilai variabel.

```
>>>(x, y) = (10, 20)
>>>x
10
>>>y
20
>>>(x, y) = (y, x)
>>>x
20
>>>y
10
```

Contoh-contoh pada prompt Python adalah sebagai berikut

```
>>> a = 1
>>> a
1
>>> b = 2
>>> b
2
>>> c = a + b
>>> c
3
>>> d = a - b
>>> d
-1
>>> print Nilai d adalah : , d
Nilai d adalah : -1
>>> print Nilai c adalah : , c
Nilai c adalah : 3
>>> e
Traceback (most recent call last):
File <stdin>, line 1, in ?

```

```
Python yang sederhana :
$ vi contoh-script-01.py
#!/usr/bin/python
a = 1
print Nilai a adalah : , a
simpan script Anda dengan :
:wq
```

Secara default, script Python disimpan dengan ekstensi .py

25.1.2 Aturan Penulisan Variabel

- Nama variabel bisa diawali dengan menggunakan huruf atau garis bawah (_), contoh: nama, _nama, namaKu, nama_variabel.
- Karakter selanjutnya dapat menggunakan berupa huruf, garis bawah (_) atau angka, contoh: __nama, n2, nilai1.
- Karakter pada nama variabel bersifat sensitif (case-sensitif). Artinya huruf besar dan kecil dibedakan. Misalnya, variabel_Kita dan variabel_kita, keduanya adalah variabel yang berbeda.
- Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python seperti if, while, for, dsb.

25.1.3 Menghapus Variabel

Ketika sebuah atau suatu variabel tidak dibutuhkan lagi, maka kita bisa menghapusnya dengan fungsi del(). Contoh:

```
>>> nama = "petanikode"
>>> print nama
petanikode
>>> del(nama)
>>> print nama
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'nama' is not defined
>>>
```

Pada perintah terakhir, kita akan mendapatkan NameError. Artinya variabel tidak terdapat di dalam memori alias sudah dihapus.

25.1.3.1 Penempatan variable pada yang semestinya Misalkan sebuah data pribadi dapat berisi nama, alamat, umur, tempat lahir, tanggal lahir, indeks prestasi kumulatif akan memberikan 6 (enam) buah variabel tipe datanya.

```
# CONTOH PROGRAM 2: #
# DATA PRIBADI ANDA #
=====
Nama = input("Nama Anda :")
Alamat = input("Alamat Tinggal Anda :")
Umur = input("Umur Anda :")
TL = input("Tempat Lahir Anda :")
TgL = input("Tanggal Lahir Anda :")
IPK = input("IPK Anda :")
"Selesaikan dengan datanya"
print("Nama Anda : ", Nama)
print("Alamat Tinggal Anda : ", Alamat)
print("Umur Anda : ", Umur)
print("Tempat Lahir Anda : ", TL)
print("Tanggal Lahir Anda : ", TgL)
print("IPK Anda : ", IPK)
```

```
==== RESTART =====
>>>
Nama Anda : Hanif
Alamat Tinggal Anda : Jl. Merdeka No 34 Palembang
Umur Anda : 23
Tempat Lahir Anda : Palembang
Tanggal Lahir Anda : 17 April 2008
IPK Anda : 3.75
Nama Anda : Hanif
Alamat Tinggal Anda : Jl. Merdeka No. 34 Palembang
Umur Anda : 23
Tempat Lahir Anda : Palembang
Tanggal Lahir Anda : 17 April 2008
IPK Anda : 3.75
>>>
```

Gambar 25.2 Tampilan Contoh Input/Output Tipe Data String

Pada Gambar 2.2 25.3 terlihat input/output pada tipe data bilangan dengan hasil yang berbeda tipe bilangan yaitu tipe dari integer (bilangan bulat) atau float (bilangan berkoma).[56]

25.1.3.2 Memberikan nilai ke dalam variable Lakukan konstanta dari permasalahan berikut! Menjumlahkan hasil dari total harga pada saat konsumen membeli beberapa barang.

Langkah 1: Inisiasi Persoalan Variabel konstanta input : kode_barang, nama_barang, harga_satuan_barang, jumlah_barang_beli, total_harga_transaksi = 0 Proses : harga_beli_barang = harga_satuan_barang * jumlah_barang_beli total_harga_transaksi = harga_beli_barang + total_harga_transaksi Output : total_harga_transaksi

Langkah 2: Menetapkan Tipe Data kd_brg, nama_brg bertipe data string jum_brg merupakan tipe data integer harga_satuan, harga_beli, total_hrg_brg bertipe data float

Langkah 3 : Kode program

25.1.3.3 Mencetak nilai dalam variable Mencetak nilai dari sebuah variabel menggunakan pernyataan print, perhatikan contoh berikut ini.

```

# LATIHAN1PY - C:/Python34/Latihan1py (3.4.3)
File Edit Format Run Options Window Help
# CONTOH PROGRAM 2 #
# MENGHITUNG BILANGAN #
=====
x1 = eval(input("X1 :"))
x2 = eval(input("X2 :"))
x3 = eval(input("X3 :"))
x4 = eval(input("X4 :"))
Jumlah=x1+x2+x3+x4
Kali=x1*x2*x3*x4
"Silahkan diinput datanya...."
print("Jumlah Semua Bilangan = ",Jumlah)
print("Kali Semua Bilangan = ",Kali)
Jumlah = Jumlah + 0.5
print("Jika ditambah 0.5 = ",Jumlah)
Kali = Kali * 0.5
print("Jika dikali 0.5 = ",Kali)
|
```

Running :

```

>>> ===== REST
>>>
X1 :2
X2 :5
X3 :7
X4 :9
Jumlah Semua Bilangan = 23
Kali Semua Bilangan = 630
Jika ditambah 0.5 = 23.5
Jika dikali 0.5 = 315.0
>>>
```

Gambar 25.3 Tampilan Contoh Input/ Output Tipe Data Bilangan

```

# latih4.py - C:/Python34/Latih4.py (3.4.3)
File Edit Format Run Options Window Help
total_hrg_brg = 0.0
kd_brg=input("Kode barang =")
nama_brg= input("Nama barang =")
harga_satuannyaeval(input("Harga satuan barang =Rp."))
jmlh_brgeval(input("jumlah barang yang dibeli="))
harga_beli = harga_satuannya * jmlh_brg
total_hrg_brg = harga_beli + total_hrg_brg
print("Total Harga yang dibayar Rp.",total_hrg_brg)
```

Running :

```

>>> ===== REST
>>>
Kode barang =12345
Nama barang =handuk
Harga satuan barang =Rp.50000
jumlah barang yang dibeli=3
Total Harga yang dibayar Rp. 150000.0
>>>
```

Gambar 25.4 kode program

```

>>> x="2"
>>> print(x)
2
>>> type(x)
<class 'str'>
>>> Y=4.0
>>> print(Y)
4.0
>>> print(type(x))
<class 'str'>
>>> print(Z=2.0)
2.0
>>> print(type(Z))
<class 'float'>
>>> print("data X dikonversi menjadi float untuk dijumlahkan dengan tipe data Y, hasilnya yang diperoleh berupa float juga")
>>> print("data Z dikonversi menjadi float untuk dijumlahkan dengan tipe data Y, hasilnya yang diperoleh berupa float juga")
>>> print(1+2)
SyntaxError: invalid syntax
>>> print(True)
True
>>> print(Tanda != artinya tidak sama dengan)
Tanda != artinya tidak sama dengan
>>> print(False)
False
>>> print(Tanda == artinya sama dengan)
Tanda == artinya sama dengan
>>>
```

Gambar 25.5 Tampilan Contoh Konversi Tipe Data String dan Integer

25.1.4 Bilangan integer dan float

Perbedaan tipe bilangan ini bisa dapat jadi potensi menimbulkan masalah. Ini contohnya

```

>>> 1/2 # bilangan integer dibagi bilangan integer
0 # tentu saja kita ini keliru, mestinya 0.5
>>> 1/2.0 # bilangan integer dibagi bilangan float
0.5 # kali ini hasilnya tepat
```

25.1.5 List

List adalah sejumlah atau suatu object yang dipisahkan oleh tanda koma (,) dan diapit oleh kurung siku ([]). Begini contohnya:

```

>>> a = [1.0, 2.0, 3.0] # membuat list
>>> a.append(4.0) # tambahkan 4.0 kedalam list
>>> print a
[1.0, 2.0, 3.0, 4.0]
>>> a.insert(0,0.0) # sisipkan 0.0 pada posisi 0
>>> print a
[0.0, 1.0, 2.0, 3.0, 4.0]
>>> print len(a) # menentukan panjang list
5
```

Jika kita memberikan suatu statemen $c = b$, maka itu tidak berarti bahwa variabel c terpisah dengan variabel a . Di python, statemen tersebut dapat diartikan hanya sebagai pemberian nama lain (alias) kepada variabel a . Artinya, perubahan yang terjadi baik itu di a ataupun di b , maka hasil akhir mereka berdua akan sama saja. Setiap perubahan terjadi di c akan berdampak di b . Untuk meng-copy a secara independen, gunakan statemen $d = a[:]$. Berikut adalah beberapa contoh lambang atau tanda untuk melengkapi setiap statement pada variabel yang ada25.6.

Table 1-1. Math Operators from Highest to Lowest Precedence

Operator	Operation	Example	Evaluates to...
$**$	Exponent	$2 ** 3$	8
$\%$	Modulus/remainder	$22 \% 8$	6
$//$	Integer division/floored quotient	$22 // 8$	2
$/$	Division	$22 / 8$	2.75
$*$	Multiplication	$3 * 5$	15
$-$	Subtraction	$5 - 2$	3
$+$	Addition	$2 + 2$	4

Gambar 25.6 Tanda operasi pada python

PEMROGRAMAN SIG LOOPING PYTHON

26.1 Perulangan Pada Python

Perulangan dalam bahasa pemrograman berfungsi untuk memerintahkan komputer melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python, yaitu perulangan dengan for dan while.

26.1.1 While dan For

Perulangan while disebut dengan uncounted loop (perulangan yang tak terhitung), sementara perulangan for disebut dengan counted loop (perulangan yang terhitung). Perbedaannya adalah ada statement while, biasanya memiliki ciri berupa pengecekan kondisi dan perulangan dilakukan diawal. Sedangkan pada statement for, memiliki ciri berupa inisialisasi perulangan dilakukan diawal statement dan perulangan tersebut akan berhenti ketika syarat / kondisi yang telah ditentukan terpenuhi[54].

26.1.2 Perintah Break, Continue dan Pass Perintah Break

26.1.2.1 Perintah Break Perintah break dipakai untuk menghentikan proses berjalananya iterasi atau perulangan pada statement for atau while[57]. Dan semua kode setelah pernyataan break akan segera diabaikan. Pernyataan break ini dapat kita gunakan pada pengulangan while dan for. Statement yang berada di bawah break tidak akan dieksekusi dan program akan keluar dari proses looping. Contoh break :

```
>>> x = 1
>>> while x < 5:
...     if x == 3:
...         break
...     print x
...     x = x+1
... else:
print "Loop sdh selesai dikrjkn"
...
1
2
```

26.1.2.2 Perintah Continue Perintah continue dapat dipakai untuk meloncati sebuah perulangan, maksudnya adalah intruksi yang seharusnya dapat dilewati, hal ini berarti instruksi tidak akan dijalankan[57]. pernyataan continue akan dilakukan pengulangan mulai dari awal lagi. Dan mengabaikan semua kode yang tersisa pada loop untuk menuju ke awal loop lagi. Misal:

```
for(int i = 1; i <= 100; i++) {
    if(i % 2 == 0) {
        continue;
    }
```

```
System.out.println(i);
}
```

Jika program tersebut dijalankan, maka hasilnya akan menampilkan angka-angka ganjil saja, hal ini disebabkan karena saat nilai i merupakan angka genap, maka statement continue akan membuat program tidak menampilkan angka genap.

26.1.2.3 Perintah Pass Sebenarnya perintah pass tidak memiliki fungsi yang sangat penting. Dan bahkan sangat jarang digunakan oleh programmer. Jadi perintah pass ini sebenarnya hanya mengisi kekosongan saja, agar program tidak eror nantinya. perintah pass akan menyebabkan program tidak melakukan tindakan. Digunakan untuk mengabaikan sesuatu statement perulangan, pengkodisian, class dan fungsi yang belum didefinisikan badan programnya agar tidak terjadi error.

```
for i in range (5) :
    if i == 5 :
        pass
    print(i)
```

Jadi seperti yang sudah dikatakan sebelumnya, perintah pass ini memiliki fungsi untuk mengisi kekosongan dari sebuah penyeleksian ataupun perulangan.

26.1.2.4 Perintah return Perintah return dapat menghentikan suatu proses dari fungsi sebelum mengakhiri fungsi tersebut. Alasan menggunakan perintah return adalah jika menemui sebuah kesalahan kondisi, yang berarti nilai suatu fungsi tersebut mengembalikan nilai null (kosong) : import math

```
def print_log (x):
    if x <= 0 :
        print x,
    "Bilangan lebih sama dengan 0"
    return
hasil = math.log (x)
print "Hasil log dari", x, "adalah", hasil
```

Fungsi print_log() mengambil sebuah parameter x. Hal yang dilakukan pertama kali adalah memeriksa apakah nilai x lebih kecil atau sama dengan 0, yang dapat menghasilkan pesan kesalahan jika di proses dalam instruksi perintah selanjutnya maka diberi perintah return untuk keluar dari fungsi tersebut. Alur jalannya program segera dikembalikan ke pemanggil dari fungsi tersebut, dan instruksi - instruksi berikutnya tidak akan dijalankan. Perhatikan! untuk memanggil fungsi dari modul math harus menggunakan perintah import `nama-modul`.

26.1.2.5 Rekursif Telah disebutkan sebelumnya, bahwa kita dapat memanggil suatu fungsi di dalam fungsi lainnya, dan Anda telah melihat beberapa contoh - contohnya. Kita juga dapat memanggil fungsi itu sendiri yang kemudian dikenal dengan istilah rekursif. Mungkin sekilas hal itu tidak memberi alasan mengapa rekursif ini adalah hal yang baik, tetapi akan berubah menjadi program yang menarik. Sebagai contohnya, lihat fungsi berikut :

```
def hitung_mundur (x):
    if x == 0 :
        print "Sudah nol koq!"
    else :
        print x
        hitung_mundur (x-1)
```

fungsi diatas menampilkan program hitung mundur dari nilai parameter x yang diberikan, parameter tersebut seharusnya sebuah bilangan integer yang positif, dan jika nilai x sama dengan 0 akan menampilkan string yang memberitahu bahwa nilai x adalah 0. Kalau tidak nol(0) maka akan memanggil fungsi itu sendiri dan memberi nilai x-1 sebagai parameternya.

Prosesnya adalah sebagai berikut, jika kita memanggil fungsi tersebut dengan nilai 3, maka nilai 3 akan di check apabila bukan nol (0) maka akan di cetak, dan memanggil fungsi itu sendiri dengan parameter 3-1, yaitu nilai 2, kemudian nilai 2 akan di periksa apakah nilai 2 sama dengan 0, jika bukan maka akan di cetak, dan memanggil fungsi tersebut dengan nilai parameter 1, di check kembali bila bukan nol (0) maka akan akan memberikan parameter x-1 ke fungsi itu sendiri, setelah itu maka fungsi tersebut di beri paramater 0 maka string Sudah nol koqdicetak, kemudian kembali lagi ke fungsi sebelumnya dengan nilai 1, kembali ke nilai 2,kembali ke nilai 3. Jadi tampilan hasilnya akan seperti berikut.

```
2
```

```
1
```

Sudah nol koq!

Hal ini akan berbeda jika perintah print diletakkan setelah pemanggilan fungsi rekursif itu sendiri. Misalnya :

```
def hitung_maju(x):
```

```
    if x == 0 :
```

```
        print "Sudah nol, Mulai!"
```

```
    else :
```

```
        hitung_maju (x-1)
```

```
        print x
```

maka tampilannya akan menjadi :

```
Sudah nol, Mulai!
```

```
1
```

```
2
```

```
3
```

26.1.3 Nested Loop

Nested Loop (Perulangan Bertingkat) adalah semua tipe perulangan yang dapat dipakai di dalam perulangan yang lain. Jadi Perulangan for bisa dipakai di dalam for yang lain, perulangan for bisa berada di dalam perulangan while, perulangan while bisa dipakai di dalam perulangan while yang lain, dan perulangan while bisa di dalam perulangan for.

26.1.3.1 Contoh Penggunaan Nested Loop

Format nested loop (for di dalam for)

```
For iterasi_var_1 in urutan_1:  
    Statements_untuk_perulangan_for_yang_di_luar  
    ...  
    For iterasi_var_1 in urutan_2:  
        Statements_untuk_perulangan_for_yang_di_dalam  
        ...  
        Statements_untuk_perulangan_for_yang_di_luar  
    ...
```

Format nested loop (while di dalam while)

```
While expressions:  
    Statements_untuk_perulangan_while_yang_di_dalam  
    ...  
    Statements_untuk_perulangan_while_yang_di_luar  
    ...
```

Contoh:

```
X = int(input(Masukkan jumlah bariss: ))  
For i in range (x) :  
    For j in range(i+1):  
        Print(*, end=)  
        Print()
```

Saat di Run Module maka :

```
Masukkan jumlah bariss: 5 (inputkan 5)  
*
```

```
**
 ***
 ****
*****
```

Muncul 5 baris isi bintang

26.1.3.2 Nested Loop for Nested Data Disini kita memiliki list data dari murid-murid. Jadi, setiap murid memiliki nama yang dipasangkan dengan list subyek(mata pelajaran) yang mereka ambil. Dan akan mencetak setiap nama murid, dan jumlah dari subyek (mata pelajaran) yang mereka ambil

```
students = [
    ("John", ["TIK", "IPS"]),
    ("Vusi", ["Matematika", "TIK", "IPA"]),
    ("Jess", ["TIK", "Bahasa Indonesia", "Ekonomi", "Pendidikan Agama Islam"]),
    ("Sarah", ["Biologi", "Matematika", "Ekonomi", "Kimia"]),
    ("Zuki", ["Sosiologi", "Ekonomi", "Biologi", "Matematika", "Bahasa Inggris"])]
```

```
for (name, subjects) in students:
    print(name, "takes", len(subjects), "courses")
```

Lalu, setelah dijalankan (run) maka akan tampil seperti ini:

```
John takes 2 courses
Vusi takes 3 courses
Jess takes 4 courses
Sarah takes 4 courses
Zuki takes 5 courses
```

26.1.4 While Loop

Perintah While pada python biasanya memiliki ciri berupa pengecekan kondisi dan perulangan dilakukan diawal[54]. Alur prosesnya adalah ketika sebuah program dijalankan dan kemudian menemukan sebuah kondisi dimana menggunakan loop atau perulangan while, jika kondisi true maka statment itu akan dieksekusi kemudian akan di cek lagi kondisinya. Setelah selesai statmentnya masih true dieksekusi lalu akan mengecek lagi kondisinya dan terus seperti itu, dan jika false statmentnya maka akan keluar dari perulangan dan akan melanjutkan ke program selanjutnya.

26.1.5 Perulangan do-while

Perulangan do-while merupakan perulangan yang mirip dengan perulangan while namun perbedaannya[57], pada perulangan do-while, maka minimal instruksi akan dijalankan sekali. Contoh statement do-while:

```
int jumlah = 100;
do{
    System.out.println(jumlah);
    jumlah++; // naikkan jumlah
}while(jumlah <= 10);
```

Jika program itu dijalankan, maka akan menghasilkan keluaran 100, yang artinya walaupun kondisinya salah, namun minimal isi instruksi akan dijalankan sekali, hal ini disebabkan karena proses do-while berbeda dengan while, yang dimana do-while pertama melakukan instruksi setelah itu baru mengecek kondisi, sedangkan while pertama mengecek kondisi baru melakukan instruksi.

26.1.6 Perulangan (for loop)

FOR Loop dipakai untuk melakukan perulangan atau iterasi mencapai batas atau jarak yang telah ditentukan/citesantoso2009bahasa. Kegunaan

1. Ketika ingin pergi ke item urutan tertentu seperti pada list atau string
2. Ketika ingin melakukan perulangan kode beberapa kali

For interaksi_var in ururan Statements Print(bukan bagian perulangan)

26.1.7 For Loop

Perulangan for disebut juga counted loop *perulangan yang terhitung*. Pengulangan for digunakan untuk pengulangan dengan muatan yang banyak[50]. keistimewaan perulangan pada for adalah , perulangan dapat di hentikan pada saat kondisi tertentu. pada python, statemen for bekerja mengulang berbagai macam tipe data sekuensial seperti pada list, string dan tuple Contohnya Seperti :

```
for a in range(0, 10):
print a
```

Hasil Outputnya :

```
python for.py
0
1
2
3
4
5
6
7
8
9
```

26.1.8 While Loop

perulangan while disebut juga uncounted loop *perulangan yang tak terhitung*. Pengulangan while biasanya digunakan untuk sesuatu yang ga pasti. Contohnya Seperti :

```
a = 0
while true:
if a < 10:
print "saat ini a bernilai: ", a
a = a + 1
else a >= 5:
break
```

Hasil Outputnya :

```
python while.py
saat ini a bernilai: 0
saat ini a bernilai: 1
saat ini a bernilai: 2
saat ini a bernilai: 3
```

26.1.9 For looping with list

Contohnya Seperti :

```
hero_dota2_character = ["Mirana", "Axe", "Tusk"]
for character in hero_dota2_character:
print character
```

Hasil Outputnya :

```
python for-list.py
Mirana
Axe
Tusk
```

26.1.10 Infinite Loop

Infine Loop adalah kondisi dimana program atau statement akan terus mengeksekusi tanpa berhenti. Kondisi tersebut dapat dihentikan dengan menekan tombol CTRL+C.

Di bawah ini merupakan contoh program Infinite Loop :

```
# Nama file: infinite_loop.py
# Program menampilkan tulisan Python tanpa henti

flag = 1

while (flag): print ("Python")
print ("Good bye!")
```

26.1.11 break and continue statements

Jeda digunakan untuk for loop atau while loop, sedangkan terus digunakan untuk melewati blok saat ini, dan kembali ke pertanyaan for atau while. Contoh pertama seperti :

```
count = 0
while True:
    print(count)
    count += 1
    if count >= 5:
        break
```

Output pertama :

```
python while.py
0
1
2
3
4
```

26.1.12 Perintah break, continue dan else

Perintah break sama seperti dalam bahasa C, yaitu keluar dari ruang lingkup yang paling kecil dari kondisi for atau while. Perintah continue sama seperti dalam bahasa C, fungsinya untuk melanjutkan statement berikutnya dalam kondisi perulangan. Pada kondisi perulangan juga diperbolehkan untuk menggunakan kalimat perintah else, yang dijalankan saat kondisi perulangan for tidak memenuhi suatu kondisi atau jika kondisi tersebut mengalami error/false, tetapi bukan pada saat kondisi perulangan dihentikan dengan perintah break. Gambar 26.1 adalah contohnya.

```
for n in range(2, 10):
    for x in range(2, n):
        if n % x == 0:
            print(n, 'sama dengan', x, '*', n/x)
            break
        else:
            print(n, 'adalah bilangan prima')
    print(n, 'adalah bilangan prima')

    2 adalah bilangan prima
    3 adalah bilangan prima 4 sama dengan 2 * 2
    5 adalah bilangan prima 6 sama dengan 2 * 3
    7 adalah bilangan prima 8 sama dengan 2 * 4
    9 sama dengan 3 * 3
```

Gambar 26.1 Contoh Penggunaan Break

Penjelasannya adalah jika kondisi dalam perulangan for $x \in \text{range}(2, n)$ tidak terpenuhi maka alur perulangannya akan pindah ke ruang lingkup perintah else.

26.1.13 Range

26.1.13.1 Fungsi range Jika ingin melakukan perulangan dengan sejumlah yang diinginkan, fungsi built-in range sangat membantu. Fungsi tersebut menghasilkan sejumlah indeks dari nilai yang telah ditentukan. Contohnya :

```
>>> range(15)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
```

Ataupun sebagian angka yang diinginkan. Contohnya :

```
>>> range(8, 15)
[8, 9, 10, 11, 12, 13, 14]
>>> range(0, 9, 3)
[0, 3, 6]
>>> range(0, 20, 3)
[0, 3, 6, 9, 12, 15, 18]
```

Contoh tersebut menunjukan kelipatan dari suatu interval bilangan yang mempunyai sintaks range(nilai-awal, nilai-akhir, kelipatan-angka).

26.1.13.2 Contoh penggunaan range nilai_awal,nilai_akhir,pencacah Berikut ini contoh penggunaan range untuk menampilkan bilangan dari 1 hingga 100 dengan penambahan/pencacah 1 dengan menambahkan `end=` agar bilangan tampil secara horizontal tidak pindah baris ke bawah

```
for i in range(1, 101, 1) :
    print(i, end=' ')
```

Lalu, setelah perintah diatas dijalankan (run) maka akan tampil bilangan seperti berikut ini : 26.2

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27			
28	29	30	31	32	33	34	35	36	37	38	39			
40	41	42	43	44	45	46	47	48	49	50	51			
52	53	54	55	56	57	58	59	60	61	62	63			
64	65	66	67	68	69	70	71	72	73	74	75			
76	77	78	79	80	81	82	83	84	85	86	87			
88	89	90	91	92	93	94	95	96	97	98	99			
100														

Gambar 26.2 Hasil penggunaan range

26.1.13.3 Contoh penggunaan range nilai_awal,nilai_akhir Berikut ini adalah contoh penggunaan range untuk menampilkan dari bilangan tertentu sampai bilangan tertentu dan menghitung banyaknya bilangan serta menghitung jumlah seluruh bilangan yang ada.

```
awal=int(input('Set Nilai Awal = '))
akhir=int(input('Set Nilai Akhir = '))

count=0
summ=0

print('Bilangan antara %d dan %d' % (awal, aakhir))

for i in range(awal, aakhir+1) :
    print(i, end=' ')
    count=count+1
    summ=summ+i

print('Bilangan di atas ada %d bilangan' % count)
print('Jumlah semua bilangan adalah %d' % summ)
```

Setelah perintah diatas dijalankan (run) maka akan tampil bilangan seperti gambar 26.3.

26.1.13.4 Contoh penggunaan range nilai_akhir Ini adalah contoh penggunaan range untuk menampilkan bilangan dari 0 sampai 100 dengan menambahkan `end=` agar bilangan tampil secara horizontal tidak pindah baris ke bawah

```
for i in range(101):
    print(i, end=' ')
```

Disini nilai akhir menggunakan operator ; bukan jadi untuk menampilkan sampai angka 100 nilai akhir harus dibuat menjadi 101. Dan setelah perintah diatas dijalankan (run) maka akan tampil hasil seperti di gambar 26.4.

26.1.14 for loop with else

For loop bisa memiliki blok else yang opsional juga. Bagian lain dijalankan jika item urutan digunakan dalam lingkaran for loop . Pernyataan break bisa digunakan untuk menghentikan sebuah loop. Dalam kasus seperti itu, bagian yang lain diabaikan.

```

Set Nilai Awal = 10
Set Nilai Akhir = 67
Bilangan antara 10 dan 67
10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 2
5 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 5
6 57 58 59 60 61 62 63 64 65 66 67 Bilangan di
atas ada 58 bilangan
Jumlah semua bilangan adalah 2233

```

Gambar 26.3 Hasil setelah di jalankan

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42			
43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62			
63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82			
83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100					

Gambar 26.4 Nilai Akhir Setelah run

Oleh karena itu, bagian loop yang lain berjalan jika tidak terjadi pemutusan.

Berikut adalah contoh untuk menggambarkan hal ini.

```

script.py
digits = [0, 1, 5]

for i in digits:
    print(i)
else:
    print("No items left.")
Output :
0
1
5
No items left.

```

26.1.15 Penggunaan loop dengan else statement

Python mendukung untuk memiliki pernyataan lain yang terkait dengan pernyataan lingkaran

Jika else statement digunakan dengan for loop, pernyataan yang lain dijalankan saat loop telah habis mengulangi daftar.

Jika else statement digunakan dengan loop sementara, pernyataan yang lain dijalankan saat kondisinya menjadi salah.

26.1.16 Middle-test loop

Middle-test loop adalah sebuah perulangan yang akan mengeksekusi pada beberapa bagian body, kemudian akan melakukan pengujian exit berarti menguji dalam kondisi exit, lalu kemungkinan akan mengeksekusi beberapa bagian body lainnya. Disini dapat menggunakan while dan break secara bersama-sama. Terkadang kita membutuhkan looping dengan pengujian di tengah daripada pengujian di atas maupun di akhir.

26.1.17 Penjelasan Penggunaan For Loop

For loop secara tradisional digunakan saat Anda memiliki blok kode yang ingin Anda ulangi beberapa kali. Python untuk pernyataan iterates atas anggota urutan dalam urutan, mengeksekusi blok setiap waktu. Kontras untuk pernyataan dengan loop ‘while’, digunakan bila suatu kondisi perlu diperiksa setiap iterasi, atau untuk mengulang blok kode selamanya.

26.1.18 Pendukung kontrol dalam penggunaan looping python

break berguna untuk menghentikan looping ketika terjadi kondisi tertentu. continue berguna untuk melanjutkan sebuah operasi ketika pada blok statemen menghasilkan nilai yang diharapkan atau yang dicari. pass kontrol ini tidak menghasilkan apapun dan pass akan berguna untuk mengecek apakah statemen berjalan apa tidak.

PEMROGRAMAN SIG KELAS DAN FUNGSI DI PYTHON

27.1 Phyton

27.1.1 Pengertian Phyton

Python merupakan bahasa pemrograman yang berorientasi obyek dinamis, dapat digunakan untuk bermacam-macam pengembangan perangkat lunak. Python menyediakan dukungan yang kuat untuk integrasi dengan bahasa pemrograman lain dan alat-alat bantu lainnya. Python hadir dengan pustaka-pustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari. Sudah banyak programmer Python yang menyatakan bahwa mereka mendapatkan produktivitas yang lebih tinggi. Mereka juga merasakan bahwa Python meningkatkan kualitas pengembangan karena kode sumber yang mereka tulis dapat terus dipelihara. Python dapat berjalan di banyak platform / sistem operasi seperti Windows, Linux/Unix, Mac OS X, OS/2, Amiga, Palm Handhelds dan telepon genggam Nokia. Saat ini Python juga telah diporting ke dalam mesin virtual Java dan .NET. Python didistribusikan di bawah lisensi OpenSource yang disetujui OSI (OpenSource Initiatives), sehingga Python bebas digunakan, gratis digunakan, bahkan untuk produk-produk komersil.

27.1.2 Contoh Script Phyton

Seringkali pengguna harus menuliskan ekspresi yang cukup kompleks dan akan digunakan secara berulang di kemudian hari. Untuk keperluan penulisan ekspresi kompleks, kita dapat membuatnya dalam sebuah script yang dibantu dengan adanya teks editor. Penulis menggunakan vi teks editor default yang terdapat pada distro GNU/Linux. Pada contoh berikut ini, kita dapat melihat contoh script Python yang sederhana : \$ vi contoh-script-01.py #! /usr/bin/python a = 1 print Nilai a adalah : , a simpan script Anda dengan : :wq! Secara default, script Python yang Anda buat akan disimpan dengan ekstensi .py . Anda dapat melakukan eksekusi script yang telah Anda buat tersebut dengan cara : \$ python contoh-script-01.py Atau : Memberikan permission x (executable) sehingga script tersebut dapat dijalankan, dengan perintah : \$ chmod +x contoh-script-01.py \$./contoh-script-01.py

27.1.3 INTERPRETER PYTHON

Bahasa pemrograman Python dilengkapi dengan suatu fasilitas seperti shell di Linux, sehingga memungkinkan kita untuk mencoba penggunaan Python secara interaktif. Lokasi instalasi Python secara default pada distribusi Linux terletak di /usr/ bin/python.

Untuk menjalankannya, dapat kita tuliskan python pada konsole Linux pengguna. \$ python Apabila Python telah terinstalasi, maka pengguna akan mendapatkan tampilan : Python 2.4.3 (#1, May 24 2008, 13:47:28) [GCC 4.1.2 20070626 (Red Hat 4.1.2-14)] on Type help, copyright, credits or license for more information.

Tanda >>> tersebut menandakan bahwa interpreter Python telah siap menerima perintah pengguna. Kita dapat langsung mengetikkan ekspresi Python pada prompt >>> yang telah tersedia. Ekspresi Python dapat berupa ekspresi sederhana berupa assignment hingga perintah yang lebih kompleks seperti deklarasi fungsi maupun class. Contoh-contoh dasar ekspresi yang dapat kita tuliskan pada prompt Python adalah sebagai berikut :

```
>>> a = 1
>>> a
```

```

1
>>> b = 2
>>> b
2
>>> c = a + b
>>> c
3
>>> d = a - b
>>> d -1
>>> print Nilai d adalah : , d
Nilai d adalah : -1
>>> print Nilai c
adalah : , c Nilai c adalah : 3
>>> e
Traceback (most recent call last):
File <stdin>, line 1, in ?
NameError: name 'e' is not defined
>>>

```

Sedangkan untuk ekspresi Python yang lebih kompleks adalah sebagai berikut :

```

>>> a = 1
>>> if a == 1:
... print Nilai a adalah 1 ...
Nilai a adalah 1
>>> def fac(n):
... if n < 2:
... return 1
... else:
... return n * fac(n-1)
... >>> fac(8)
40320 >>>
fac(10)
3628800
>>>

```

Pada contoh diatas, tanda . . . merupakan secondary prompt, biasanya digunakan untuk kondisi atau kelanjutan dari ekspresi sebelumnya. Apabila pengguna telah selesai menggunakan interpreter Python, dapat keluar dengan menekan Ctrl D.

27.2 Class

27.2.1 Pengertian Class

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

27.2.2 Area Pokok Class

Class memiliki tiga area pokok :

1. Nama (dan stereotype)
2. Atribut
3. Method (operasi)

Atribut dan metoda dapat memiliki salah satu sifat berikut :

- Private(-), tidak dapat dipanggil dari luar class yang bersangkutan
- Protected(#), hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya
- Public(+), dapat dipanggil oleh siapa saja

27.2.3 Hubungan Antar Class

1. Asosiasi, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Panah navigability menunjukkan arah query antar class. Lambang :
2. Agregasi, yaitu hubungan yang menyatakan bagian (terdiri atas..) atau bisa disebut relasi mempunyai sebuah. Lambang :
3. Composition, yaitu sebuah kelas tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi composition. Lambang :
4. Dependency, yaitu hubungan yang menunjukkan operasi pada suatu kelas yang menggunakan class yang lain. Lambang :
5. Generalisasi / Pewarisan, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Lambang

27.3 Fungsi

Dalam menulis program, tentunya kita akan & code yang pernah kita tulis sebelumnya, pastilah akan kita gunakan kembali, dengan beberapa nilai yang berbeda. Tentu saja kita tidak mungkin menuliskan kembali kode yang ingin dipanggil ulang tersebut. Solusinya, kita dapat mengelompokkan kode-kode yang sering dipanggil ulang dalam suatu kelompok kode yang disebut fungsi. Selain itu juga Anda dapat memecah masalah besar menjadi masalah-masalah yang lebih kecil. Dalam C atau bahasa pemrograman lain, biasanya digunakan istilah function. Beberapa hal yang perlu diperhatikan terkait dengan penggunaan fungsi :

1. deklarasi fungsi, dikatakan kata kunci def, berguna untuk membuat obyek fungsi dan selanjutnya melakukan assignment obyek fungsi tersebut dengan sebuah nama.
2. Pada istilah parameter y preferences dan passing parameter by value, maka bahasa pemrograman Python melakukan passing parameter by assignment.
3. Seperti pada variabel kita tidak perlu mendeklarasikan tipenya terlebih dahulu sehingga parameter pada fungsi bisa digunakan untuk berbagai tipe obyek yang sesuai. Bentuk umum : def fungsi(argumen1, argumen2, ..., argumen n): Statement1 ... Statement n return returnvariable Keterangan : Fungsi diawali dengan kata kunci def, diikuti nama fungsi, boleh diikuti parameter formal yang ditulis dalam tanda kurung, dan deklarasi fungsi ditutup dengan tanda titik dua (:).

27.3.1 FUNGSI-FUNGSI UMUM

27.3.1.1 FUNGSI STRING Fungsi string digunakan memanipulasi string untuk berbagai macam kebutuhan. Disini akan dibahas beberapa fungsi string yang sering digunakan dalam membuat program aplikasi web.

27.3.1.2 FUNGSI DATE Digunakan untuk mengambil tanggal dan jam. Hasil dari fungsi ini adalah sebuah string yang berisi tanggal/jam sesuai dengan format yang diinginkan. Format yang dikenal dalam fungsi date ini adalah sebagai berikut:

```
a - "am" or "pm"
A - "AM" or "PM"
B - Swatch Internet time
d - day of the month, 2 digits with leading zeros; i.e. "01" to "31"
D - day of the week, textual, 3 letters; i.e. "Fri"
F - month, textual, long; i.e. "January"
g - hour, 12-hour format without leading zeros; i.e. "1" to "12"
G - hour, 24-hour format without leading zeros; i.e. "0" to "23"
h - hour, 12-hour format; i.e. "01" to "12"
H - hour, 24-hour format; i.e. "00" to "23"
i - minutes; i.e. "00" to "59"
I (capital i) - "1" if Daylight Savings Time, "0" otherwise.
j - day of the month without leading zeros; i.e. "1" to "31"
l (lowercase 'L') - day of the week, textual, long; i.e. "Friday"
L - boolean for whether it is a leap year; i.e. "0" or "1"
m - month; i.e. "01" to "12"
M - month, textual, 3 letters; i.e. "Jan"
n - month without leading zeros; i.e. "1" to "12"
s - seconds; i.e. "00" to "59"
S - English ordinal suffix, textual, 2 characters; i.e. "th", "nd"
t - number of days in the given month; i.e. "28" to "31"
T - Timezone setting of this machine; i.e. "MDT"
U - seconds since the epoch
w - day of the week, numeric, i.e. "0" (Sunday) to "6" (Saturday)
```

```

Y - year, 4 digits; i.e. "1999"
y - year, 2 digits; i.e. "99"
z - day of the year; i.e. "0" to "365"
Z - timezone offset in seconds (i.e. "-43200" to "43200")
Sintaks:
date(string format)
Contoh:
date( Y -m- d ); // menghasilkan 2001-07-28
date( l , j F Y ); // menghasilkan Saturday, 28 July 2001
date( H :i: s ); // menghasilkan 20:15:07

```

27.3.1.3 Fungsi Mail Digunakan untuk mengirimkan e-mail ke alamat e-mail tertentu. Sintaks: mail(string tujuan , string subject , string isi [, string

27.4 Contoh Class dan Fungsi pada Phyton

27.4.1 Contoh Class

Untuk membuat sebuah class ini, harus kita awali dengan sebuah kata kunci. Yaitu class yang kemudian diikuti dengan nama class nya.. Dan yang terakhir adalah tanda kurung buka dan tutup serta tanda titik dua () dan :: untuk lebih mudahnya kita bisa lihat atau simak contohnya di bawah ini..

```

class namaClass () :
    def metode 1 (self) :
        Isi metode
    def metode 2 (self) :
        Isi metode

```

Jadi, untuk memanggil sebuah class, sama saja seperti layak nya memanggil metode.. Kita cukup menyebutkan nama classnya dengan diakhiri dengan tanda kurung buka dan tutup seperti di bawah ini..

```
namaClass()
```

untuk memanggil metodenya, kita cukup menggunakan memanggil class yang kemudian diikuti dengan pemanggilan nama metode yang tersedia di dalam class tersebut dengan dipisahkan oleh tanda titik. Seperti di bawah ini..

```
namaClass().namaMetode()
```

27.4.2 Contoh Fungsi

Fungsi adalah bagian dari program yang dapat digunakan ulang. Hal ini bisa dicapai dengan memberi nama pada blok statemen, kemudian nama ini dapat dipanggil di manapun dalam program. Kita telah menggunakan beberapa fungsi built-in seperti range. Fungsi dalam Python didefinisikan menggunakan kata kunci def. Setelah def ada nama pengenal fungsi diikuti dengan parameter yang diapit oleh tanda kurung dan diakhiri dengan tanda titik dua :: Baris berikutnya berupa blok fungsi yang akan dijalankan jika fungsi dipanggil.

```

#!/usr/bin/python

def halo_dunia():
    print 'Halo Dunia!'
halo_dunia()  # memanggil fungsi halo_dunia
halo_dunia()  # fungsi halo_dunia dipanggil lagi

```

27.4.3 Contoh Script Shapefile

Shapefile kita modifikasi dengan library python pyshp[47].

```

## Not run:

#Read entire shapefile
shapefile <- read.shapefile("links")

#Write entire shapefile
write.shapefile(shapefile, "temp", T)

```

```

#Read shp, shx, or dbf file
dbf <- read.dbf("links.dbf")

#Write shp, shx, or dbf file
write.dbf(dbf, "links.dbf", T)

#Calculate header (to clean up GeoMedia shapefile exports)
shapefile <- calc.header(shapefile)

#Add the X and Y coordinates to the dbf list of the shapefile list object
shapefile <- add.xy(shapefile)

#Scale the shapefile by scale.factor
shapefile <- scaleXY(shapefile, scale.factor)

#Samples of using the convert.to.shapefile function to write out simple shapefiles
#from basic R data.frames

#Point
dd <- data.frame(Id=c(1,2),X=c(3,5),Y=c(9,6))
ddTable <- data.frame(Id=c(1,2),Name=c("Item1","Item2"))
ddShapefile <- convert.to.shapefile(dd, ddTable, "Id", 1)
write.shapefile(ddShapefile, "c:/test", arcgis=T)

#PolyLine
dd <- data.frame(Id=c(1,1,1,2,2,2),X=c(3,5,8,6,7,8),Y=c(9,8,3,6,7,4))
ddTable <- data.frame(Id=c(1,2),Name=c("Item1","Item2"))
ddShapefile <- convert.to.shapefile(dd, ddTable, "Id", 3)
write.shapefile(ddShapefile, "c:/test", arcgis=T)

#Polygon
dd <- data.frame(Id=c(1,1,1,1,2,2,2,2),X=c(3,5,8,3,6,7,8,6),Y=c(9,8,3,9,6,7,4,6))
ddTable <- data.frame(Id=c(1,2),Name=c("Item1","Item2"))
ddShapefile <- convert.to.shapefile(dd, ddTable, "Id", 5)
write.shapefile(ddShapefile, "c:/test", arcgis=T)

#Convert to list of shapes
ddAsList <- by(dd,dd$Id, function(x) x)

#Convert to data.frame
dd <- do.call(rbind, ddAsList)

#Read in shp file and convert to simple format
shpTest <- read.shp("c:/test.shp")
simpleShpFormat <- convert.to.simple(shpTest)
simpleShpFormat <- change.id(simpleShpFormat, c("a","b"))
simpleAsList <- by(simpleShpFormat, simpleShpFormat[,1], function(x) x)
backToShape <- convert.to.shapefile(simpleShpFormat,
data.frame(index=c("a","b")), "index", 5)
write.shapefile(backToShape, "c:/test", arcgis=T)

#Polyline simplification with dp algorithm
x <- c(5,3,4,1,8,9,10,11)
y <- c(6,4,2,1,1,5,2,3)
points <- list(x=x,y=y)
plot(points, type="l")
simpleLine <- dp(points, 2)
lines(simpleLine, type="l", col="blue")

## End(Not run)

```


PEMROGRAMAN SIG OPENLAYER

28.1 deskripsi openlayer

OpenLayers adalah library Javascript murni yang dipakai untuk menampilkan sebuah data peta di setiap web browser, tanpa server side dependencies. OpenLayer mengimplementasikan JavaScript API untuk membangun sebuah rich web-based geographic applications yang hampir sama seperti Google maps dan MSN Virtual Earth APIS. Open Layer adalah Software, yang dibangun oleh komunitas Open Source.

OpenLayers Syntax OpenLayers Classes Ditulis menggunakan gaya klasik yang fungsi-fungsinya ditulis dengan huruf besar pada awal syntax. contohnya : var map = new OpenLayers.Map(map, options); Syntax diatas menjelaskan sebuah map dengan semua property dari Open Layers.

OpenLayers Syntax OpenLayers Classes Ditulis menggunakan gaya klasik yang fungsi-fungsinya ditulis dengan huruf besar pada awal syntax. contoh : var map = new OpenLayers.Map(map, options); Syntax diatas menjelaskan sebuah map dengan semua property dari Open Layers. Fungsi diatas adalah fungsi prototype dari openlayer. Property yang akan diset dan diakses telah terdokumentasi di bagian API Dokumentation.

28.1.1 Base Layers

Base Layers adalah layer yang bersifat mutually Exclusive yang berarti hanya satu yang bisa diaktifkan setiap saat. Base layer yang aktif akan menentukan proyeksi sistem koordinat yang ada dan memperbesar level yang terdapat dalam peta. Base Layer atau bukan, hal ini ditentukan oleh property isBaseLayer pada layer. Kebanyakan Layer raster memiliki Properti isBaseLayer yang diset dengan true secara default. Ini bisa diganti pada layer option. Base layers selalu ditampilkan di bawah overlay layers.

28.1.2 Non Base Layers

Non base layers, terkadang disebut juga sebagai overlay, adalah merupakan Base Layers alternative. Non-base layer yang banyak dapat diaktifkan setiap saat. Layer ini tidak memegang control untuk zoom level dari peta, tapi dapat diaktifkan atau dinonaktifkan pada skala yang tepat dengan min/max scale/parameter resolusi, sehingga hanya dapat diaktifkan pada level tertentu.

28.1.3 Raster Layers

Raster layer adalah layer bayangan. Layer ini merupakan fixed projection yang tidak dapat dirubah pada client side.

28.1.4 Google

Layer yang digunakan untuk data Google Maps PAA OpenLayer. Untuk informasi API, dapat dilihat pada Google Layer Api Docs. Contoh penggunaanya dapat dilihat pada contoh penggunaan Spherical Mercator. Jika kita ingin untuk menumpang tindihkan (overlay) layer dasar Google Maps dengan data lain, kita harus berinteraksi dengan layer Google Maps pada koordinat proyeksi. Kita dapat baca selengkapnya pada proyeksi Spherical Mercator dan berbagai layer komersial yang dipergunakan di dalam dokumentasi Spherical Mercator. Class Google Layer di design hanya sebagai base layer.

Image

Sebagai informasi API selengkapnya lihat pada Image Layer API Docs.

KaMap

Sebagai informasi API selengkapnya lihat pada KaMap Layer API Docs.

KaMapCache

Sebagai informasi API selengkapnya lihat pada KaMapCache Layer API Docs.

MapGuide

Sebagai informasi API selengkapnya lihat pada MapGuide Layer API Docs.

MapServer

Ini tidak diperlukan untuk berinteraksi dengan Map Server. Secara umum, lebih diminati daripada Layer MapServer, sebab MapServer mengekspos sebagian besar fungsi CGI dalam modus WMS. Lapisan MapServer sering dapat menyebabkan peta yang tampaknya bekerja, tetapi masih bermasalah pada proyeksi atau misconfigurasi serupa lainnya. Dalam menggunakan WMS kita harus memiliki alasan yang kuat dibandingkan dengan menggunakan MapServer. Jika kita menggunakan layer MapServer, dan peta sedang diulang beberapa kali, ini menunjukkan bahwa kita belum melakukan konfigurasi dengan peda dan masih dalam proyeksi yang berbeda. openLayer tidak dapat membaca informasi dari MAPFILE, dan harus dikonfigurasi secara explicit.

28.1.5 Overlay layer

Merupakan lapisan yang memiliki sumberdata dalam format selain citra. Ini termasuk subclass dari kedua layer Marker dan layer Vector.

28.1.6 Boxes

boxes lebih baik untuk mengimplementasikan fungsi dengan vektor layer dan polygon geometric dengan mempertahankan untuk kompatibilitas mundur.

28.1.7 GML

GML layer merupakan subclass dari vektor layer yang berguna untuk membaca data dan menampilkannya. Contoh : var layer = new OpenLayers.Layer.GML(GML, gml/polygon.xml) map.addLayer(layer);

28.1.8 GeoRSS

GeoRSS akan lebih baik menggunakan layer GML dengan SelectFeature Control dibandingkan dengan Layer GeoRSS yang tidak menggunakan layer GML.

28.1.9 Markers

Markers base layer sangat mudah diaplikasikan dan penggunaan fungsi addMarkers untuk menambah marker pada layer. Makaers hanya mendukung point/ titik.

28.1.10 Text

Text layer merupakan data-data dengan format Teks dan menampilkan hasilnya sebagai marker yang dapat diklik. Layer teks adalah subclass dari Marker layer, dan tidak mendukung format garis atau polygon.

28.1.11 Vector

Vector layer adalah basis dari geometry lanjutan pada class open layer seperti (GML dan subclass WFS). Ketika membuat feature pada pengkodean javaScript, menggunakan vektor layer secara langsung adalah cara yang benar. Untuk informasi API selengkapnya lihat pada Vector Layer API Docs.

28.1.12 WFS

Untuk informasi API selengkapnya dapat dilihat pada WFS Layer API Docs. Generic Subclasses EventPane FixedZoomLevels Grid HTTPRequest SphericalMercator

28.1.12.1 Controls Controls adalah OpenLayers classes yang mempengaruhi kondisi dari peta atau menampilkan informasi tambahan kepada user. Control sebagai antar muka utama dengan peta.

28.1.12.2 panels Control panel terdiri dari kumpulan control yang dapat diakses di aplikasi Styling Panels. Panels telah dikonfigurasi dengan CSS. ItemActive dan ItemInactive telah ditambahkan pada control displayClass. Semua controls mempunyai displayClass yang dapat dioverride propertinya kedalam CSS. Panel items sudah desetting dengan menggabungkan style dari Panel dengan style control. berikut adalah contoh nya :

```

olControlNavToolbar div
display:block;
width: 28px;
height: 28px;
top: 300px;
left: 6px;
position: relative;
.olControlNavToolbar .olControlNavigationItemActive
background-image: url(img/panning-hand-on.png);
background-repeat: no-repeat;
.olControlNavToolbar .olControlNavigationItemInactive
background-image: url(img/panning-hand-off.png);
background-repeat: no-repeat;

```

28.1.12.3 Customizing an Existing Panel Beberapa panel memiliki beberapa kontrol yang digabungkan, tapi tidak semua panel memiliki kontrol. Namun, untuk membuat kontrol relatif sederhana. Misalnya, untuk membuat kontrol toolbar editing yang mempunyai fungsi menggambar garis, Anda bisa melakukannya dengan kode berikut: var layer = new OpenLayers.Layer.Vector();

```

var panelControls =
new OpenLayers.Control.Navigation(),
new OpenLayers.Control.DrawFeature(layer,
OpenLayers.Handler.Path,
displayClass: olControlDrawFeaturePath)
];
var toolbar = new OpenLayers.Control.Panel(
displayClass: olControlEditingToolbar,
defaultControl: panelControls[0]
);
toolbar.addControls(panelControls);
map.addControl(toolbar);

```

28.2 Map Controls

28.2.1 ArgParser

Membawa argumen URL, dan update peta. Dalam kontrol ArgParser pada sistem harus terdapat getCenter () supaya dapat mengembalikan nilai null sebelum mengacu pada peta untuk pertama kalinya. Berikut sintaxnya : var map = new OpenLayers.Map(map);

```

var layer = new OpenLayers.Layer();
map.addLayer(layer);
// Ensure that center is not set
if (!map.getCenter())
map.setCenter(new OpenLayers.LonLat(-71, 42), 4);
kontrol ArgParser secara default = enabled.

```


PEMROGRAMAN SIG LEAFLETJS

29.1 sejarah leafletjs

Leaflet adalah JavaScript Library terkemuka yang berfitur opensource untuk membangun peta interaktif yang Mobile friendly. Dengan ukuran hanya sekitar 38 KB, ia memiliki semua fitur pemetaan yang dibutuhkan sebagian besar pengembang. Kelebihannya karena opensource lebih mudah dikembangkan oleh peneliti selanjutnya dan mudah untuk mengadaptasi teknologi baru pada GIS. Pada penerapannya SIG memerlukan data spasial yaitu data yang merujuk kepada posisi sebuah objek dalam bentuk koordinat dalam ruang bumi. GIS adalah sistem yang dirancang untuk memperoleh, menyimpan, mengupdate, memanipulasi, menganalisis dan menampilkan semua bentuk informasi yang bersifat geografis. Dengan penggunaan leaflet, data-data spasial seperti gedung dan ruangan yang berupa format geoJson dapat disimpan di dalam server, tanpa harus terhubung ke internet hanya dengan menggunakan intranet. Untuk mengakses data-data tersebut digunakan plugin jQuery dan bootstrap untuk menampilkan peta ke halaman browser. Kelebihan menggunakan leaflet adalah leaflet menyediakan fungsionalitas untuk menambahkan penanda, pop up, garis overlay, dan bentuk menggunakan lapisan, zoom, pan, tapi ini hanya fitur ini leaflet.

29.2 pengertian leafletjs

LeafletJs merupakan library atau kumpulan fungsi berbasis javascript yang digunakan untuk menampilkan peta interaktif pada halaman web. Leaflet menyediakan Map API (Application Programming Interface) yang memudahkan web developer untuk menampilkan peta berbasis Tile pada halaman web. Pengguna peta juga dapat berinteraksi dengan menggunakan fungsi telah disediakan oleh Leaflet. Sebagaimana juga webmap API lainnya, Leaflet memiliki beberapa komponen dasar sebagai berikut : 1. Map adalah komponen induk yang memuat berbagai komponen lainnya. Bayangkan komponen Map sebagai muka peta kosong yang nantinya akan dapat diisi dengan komponen lain (seperti tilelayer, marker dan lain sebagainya). Pada komponen inilah didefinisikan ukuran peta pada halaman web (melalui fungsi CSS, width dan height), koordinat pusat(center) peta (dalam latitude dan longitude), serta level zoom awal II-13 (antara 0-20, level 20 menunjukkan perbesaran paling tinggi). Komponen dapat ditambahkan pada Map melalui method .addTo(Map) 2. Tilelayer (sering kali disebut dengan slippy map), merupakan komponen yang menyediakan latar belakang peta pada sebuah webmap/petaonline. Peta latar yang disediakan ditampilkan dalam bentuk kotak-kotak (tile) yang memiliki tampilan berbeda pada level zoom yang berbeda. 3. Marker, simbologi sederhana untuk menyatakan titik. Default simbologi untuk marker pada leaflet, meskipun leafletJS dan Map API lainnya memungkinkan developer peta untuk mengganti simbologi dengan bebas 4. Popup, adalah jendela kecil berisi informasi terkait dengan marker tertentu. Popup biasanya digunakan untuk menunjukkan informasi terkait titik tersebut, misalnya berupa rangkaian teks, gambar atau grafik 5. Event, merupakan kejadian yang dapat diamati oleh leaflet pada muka peta. Fungsi event digunakan untuk menyediakan interaktifitas dengan pengguna. 6. Control, kontrol pada leaflet merupakan pelengkap muka peta dalam leaflet merupakan tombol zoom pada peta, menu pencarian, menu pemilihan layer. 7. Vector layer, layer pada leaflet merupakan data spasial jenis vektor yang dapat ditambahkan pada komponen map Leaflet . 8. Plugin, komponen plugin memperkaya fungsi-fungsi yang sudah ada pada leaflet dengan berbagai fungsi tambahan yang dapat digunakan apabila dibutuhkan.

29.3 penggunaan leafletjs

Leaflet merupakan alternatif baru bagi para perintis peta web, seperti open layers ataupun google maps api. Ini juga dapat meringankan open source dan bertujuan untuk membentuk dan membantu mengembangkan dalam proses pembuatan peta yang indah yang compatible di seluruh pc (desktop) dan juga ponsel tanpa harus mengorbankan performa dari apa yang terjadi ketika selesai pembuatan sebelum menampilkan peta web dengan leaflet, kita diharuskan mengunduh paket leafletjs dan menyimpannya di pc yang akan digunakan. sebagai contoh, kita akan menampilkan peta web dengan adanya pilihan basemap. Untuk menambahkan fungsi pilihan basemap, download plugin leaflet providers master. lalu langkah selanjutnya adalah : 1. apabila file unduhan di atas disimpan di xampp \ htdocs \ webgis \ latihan-Leaflet 2. copy syntax di atas ke dalam text-editor yang digunakan (bisa pakai notepad++, sublime-text, atau software sejenis lainnya) 3. simpan dalam format html di xampp \ htdocs \ webgis \ Latihan-leaflet, dengan nama index.html. atau apabila file di atas disimpan di dalam folder biasa, pastikan folder plugin leaflet yang sebelumnya telah diunduh di simpan di folder yang sama juga.jalankan file index.html dengan browser yang digunakan (wajib ada koneksi internet)

29.4 fungsi leafletjs

leaflet yang interaktif dapat menampilkan peta dan mampu menghitung nilai zona tanah untuk kebutuhan jual beli pada suatu sistem informasi geografi. Selain itu penggunaan leaflet dengan di barengi menggunakan bootstrap dapat memberikan system informasi yang lebih terperinci mengenai suatu letak geografis. Dengan penggunaan leaflet, data-data spasial seperti gedung dan ruangan yang berupa format geoJson dapat disimpan di dalam server. *Bootstrap adalah sebuah alat bantu untuk membuat sebuah tampilan halaman website yang dapat mempercepat pekerjaan seorang pengembang website ataupun pendesain halaman website. Selain itu penggunaan bootstrap juga untuk mempercantik desain sistem. Peta ditampilkan menggunakan leaflet javascript yang mendukung file berformat geoJSON. Geojson merupakan format data yang berbasis JSON (Javascript Object Notation) dan dapat menampung unsur-unsur geografis. Kelebihannya adalah kompatibel dengan banyak model pemrograman pada peta, dapat digunakan pada leaflet.js dan google maps. Pada penelitian ini membatasi permasalahan, yaitu sistem yang dikembangkan merupakan sistem informasi geografis yang menampilkan ruangan atau lokasi.

29.5 permulaan leafletjs

Untuk permulaan Leaflet akan menunjukan pada anda para pengguna untuk menunjukan bagaimana mengatur atau mengsetup lingkungan dalam pengembangan Leaflet dan membuat anda untuk menggunakan basis kode yang disediakan, yang selanjutnya akan dapat melihat secara mendalam mengenai peta dan belajar tentang bagaimana untuk membangun leaflet dari ubin sumber dan yang akan menampilkannya dari penyedia yang berbeda.

29.6 Kelebihan dan Kekurangan Leaflets

- a. Kelebihan Dapat disimpan lama Sebagai reverensi Jangkauan dapat jauh Membantu media lain Isi dapat dicetak kembali Dapat sebagai bahan diskusi
- b. Kekurangan Bila cetakan kurang menarik orang enggan menyimpannya Pada umumnya orang tidak mau membaca karena hurufnya terlalu kecil Tidak bisa digunakan oleh sasaran yang buta huruf

29.7 TWEAK BASIS KODE Leaflets

Untuk penggunaan leaflet di ponsel akan menjelaskan tentang cara men-tweak basis kode agar sesuai dengan perambaan seluler yang ada dan memanfaatkan fitur lokasi pada leaflet. Pilihan lokasi probing, akurasi tinggi serta sebagainya dan juga tentang update lokasi kejadian maupun kesalahan terperinci ada untuk mengetahui perbaikan yang terbaik demi kenyamanan penggunanya.

Bagian V

APLIKASI SIG

APLIKASI SIG MAPFILE

30.1 Raster Layer dan Vektor Layer

30.1.1 Raster Layer

Dalam raster layer setiap lokasi direpresentasikan sebagai suatu posisi sel. Sel ini diorganisasikan dalam bentuk kolom dan baris sel-sel dan biasa disebut sebagai grid. Dengan kata lain, layer raster menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan struktur matriks atau piksel-piksel yang membentuk grid. Setiap piksel atau sel ini memiliki atribut tersendiri, termasuk koordinatnya yang unik. Raster layer terdiri dari piksel (atau sel), dan setiap piksel memiliki nilai yang terkait. Menyederhanakan sedikit, foto digital adalah contoh dataset raster dimana setiap nilai piksel sesuai dengan warna tertentu. Dalam GIS, nilai piksel dapat mewakili elevasi di atas permukaan laut, atau konsentrasi kimiawi, atau curah hujan dan lain-lain. Intinya adalah bahwa semua data ini diwakili sebagai kotak sel (biasanya persegi). Perbedaan antara model elevasi digital (DEM) di GIS dan foto digital adalah bahwa DEM mencakup informasi tambahan yang menjelaskan di mana tepi gambar berada di dunia nyata, bersamaan dengan seberapa besar setiap sel berada di tanah. Ini berarti GIS Anda dapat memposisikan gambar raster Anda (DEM, hillshade, peta lereng dll) dengan benar relatif terhadap satu sama lain, dan ini memungkinkan Anda membangun peta Anda.

Sementara struktur data raster sederhana, ini sangat berguna untuk berbagai aplikasi. Dalam GIS, penggunaan data raster termasuk dalam empat kategori utama:

Raster sebagai basemaps Penggunaan data raster yang umum di GIS adalah sebagai tampilan latar belakang untuk lapisan fitur lainnya. Misalnya, orthophotographs yang ditampilkan di bawah lapisan lain memberi pengguna peta keyakinan bahwa lapisan peta selaras spasial dan mewakili objek nyata, serta informasi tambahan. Tiga sumber utama raster basemaps adalah orthophotos dari fotografi udara, citra satelit, dan peta yang dipindai. Berikut adalah raster yang digunakan sebagai basemap untuk data jalan.

Raster sebagai peta permukaan

Raster sangat cocok untuk mewakili data yang terus berubah melintasi lansekap (permukaan). Mereka menyediakan metode yang efektif untuk menyimpan kontinuitas sebagai permukaan. Mereka juga menyediakan representasi permukaan yang teratur. Nilai elevasi yang diukur dari permukaan bumi adalah aplikasi peta permukaan yang paling umum, namun nilai lainnya, seperti curah hujan, suhu, konsentrasi, dan kepadatan populasi, juga dapat menentukan permukaan yang dapat dianalisis secara spasial. Raster di bawah ini menampilkan elevasi-menggunakan hijau untuk menunjukkan ketinggian yang lebih rendah dan sel-sel merah, merah muda, dan putih untuk menunjukkan ketinggian yang lebih tinggi.

30.1.1.1 cara menambahkan Raster Layer Deklarasi lapisan raster sederhana terlihat seperti ini. File DATA ditafsirkan relatif terhadap SHAPEPATH, sama seperti shapefile.

```
LAYER
  NAME "JacksonvilleNC_CIB"
  DATA "Jacksonville.tif"
  TYPE RASTER
  STATUS ON
END
```

Layer raster dapat memiliki informasi PROJECTION, METADATA, PROCESSING, MINSCALE, dan MAXSCALE. Tidak dapat memiliki informasi label, CONNECTION, CONNECTIONTYPE, atau FEATURE.

30.1.1.2 Classifying Rasters Layer Raster dapat diklasifikasikan dengan cara yang mirip dengan vektor, dengan beberapa pengecualian.

Tidak perlu menentukan CLASSITEM. Nilai piksel mentah itu sendiri ("[pixel]") dan, untuk gambar yang diberi palet, warna merah, hijau dan biru yang dikaitkan dengan nilai piksel itu ('[merah]', '[hijau]' dan '[biru]') tersedia untuk digunakan dalam klasifikasi. Bila digunakan dalam ekspresi yang dievaluasi, kata kunci pixel, merah, hijau dan biru harus dalam huruf kecil.

```
LAYER
  NAME "JacksonvilleNC_CIB"
  DATA "Jacksonville.tif"
  TYPE RASTER
  STATUS ON
  CLASSITEM "[pixel]"
  # class using simple string comparison, equivalent to ([pixel] = 0)
  CLASS
    EXPRESSION "0"
    STYLE
      COLOR 0 0 0
    END
  END
  # class using an EXPRESSION using only [pixel].
  CLASS
    EXPRESSION ([pixel] >= 64 AND [pixel] < 128)
    STYLE
      COLOR 255 0 0
    END
  END
  # class using the red/green/blue values from the palette
  CLASS
    NAME "near white"
    EXPRESSION ([red] > 200 AND [green] > 200 AND [blue] > 200)
    STYLE
      COLOR 0 255 0
    END
  END
  # Class using a regular expression to capture only pixel values ending in 1
  CLASS
    EXPRESSION /*1/
    STYLE
      COLOR 0 0 255
    END
  END
END
```

30.1.2 Vektor Layer

Data vektor adalah data yang diperoleh dalam bentuk koordinat titik yang menampilkan, menempatkan dan menyimpan data spasial dengan menggunakan titik, garis atau area (poligon). Terdapat tiga tipe bentuk data vektor (titik, garis, dan poligon) yang bisa digunakan untuk menampilkan informasi pada peta. Titik bisa digunakan sebagai lokasi sebuah tempat atau posisi tertentu dalam peta. Garis bisa digunakan untuk menunjukkan route suatu perjalanan atau menggambarkan batas suatu wilayah dan juga batas suatu kawasan hutan atau area tertentu. Poligon bisa digunakan untuk menggambarkan sebuah danau atau sebuah luasan areal yang kemudian dapat analisis luasan pada areal-areal tersebut. Di dalam model layer vektor, garis-garis atau kurva (busur atau arcs) merupakan sekumpulan titik-titik terurut yang dihubungkan (Prahasta, 2001). Poligon akan terbentuk penuh jika titik awal dan titik akhir poligon memiliki nilai koordinat yang sama dengan titik awal. Sedangkan bentuk poligon disimpan sebagai suatu kumpulan list yang saling terkait secara dinamis dengan menggunakan pointer/titik. Vektor Layer terdiri dari masing-masing titik, yang (untuk data 2D) disimpan sebagai pasangan koordinat (x, y). Poin dapat digabungkan dalam urutan tertentu untuk membuat garis, atau bergabung dalam cincin tertutup untuk membuat poligon, namun semua data vektor pada dasarnya terdiri dari daftar koordinat yang menentukan simpul, beserta peraturan untuk menentukan apakah dan bagaimana simpul tersebut digabungkan. . Perhatikan bahwa sementara data raster terdiri dari sel array yang teratur, titik-titik pada dataset vektor tidak boleh secara teratur spasi. Bidang vektor digunakan misalnya di meteorologi untuk menyimpan / menampilkan arah angin dan magnitude-nya. Sumbernya adalah dua band data raster, band pertama mewakili komponen U dari vektor, dan pita kedua komponen V. Dengan menggunakan nilai u, v pada lokasi tertentu, kita dapat menghitung rotasi dan besarnya dan menggunakan untuk menggambar anak panah dengan ukuran yang sebanding dengan magnitude dan menunjuk ke arah fenomena (angin, arus, dll.)

30.1.3 Menampilkan Map Bandung

```

MAP
name "bandung"
status on
extent 392000 9094000 483500 9166500
imagetype png
size 800 600
shapepath "../data"
imagecolor 255 200 34

WEB
imagepath "/tmp/ms_tmp/"
imageurl "/ms_tmp/" END

LAYER
name "bandung"
data "bandung_indonesia_osm_polygon"
status default
type polygon

CLASS
color 123 34 56
outlinecolor 255 2 3
END
END
END

```

30.1.4 Class

Class adalah prototype, atau blueprint, atau rancangan yang mendefinisikan variable dan method-methode pada seluruh objek tertentu. Class berfungsi untuk menampung isi dari program yang akan dijalankan, di dalamnya berisi atribut / type data dan method untuk menjalankan suatu program. Class merupakan suatu blueprint atau cetakan untuk menciptakan suatu instant dari object. class juga merupakan grup suatu object dengan kemiripan attributes/properties, behaviour dan relasi ke object lain. Contoh : Class Person, Vehicle, Tree, Fruit dan lain-lain.

30.1.5 STYLE Objects

Yang dimaksud dengan objek pada java adalah sekumpulan software yang terdiri dari variable dan method-method yang terkait. Objek juga merupakan benda nyata yang dibuat berdasarkan rancangan yang didefinisikan di dalam class

Object adalah instance dari class. Jika class secara umum mewakili sebuah object, sebuah instance adalah representasi nyata dari class itu sendiri. Contoh : Dari class Fruit kita dapat membuat object Mangga, Pisang, Apel, dan lain-lain.

30.1.6 MapServer

MapServer adalah aplikasi Open Source yang memungkinkan sebuah data peta diakses melalui web. Teknologi ini pertama kali dikembangkan oleh Universitas Minesotta Amerika Serikat. Hadirnya MapServer menjadikan pekerjaan membuat Peta Digital menjadi lebih mudah dan interaktif. Interaktif peta disini diartikan bahwa pengguna dapat dengan mudah melihat dan mengubah tampilan peta seperti zoom, rotate, dan menampilkan informasi (seperti menampilkan info jalan) dan analisis(seperti menentukan rute perjalanan) pada permukaan geografi. Diagram berikut menggambarkan bagaimana user berinteraksi dengan peta interaktif berbasis MapServer. Map Server bekerja secara berdampingan dengan aplikasi web server. Web Server menerima request peta melalui MapServer. MapServer mengenerate request terhadap peta dan mengirimkannya ke web server. Fungsi utama dari MapServer adalah melakukan pembacaan data dari banyak sumber dan menempatkannya kedalam layer-layer secara bersamaan menjadi file graphic. Salah satu layernya bisa saja berupa gambar satelit. Setiap layer saling overlay satu dengan lainnya dan ditampilkan kedalam web browser.

Untuk lebih memudahkan pemakaian MapServer, MapServer membuat sebuah aplikasi yang disebut MS4W atau MapServer for Windows. MS4W ditujukan untuk pengguna MapServer pemula yang menggunakan sistem operasi Microsoft Windows. Didalam MS4W sudah dilengkapi dengan WebServer (Apache), MapServer dan MapScript (PHP, C Sharp, Python, Java) dan dilengkapi pula dengan dukungan terhadap berbagai macam database.

Mapserver menghasilkan keluaran berupa file graphic berdasarkan masukan yang diberikan oleh user. Komponen kuncinya adalah MapServer executable yang terdiri dari CGI program, file peta, sumber data dan output gambar. Seperti pada gambar di bawah ini semua komponen bekerja bersama-sama, setelah user melakukan request/permintaan maka CGI akan mengakses file peta, menggambarkan informasi yang didapat dari sumber data dan kembali menampilkannya pada peta.

Secara sederhana MapServer menjalankan executable aplikasi CGI pada web server yang secara teknis merupakan proses stateless berbasis pada HTTP. Stateless adalah sebuah proses permintaan yang dilanjutkan dengan stop running. Aplikasi CGI

menerima permintaan dari web server, kemudian proses dilakukan dan mengembalikan respon atau data ke web server. CGI bekerja sangat sederhana tidak diperlukan sebuah pemrograman untuk dapat menggunakannya. Kita tinggal melakukan edit berdasarkan text base, konfigurasi runtime file, membuat halaman web, dan menempatkannya bekerja pada web server. MapServer CGI executable bekerja sebagai perantara antara file peta dengan program web server yang meminta peta. Permintaan di lewatkan dalam bentuk CGI parameter dari web server menuju MapServer. Gambar yang dibuat oleh MapServer selanjutnya memberikan feed back ke web server dan selanjutnya menuju user melalui web browser.

MapServer seperti sebuah mesin yang membutuhkan bahan bakar untuk dapat bekerja dan membutuhkan sistem pengiriman (delivery system) bahan bakar untuk mencapai mesin . Program MapServer perlu mengetahui layer peta yang akan digambar, bagaimana menggambarkannya, dan dimana lokasi sumber datanya. Data merupakan bahan bakarnya dan file peta atau .map.file merupakan sistem pengirimannya (delivery system). File Peta adalah text konfigurasi yang terdiri dari list setting yang digunakan untuk menggambar dan berinteraksi dengan peta. Informasi yang termuat di dalamnya adalah layer data apa yang akan digambar, dimana focus geografis petanya, sistem proyeksi yang digunakan, format apa yang akan digunakan untuk menampilkan gambar, dan cara menentukan legenda dan skala pada peta.

Contoh script dasar pemetaan dengan satu layer.

```
MAP
SIZE 600 300
EXTENT -180 -90 180 90
LAYER
NAME countries
TYPE POLYGON
STATUS DEFAULT
DATA countries.shp
CLASS
OUTLINECOLOR 100 100 100
END
END
END
```

Ketika request atau permintaan datang dari aplikasi MapServer maka request tersebut mesti menyebutkan sepesifikasi file peta yang diinginkan. Kemudian MapServer membuat petanya berdasarkan pada setting pada file peta yang diberikan tadi.

30.1.7 About STYLE Objects

Sama seperti Anda menggunakan gaya paragraf dan karakter untuk memformat teks dengan cepat, Anda dapat menggunakan gaya objek untuk membuat grafik dan bingkai dengan cepat. Gaya objek termasuk pengaturan untuk stroke, warna, transparansi, bayangan drop, gaya paragraf, bungkus teks, dan banyak lagi. Anda dapat menetapkan efek transparansi yang berbeda untuk objek, isi, goresan, dan teks.

Anda dapat menerapkan gaya objek ke objek, kelompok, dan bingkai (termasuk bingkai teks). Sebuah gaya dapat menghapus dan mengganti semua pengaturan objek atau hanya dapat mengganti pengaturan tertentu, sehingga pengaturan lainnya tidak berubah. Anda mengontrol pengaturan mana yang mempengaruhi gaya dengan memasukkan atau mengecualikan kategori pengaturan dalam definisi.

Anda juga bisa menerapkan gaya objek ke bingkai grid. Secara default, setiap grid bingkai yang Anda buat menggunakan gaya objek [Kotak Dasar]. Anda bisa mengedit gaya [Grid Dasar] atau Anda bisa menerapkan gaya objek lain ke grid. Saat Anda membuat atau mengedit gaya objek untuk bingkai kotak, gunakan bagian Story Options untuk menentukan arah penulisan, tipe frame, dan kotak bernama.

Saat membuat gaya, Anda mungkin menemukan beberapa gaya memiliki beberapa karakteristik yang sama. Alih-alih menetapkan karakteristik tersebut setiap kali Anda menentukan gaya berikutnya, Anda dapat mendasarkan satu gaya objek ke objek lainnya. Bila Anda mengubah gaya dasar, atribut bersama apa pun yang muncul dalam perubahan gaya "orang tua" pada gaya "anak" juga. ——

APLIKASI SIG DESKTOP QGIS

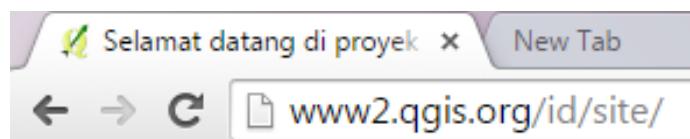
31.1 QGIS

QGIS merupakan sebuah perangkat SIG Open Source yang user friendly dengan lisensi di General Public License. QGIS merupakan suatu proyek yang tidak resmi dari Open Source OSGeo. QGIS dapat dijalankan pada OS Linux, Unix, Windows, Android dan Mac OSX. serta mendukung banyak format dan fungsionalitas data dari vektor, basisdata dan raster. Quantum GIS mendukung GPS tools untuk upload atau download data ke unit GPS. Pengguna juga bisa mengkonversi format GPS ke format GPX atau melakukan kegiatan import atau export data format GPX. Jika pengguna mempunyai sebuah web server yang telah terpasang fitur UMN MapServer, pengguna dapat melakukan publikasi map di internet untuk berbagi dengan pengguna lainnya.

SIG adalah sebuah perangkat sistem yang dirancang untuk memungkinkan orang-orang bekerja dengan data yang berkaitan. SIG memungkinkan pembuatan, penyimpanan, manipulasi, dan analisis data geografis. SIG merupakan satu konsep yang dapat melibatkan hardware dan software yang rumit. Tetapi, untuk memenuhi tujuan kebanyakan orang, yang dibutuhkan adalah sebuah perangkat lunak SIG yang sederhana, dan pada unit ini kita akan mempelajari bagaimana menggunakan aplikasi open-source yang unggul, QGIS.

31.1.1 Getting QGIS

1. Buka browser internet Anda dan ketikkan pada bagian atas jendela browser Anda dengan tulisan qgis.org seperti gambar 31.1. Kemudian tekan Enter.



Gambar 31.1 Alamat Situs Qgis

Situs resmi QGIS akan terlihat seperti gambar 31.2.

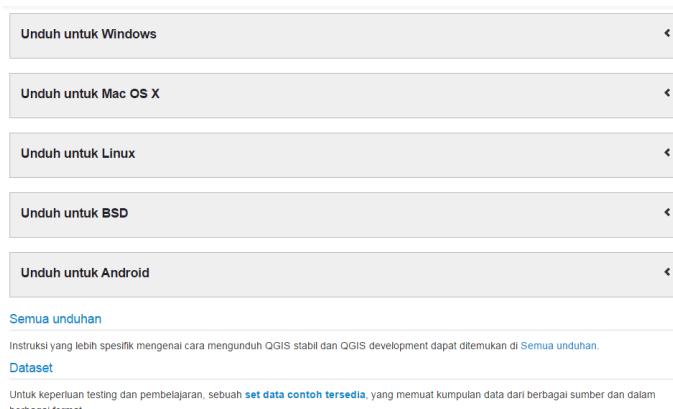
2. Klik Unduh Sekarang
3. Jika Anda menggunakan Windows, klik pada QGIS Standalone Installer Version 2.8 (32 bit). Nomor versi komputer Anda mungkin berbeda seperti diperlihatkan pada gambar 31.3.
4. Jika Anda tidak menggunakan Windows, pilih Sistem Operasi Anda dari menu yang tersedia pada gambar 31.4. Ikuti intruksi instalasi.
5. Ketika file selesai didownload, jalankan dan ikuti perintah untuk menginstal QGIS.



Gambar 31.2 Tampilan Situs Qgis



Gambar 31.3 Pilihan Paket Aplikasi



Gambar 31.4 Pilihan Sistem Operasi

31.1.2 Installing QGIS

1. Buka folder dimana anda menyimpan file instalasi QGIS dan klik installernya seperti terlihat pada gambar 31.5.
2. Jalankan file instalasi tersebut. Jika Anda menginstal QGIS versi 2.x, akan terlihat seperti gambar 31.6.
3. Klik Next
4. Klik I Agree untuk setuju dengan syarat dan ketentuan yang berlaku pada saat keluar bagian seperti gambar 31.7.
5. Pada jendela berikutnya Anda akan ditanyakan dimana Anda akan menginstall QGIS. Pada kasus umum, pengaturan bawaan yang ada sudah dapat digunakan. Klik Next pada gambar 31.8.
6. Pada jendela berikutnya, klik Install tanpa mencentang apapun yang ada di dalam kotak pada gambar 31.9.
7. QGIS akan memulai untuk menginstall. Ini mungkin akan membutuhkan beberapa waktu untuk selesai. Terlihat proses pada gambar 31.10.
8. Klik Finish (gambar 31.11) untuk melengkapi instalasi. Kemudian komputer Anda akan me-reboot secara otomatis.
9. Buka QGIS dari Start Menu, gambar 31.12 adalah tampilan QGIS.

31.1.3 Pemberian Label

Pemberian label merupakan cara untuk mengkomunikasikan informasi seperti nama dari suatu tempat. Contohnya kita ingin menunjukkan tiap-tiap jenis vegetasi tersebut berada dimana. Dengan menggunakan sebuah label, akan tampak seperti gambar 31.13. Bisa kita lihat, hal tersebut tampak tidak ideal, jadi kita membutuhkan solusi yang lain.



Gambar 31.5 File Installer Qgis

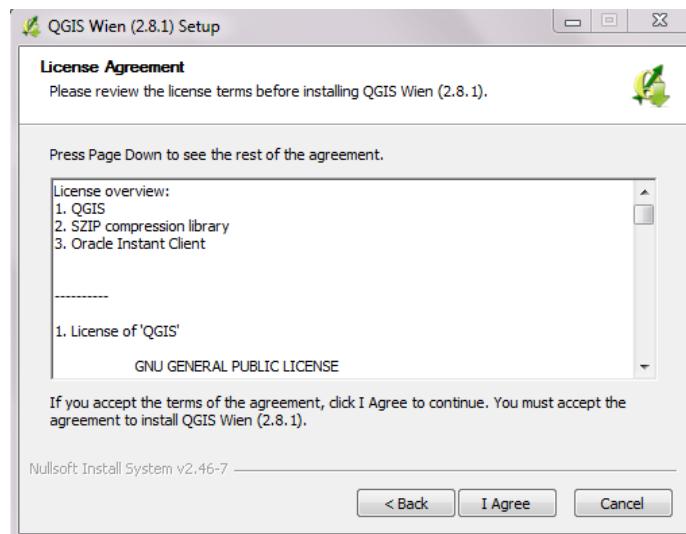


Gambar 31.6 Awal Mulai Instalasi

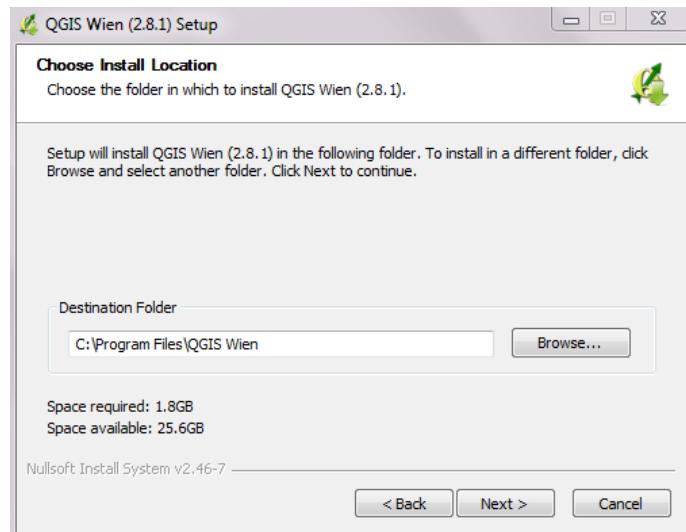
31.1.4 Toolbar

Pada bagian atas dari tampilan QGIS terdapat banyak sekali tool, dimana masing-masing tool tersebut masuk ke dalam beberapa kategori toolbar. Misalnya file mengizinkan pengguna untuk menyimpan dan memulai proyek baru(gambar 31.14).

Perhatikan gambar 31.15. Dengan menggarakan mouse ke ikon, nama pada tool akan muncul untuk membantu mengidentifikasi setiap tool yang ada. Tool dikelompokkan sesuai fungsi pada toolbars. dengan meng-klik menggunakan mouse ,dapat memindahkan toolbar ke tempat yang lebih sesuai.



Gambar 31.7 Form Persetujuan Instalasi



Gambar 31.8 Lokasi Instalasi

31.1.5 Status Bar

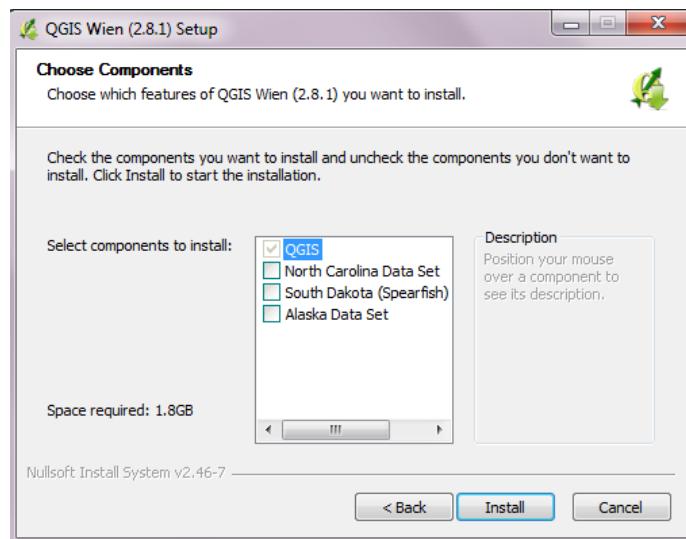
Status bar akan menampilkan informasi mengenai peta Anda. pengguna juga diperbolehkan untuk mengatur skala pada peta dan melihat koordinat peta.

Koordinat peta ini sama dengan tipe koordinat yang telah Anda pelajari ketika Anda belajar mengenai GPS. Mungkin hal ini masih belum terlalu jelas untuk Anda, tapi seiring dengan meningkatnya kemampuan Anda di SIG, hal ini nantinya akan terlihat masuk akal.

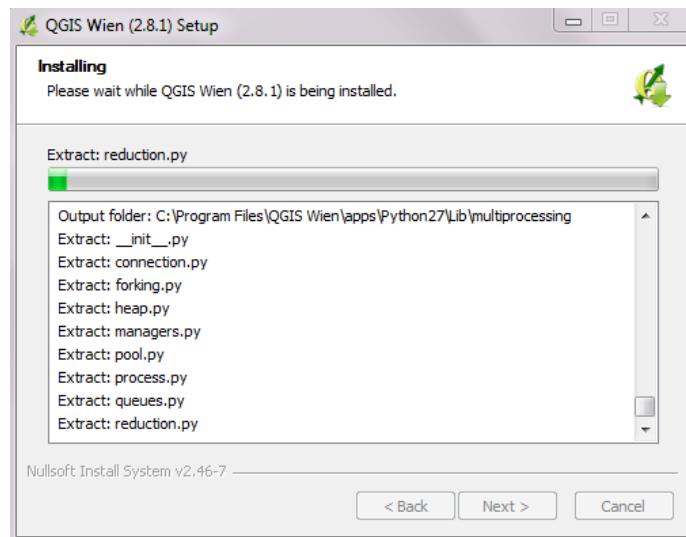
31.1.6 Menu Bar

Menu bar (gambar 31.17) memberikan akses ke berbagai fitur QGIS menggunakan hierarki menu standar. Menu utama dan beberapa pilihan menu yang tercantum dengan ikon yang sesuai seperti yang ditampilkan pada toolbar di bawah ini. serta cara pintas keyboard. Cara pintas keyboard juga dapat dikonfigurasi secara manual (cara pintas yang disajikan dalam bagian ini adalah default), menggunakan [Configure Shortcuts] alat di bawah Setting. Beberapa pilihan menu hanya muncul jika plugin yang sesuai dimuat. Untuk informasi lebih lanjut tentang alat dan toolbar, lihat Bagian Toolbar.

1. Project
2. Edit
3. View
4. Setting
5. Plugin
6. Vector



Gambar 31.9 Pilihan Paket Instalasi



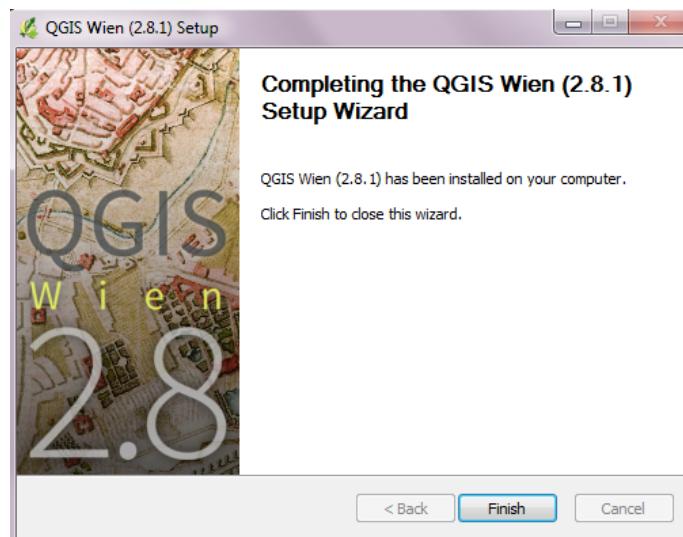
Gambar 31.10 Proses Instalasi

7. Raster
8. Database
9. Pengolahan
10. Bantuan

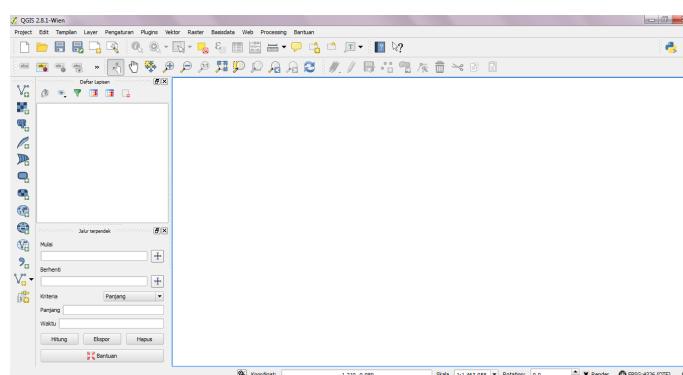
31.1.7 Atribut

Data gis mempunyai fitur dan attribut. Atribut adalah data terstruktur mengenai setiap fitur. Cara ini menunjukkan bagaimana cara melakukan query standard pada attribut di QGIS. Berikut langkahnya :

1. Buka proyek yang telah dikerjakan sebelumnya.
2. Pilih salah satu jalan pada panel daftar Layer.
3. Klik kanan, dan klik tombol Open Attribute Table : Akann terlihat tabel dengan data yang lebih banyak tentang layer jalan. Data ekstra tersebut disebut data atribut. Garis-garis yang Anda dapat lihat pada peta Anda menggambarkan kemana garis tersebut menuju ini merupakan data spatial.
4. Lihatlah pada tabel atribut. Setiap baris tabel menghubungkan satu fitur pada layer jalan. Setiap kolom mengandung satu atribut.
5. Tutup tabel atribut



Gambar 31.11 Selesai Instalasi



Gambar 31.12 Tampilan Awal Qgis



Gambar 31.13 Label yang menumpuk

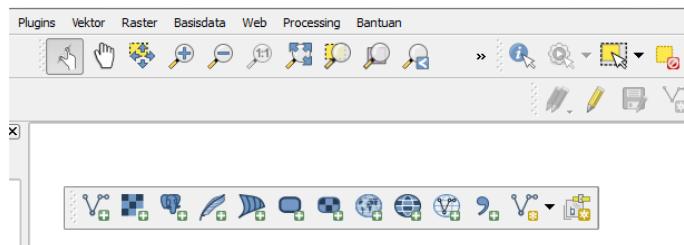


Gambar 31.14 gambar toolbar yang ada pada QGis

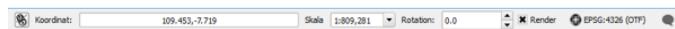
31.1.8 Label Tool

Untuk menambahkan label pada QGIS ada beberapa cara lebih baik dibandingkan yang lain. Berikut langkah-langkahnya :

1. Pergi ke menu item View lalu ke Toolbars.
2. Pastikan item Label telah memiliki tanda centang disebelahnya. Jika belum, klik pada item Label dan fitur tersebut akan diaktifkan. Terlihat seperti gambar 31.27.



Gambar 31.15 gambar toolbar yang ada pada QGis



Gambar 31.16 gambar status bar yang ada pada QGis

Menu Pilihan	Shortcut	Toolbar
New	Ctrl + N	Project
Open	Ctrl + O	Project
New from template ,		Project
Open > Recent		Project
Save	Ctrl + S	Project
Save As	Ctrl + Shift + S	Project
Save as Image		Project
New Print Composer	Ctrl + P	Project
Composer manager ...		Project
Print Composer ,		
Exit QG	Ctrl + Q	

Gambar 31.17 Gambar project pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
Undo	Ctrl+Z	Advanced Digitizing
Redo	Ctrl+Shift+Z	Advanced Digitizing
Cut Features	Ctrl+X	Digitizing
Copy Features	Ctrl+C	Digitizing
Paste Features	Ctrl+V	Digitizing
Add Feature	Ctrl+. .	Digitizing
Move Feature(s)		Digitizing
Delete Selected		Digitizing
Rotate Feature(s)		Advanced Digitizing
Simplify Feature		Advanced Digitizing
Add Ring		Advanced Digitizing
Add Part		Advanced Digitizing
Delete Ring		Advanced Digitizing
Delete Part		Advanced Digitizing
Reshape Features		Advanced Digitizing
Offset Curves		Advanced Digitizing
Reshape Features		Advanced Digitizing
Offset Curves		Advanced Digitizing
Split Features		Advanced Digitizing
Merge Selected Features		Advanced Digitizing
Merge Attr. of Selected Features		Advanced Digitizing
Node Tool		Digitizing
Rotate Point Symbols		Advanced Digitizing

Gambar 31.18 Gambar edit pada Menu Bar

3. Klik layer POI_Bandung_OSM yang terdapat pada panel Daftar Layer, sehingga layer tersebut tersetor
4. Klik tombol Layer Labelling Options
5. Setelah klik tombol diatas maka akan muncul halaman pengaturan Layer Labelling. Centang kotak yang ada tulisan Layer this Label With
6. Pilih Field Name untuk pemberian label

Menu Pilihan	Shortcut	Toolbar
Pan Map		Map Navigation
Pan Map to Selection		Map Navigation
Zoom In	Ctrl++	Map Navigation
Zoom Out	Ctrl+-	Map Navigation
Select >		Attributes
 Identify Features	Ctrl+Shift+I	Attributes
Measure >		Attributes
 Zoom Full	Ctrl+Shift+F	Map Navigation
 Zoom To Layer		Map Navigation
 Zoom To Selection	Ctrl+J	Map Navigation
 Zoom Last		Map Navigation
 Zoom Next		Map Navigation
 Zoom Actual Size		Map Navigation
Decorations >		
 Map Tips		Attributes
 New Bookmark	Ctrl+B	Attributes
 Show Bookmarks	Ctrl+Shift+B	Attributes
 Refresh	Ctrl+R	Map Navigation

Gambar 31.19 Gambar view pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
Panels >		
Toolbars >		
Toggle Full Screen Mode	Ctrl+F	
 Project Properties ...	Ctrl+Shift+P	
 Custom CRS ...		
Style Manager...		
 Configure shortcuts ...		
 Customization ...		
 Options ...		
Snapping Options ...		

Gambar 31.20 Gambar setting pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
 Manage and Install Plugins		
Python Console		
GRASS >		GRASS

Gambar 31.21 Gambar plugin pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
Coordinate Capture >		Vector
Dxf2Shp >		Vector
GPS >		Vector
Open Street Map >		
Road Graph >		
Spatial Query >		Vector

Gambar 31.22 Gambar vektor pada Menu Bar

7. Klik OK maka peta akan muncul dengan label

31.1.9 Fitur Dasar Quantum GIS Dalam Pengelolaan Data Vektor dan Raster

Sebagai perangkat lunak Sistem Informasi Geografik, Quantum GIS memiliki kapabilitas untuk menampilkan, mengolah dan menyajikan data. Contohnya sebagai berikut:

Menu Pilihan	Shortcut	Toolbar
Raster calculator		
Georeferencer		Raster
Heatmap		Raster
Interpolation		Raster
Zonal Statistics		Raster

Gambar 31.23 Gambar raster pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
eVis		Database
Split		Database

Gambar 31.24 Gambar database pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
Toolbox		Toolbox
Graphical Modeler		
History and Logs		
Options and configuration		
Results viewer		
Commander	Ctrl+Alt+M	

Gambar 31.25 Gambar pengolahan pada Menu Bar

Menu Pilihan	Shortcut	Toolbar
Help Contents	F1	Help
What's This?	Shift+F1	Help
API Documentation		
Need support?		
qg Home Page	Ctrl+H	
Check qg Version		
About		
2010 qg Sponsors		

Gambar 31.26 Gambar bantuan pada Menu Bar



Gambar 31.27 Toolbar Label tampak seperti ini

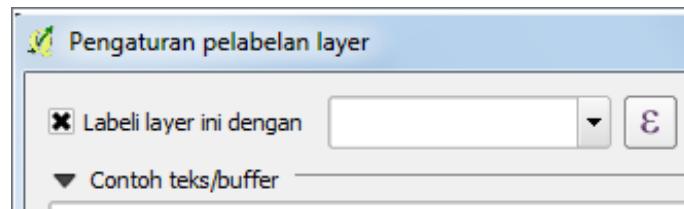
1. Membaca dan mengedit data dalam format vektor dan raster, termasuk data atribut Quantum GIS dapat membaca dan menolah data dalam banyak format, baik dalam bentuk raster maupun vektor.
2. Konversi sistem koordinat dan proyeksi peta Konversi sistem koordinat dan proyeksi peta dapat dilakukan dengan mudah di Quantum GIS, dengan memilih opsi CRS ketika menyimpan data, seperti gambar berikut:
3. Navigasi peta Pada Quantum GIS, navigasi peta bisa dilakukan melalui toolbar khusus, dengan fungsi navigasi yang bisa digunakan antara lain: Perbesar (zoom in) dan Perkecil (zoom out), Penggeseran (pan).
4. Setting tampilan peta Tampilan peta dapat diatur melalui menu Layer & Properties. Hal-hal yang dapat diatur antara lain: warna dan pola arsiran, warna dan ketebalan garis, bentuk dan ukuran simbol, dan sebagainya. Tampilan setting layer seperti berikut:

31.1.10 Menambahkan Data Vektor

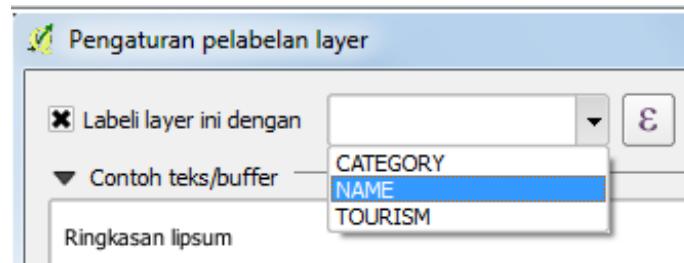
1. Buka proyek QGIS yang baru. Peta dan daftar layer yang akan tampak masih kosong.
2. Terdapat dua cara untuk menambahkan sebuah layer vektor baru pada proyek.
3. Klik pada tombol Navigasi ke direktori qgis/sarijadi/ dan pilih Jalan_Sarijadi.OSM, POI_Sarijadi dan Kecamatan_Sukasari. Anda dapat memilih lebih dari satu file dengan menahan tombol CTRL pada keyboard dan klik tiap file. Klik Open lalu Open lagi. Peta Anda sekarang akan terlihat seperti gambar 31.33.



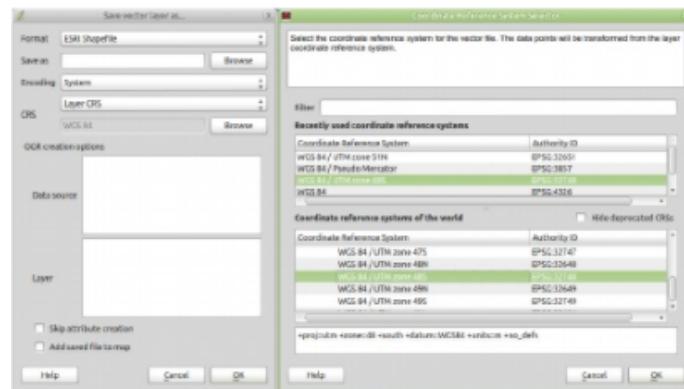
Gambar 31.28 tombol Layer Labelling Options



Gambar 31.29 tombol Layer this Label With



Gambar 31.30 Field Name

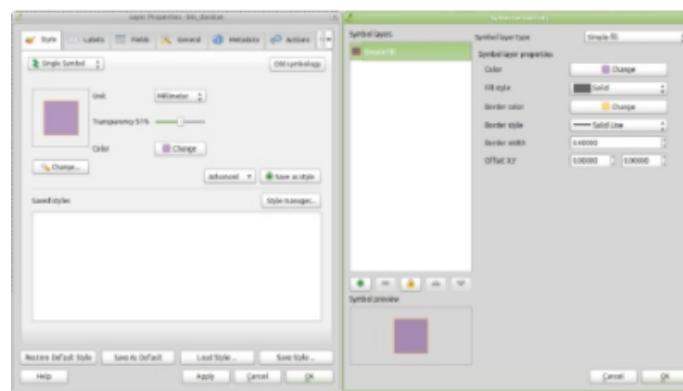


Gambar 31.31 Gambar konversi sistem koordinat dan proyeksi peta pada QGIS

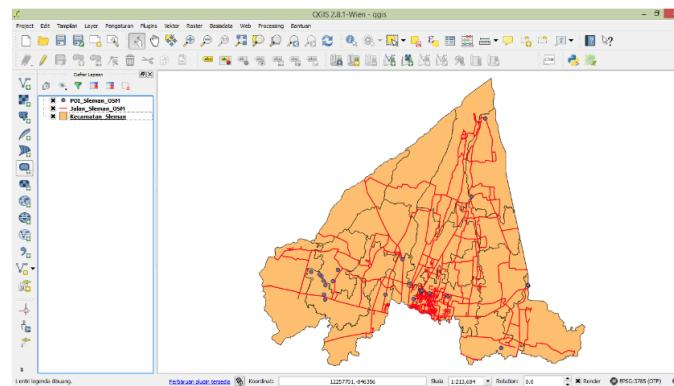
31.1.11 Mode Algoritma

Pada mode algoritma ini menggunakan algoritma yang berbeda untuk memilah sebuah data ke kelas yang berbeda.

- Interval sama: seperti namanya, metode ini akan menghasilkan kelas dengan ukuran yang sama.
- Kuantil : Merupakan Metode untuk memutuskan kelas di mana jumlah nilai pada setiap kelas adalah sama.



Gambar 31.32 Gambar setting tampilan peta pada QGIS



Gambar 31.33 Gambar vektor pada qgis

31.1.12 Klasifikasi Data Vektor

Untuk menampilkan simbol-simbol yang berbeda pada sebuah objek. Contoh: Peta Kuliner, isinya objek yang telah diklasifikasikan berdasar jenis tempat kuliner.

31.1.13 MS4W

MapServer merupakan pengembangan sebuah perangkat lunak open source yang dapat digunakan untuk mengembangkan aplikasi internet-based yang melibatkan tampilan data spasial peta digital[58].

Tabel 31.1 Paket dasar M4SW

1	Webserver Apache
2	PHP
3	MapServer CGI
4	PHP/Mapscript
5	Program utiliti (pustaka) GDAL dan OGR
6	Program utiliti MapServer (shp2img, legend, scalebar, sortshp, sym2img, shptree, dan tile4ms)
7	Ekstensi OGR/PHP
8	OWTChart

PYTHON SHAPEFILE

32.1 Pyshp

Shapefile adalah file yang berisi domain map dataset dan dapat dibuka dengan aplikasi-aplikasi tertentu yang memiliki fitur GIS di dalamnya. Python adalah bahasa pemrograman yang dapat digunakan untuk membuka shapefile tersebut. Pyshp adalah library python yang berfungsi agar bisa membaca shapefile, salah satunya format shp. Pip adalah Package Management System yang berfungsi untuk meng-install dan me-manage paket software yang ada di dalam Python. Metode yang saya lakukan diatas adalah menghitung jumlah shape yang tersedia di dalam file shapefile tersebut.

32.1.1 ESRI Shapefile

Shapefiles awalnya dikembangkan oleh Environmental Systems Research Institute (ESRI, 1998)32.1 dan merupakan format data



Gambar 32.1 ESRI.

geospasial untuk perangkat lunak sistem informasi geografis , seperti ESRI ArcGIS dan Quantum GIS. Format shapefile spasial menggambarkan geometri sebagai fitur titik, polyline, atau poligon

Shapefile adalah pengelompokan beberapa file untuk mewakili aspek geodata yang berbeda:

1. .shp: bentuk format; geometri fitur itu sendiri.

2. .shx: bentuk format indeks; indeks posisi geometri fitur untuk memungkinkan pencarian cepat.

3. .dbf: format atribut; atribut columnar untuk setiap bentuk, dalam format dBase IV.

Atribut fitur dapat diperiksa di perangkat lunak SIG, atau dengan membuka berkas .dbf di Microsoft Excel, misalnya. Ada juga beberapa file pilihan dalam format shapefile. Yang paling penting adalah file .prj yang menggambarkan sistem koordinat dan informasi proyeksi yang digunakan. Shapefiles yang dihasilkan untuk model bahanaya NBCC2015 ditentukan dengan proyeksi WGS84

32.1.2 Instalasi pyshp

Install Python terlebih dahulu Install PIP di python Buka cmd lalu ketik: Python m pip install pyshp ,lalu enter Kemudian upgrade pip dengan mengetikkan di cmd: Python m pip install upgrade pip ,lalu enter Lalu masuk ke python

32.1.3 Dasar Penggunaan

berikut ini adalah dasar penggunaan module pyshp di python

Membuat file pada pyshp

```
import shapefile
a = shapefile.Writer(shapeType=1)
a.field('nama','C','40')
a.field('alamat','C','255')
a.save('namafile.shp')
```

Mengedit atau Menambah record

```
import shapefile
a = shapefile.Editor(shapefile='namafile.shp')
a.record('politeknis pos','jl.sarijadi')
a.point(-6.8731953,107,5737873,0,0)
a.save('namafile')
```

Menghapus record

```
import shapefile
a = shapefile.Editor('namafile.shp')
a.shape(0)
#masukan data ke berapa
#karena array jadi dimulai dari 0
a.delete(0)
a.save('namafile')
```

Membaca record

```
import shapefile
a = shapefile.Reader('namafile.shp')
a.records() #menampilkan semua record
a.record(0) #spesifik record
```

DAFTAR PUSTAKA

1. R. Husein, "Konsep dasar sistem informasi geografis (geographics information system)," 2006.
2. E. Prahasta, "Sistem informasi geografis konsep-konsep dasar (perspektif geodesi & informatika," 2009.
3. A. K. Hua, "Sistem informasi geografi (gis): Pengenalan kepada perspektif komputer (geographic information system (gis): Introduction to the computer perspective)," *Geografia-Malaysian Journal of Society and Space*, vol. 11, no. 1, 2017.
4. L. Smart, *Maps that made history: the influential, the eccentric and the sublime*. Dundurn, 2005.
5. G. K. Plochmann, "Dictionary of scientific biography," *Journal of the History of Philosophy*, vol. 21, no. 1, pp. 127–128, 1983.
6. K. Lasky, *The librarian who measured the earth*. Little, Brown Books for Young Readers, 2008.
7. P. Smith, "A dictionary of greek and roman biography and mythology, edited by william smith," *Ann Arbor: University of Michigan Library*, 2005.
8. S. Grimby, *Encyclopedia of the ancient world*. Routledge, 2013.
9. D. W. Roller *et al.*, *Eratosthenes' Geography*. Princeton University Press, 2010.
10. C. Eckerman, "Eratosthenes roller eratosthenes' geography. fragments collected and translated, with commentary and additional material. pp. xvi+ 304, maps. princeton and oxford: Princeton university press, 2010. cased,£ 34.95, us 49.50. isbn : 978 – 0 – 691 – 14267 – 8," 2011.
11. N. Nicastro, *Circumference: Eratosthenes and the ancient quest to measure the globe*. Macmillan, 2008.
12. T. G. Chondros, "Archimedes life works and machines," *Mechanism and Machine Theory*, vol. 45, no. 11, pp. 1766–1775, 2010.
13. A. B. Gulick, "Athenaeus the deipnosophists, with an english translation, the loeb classical library."
14. T. Campbell, "A descriptive census of willem blaeu's sixty-eight centimetre globes 1," *Imago Mundi*, vol. 28, no. 1, pp. 21–50, 1976.
15. H. Richter, "Willem jansz. blaeu with tycho brahe on hven, and his map of the island: some new facts," 1939.
16. G. W. Wetherill, "Formation of the earth," *Annual Review of Earth and Planetary Sciences*, vol. 18, no. 1, pp. 205–256, 1990.
17. M. Zuhdi, "Sistem koordinat geografik," 2012.
18. A. Widiyati, "Meningkatkan hasil belajar geografi pada materi sejarah pembentukan bumi dengan kartu indeks dan media film," *Jurnal Geografi Gea*, vol. 11, no. 2, 2011.
19. P.-N. W. from Byrd Polar Research Center, D. of Geology, and C. O. . U. Mineralogy, The Ohio State University, "The cenozoic history of antarctica and its global impact," in *Journal of Antarctic Science* 2, vol. 1, 1990, pp. 3–2.
20. P. T. Surya, "Sejarah bumi."
21. A. Hallam, "Alfred wegener and the hypothesis of continental drift," *Scientific American*, vol. 232, no. 2, pp. 88–97, 1975.
22. F. Mufidah and M. Jamhuri, "Solusi numerik persamaan poisson menggunakan jaringan fungsi radial basis pada koordinat polar," *CAUCHY-JURNAL MATEMATIKA MURNI DAN APLIKASI*, vol. 3, no. 4, pp. 225–232, 2015.
23. T. S. Shong, K. S. Chan, V. Sengodan, and N. Jailani, "Analisis jenis kesilapan dalam pembelajaran geometri koordinat (error analysis of students in the learning of coordinate geometry)," *Jurnal Pendidikan Matematik*, vol. 1, no. 1, pp. 19–30, 2013.
24. H. B. Rachman, "Planet bumi (1)."
25. H. Setyanto and F. F. R. S. Hamdani, "Kriteria 29: Cara pandang baru dalam penyusunan kalender hijriyah," *Al-Ahkam*, vol. 25, no. 2, pp. 205–220, 2015.
26. H. S. Wicaksono, "Ta: Perancangan dan pembuatan kalender digital berdasarkan 4 sistem penanggalan berbasis microcontroller," Ph.D. dissertation, Stikom Surabaya, 2008.

27. F. Umami, T. A. Kusmayadi, and S. Suyono, "Eksperimentasi pembelajaran matematika dengan model pembelajaran kooperatif tipe jigsaw dengan pendekatan kontekstual berbasis lesson study pada materi bangun ruang sisi lengkung ditinjau dari gaya belajar siswa kelas ix mts negeri kabupaten madiun," *Jurnal Pembelajaran Matematika*, vol. 1, no. 4, 2013.
28. A. Suharjana, "Mengenal bangun ruang dan sifat-sifatnya di sekolah dasar," 2008.
29. F. Nurfarikhin, "Hubungan kemampuan pemahaman konsep dan kemampuan penalaran dengan kemampuan pemecahan masalah pada materi bangun ruang sisi lengkung peserta didik kelas ix mts nu 24 darul ulum pidodo kulon patebon kendal," Ph.D. dissertation, IAIN Walisongo, 2010.
30. M. Adieb, "Studi komparasi penentuan arah kiblat istiwaaini karya slamet hambali dengan theodolite," Ph.D. dissertation, IAIN Walisongo, 2014.
31. A. Sunyoto, "Api location (jsr 179): standar penentuan posisi untuk telepon seluler berkemampuan java," *Jurnal Dasi*, vol. 10, no. 1, 2009.
32. M. Khusurur and J. Arifin, "Mengenal equation of time, mean time, universal time/greenwich mean time dan local mean time untuk kepentingan ibadah," *YUDISIA: Jurnal Pemikiran Hukum dan Hukum Islam*, vol. 5, no. 1, 2016.
33. A. H. Yanti, V. Dhewiyanty, and T. R. Setyawati, "Prevalensi dan intensitas larva infektif nematoda gastrointestinal strongylida dan rhabditida pada kultur feses kambing (*capra sp.*) di tempat pemotongan hewan kambing pontianak," *Protobiont*, vol. 4, no. 1, 2015.
34. M. Arkanudin, "Teknik penentuan arah kiblat," *Jakarta: Lembaga Pengkajian dan Pengembangan Ilmu Falak dan Rukyatul Hilal Indonesia*, 2008.
35. D. H. Maling, *Coordinate systems and map projections*. Elsevier, 2013.
36. A. Puntodewo, S. Dewi, and J. Tarigan, *Sistem informasi geografis untuk pengelolaan sumberdaya alam*. CIFOR, 2003.
37. E. Irwansyah, *Sistem Informasi Geografis: prinsip dasar dan pengembangan aplikasi*. DigiBook Yogyakarta, 2013.
38. *Rekayasa Perangkat Lunak : software Engineering*. ANDI Yogyakarta, 2010.
39. E. Budiyanto, *Sistem Informasi Geografis Menggunakan Arcview GIS*. ANDI Yogyakarta, 2002.
40. M. Lupp, "Open geospatial consortium," in *Encyclopedia of GIS*. Springer, 2008, pp. 815–815.
41. R. García, J. P. de Castro, E. Verdú, M. J. Verdú, and L. M. Regueras, "Web map tile services for spatial data infrastructures: Management and optimization," in *Cartography-A Tool for Spatial Analysis*. InTech, 2012.
42. J. Kadlec, A. W. Miller, and D. P. Ames, "Extracting snow cover time series data from open access web mapping tile services," *JAWRA Journal of the American Water Resources Association*, vol. 52, no. 4, pp. 916–932, 2016.
43. E. K. Dewi, "Optimalisasi web gis dengan metode tiling."
44. M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, 2008.
45. H. SURYA TAMIN, "Sistem informasi geografis lokasi atm di kota medan berbasis web," 2013.
46. M. D. Kennedy, *Introducing geographic information systems with ARCGIS: a workbook approach to learning GIS*. John Wiley & Sons, 2013.
47. B. Stabler and M. B. Stabler, "The shapefiles package," 2006.
48. Y. Widiatmoko and F. Wahid, "Aplikasi web data spasial kependudukan indonesia dengan scalable vector graphics (svg)," *Media Informatika*, vol. 4, no. 1, 2009.
49. F. Mahendra, "Sistem informasi geografis berbasis web inventarisasi jalan kota pekanbaru menggunakan google maps api," Ph.D. dissertation, Universitas Islam Negeri Sultan Syarif Kasim Riau, 2014.
50. G. Van Rossum *et al.*, "Python programming language." in *USENIX Annual Technical Conference*, vol. 41, 2007, p. 36.
51. G. Van Rossum and F. L. Drake Jr, *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995.
52. S. Suparno, D. Fisika-FMIPA, and U. Indonesia, "Komputasi untuk sains dan teknik," *Departemen Fisika-FMIPA Universitas Indonesia*, 2013.
53. E. Utami and S. Raharjo, *logika algoritma dan implementasinya dlm bahasa python di gnu/linux*. Penerbit Andi, 2004.
54. B. Santoso, G. Serpong, and I. Tangerang, "Bahasa pemrograman python di platform gnu/linux," *Jurnal Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Komunikasi Universitas Multimedia Nusantara Gading Serpong Tangerang*, 2009.
55. E. Utami and M. Ridho, "Pemrograman terdistribusi menggunakan python merupakan alternatif yang sederhana dan menarik dalam membangun sistem sekelas superkomputer dengan alokasi dana yang rendah."
56. H. Irfani and D. Dafid, "Bahan ajar dasar pemrograman," 2016.
57. H. ARFIAN, "Rekayasa algoritma winnowing pada sistem pengelolaan skripsi sebagai upaya menekan tingkat plagiarisme," *Skripsi, Fakultas Ilmu Komputer*, 2012.
58. A. Suprianto, "Aplikasi sistem informasi geografis berbasis web pemetaan lokasi panti sosial di kota pangkalpinang."