BAB II
LANDASAN TEORI

2.1 Konsep Dasar Sistem

Terdapat dua kelompok di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurenya dan yang menekankan pada komponen atau elemennya.

Menurut [Jogiyanto Hartono,1999]

Pendekatan sistem yang lebih menekankan pada prosedur mendefinisiksn sistem sebagai berikut :

"suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersana-sama untuk melakukan suatu kegiatan atau untuk menyelsaikan suatu sasaran yang tertentu"

Pendekatan sistem yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut :

"sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu"

2.1.1 Karakteristik Sistem

Suatu sistem mempenyai karakteristik atau sifat-sifat yang tertenyu yaitu :

1. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling bekerja sama membentuk satu kesatuan.

2. Batasan Sistem

   Batasan sistem merupakan daerah yang membatasi antara satu sistem dengan sistem lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem

   Lingkungan luar dari suatu sistem adalah apapun batas dari sistem yang mempengaruhi operasi sistem.

4. Penghubung Sistem

   Penghubung sistem merupakan penghubung antara satu sub sistem dengan sub sistem yang lainnya.

5. Masukan Sistem

   Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) adalah energi yang dimasukan agar sistem tersebut dapat beroperasi, atau masukan sinyal (*signal input*) adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran Sistem

   Keluaran sistem adalah hasil dari energi yang diolah dan di klasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolahan Sistem

   Pengolahan sistem adalah suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah suatu masukan menjadi keluaran.

8. Sasaran Sistem

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*) , jika sistem tidak mempunyai sasatan, maka operasi sistem tidak akan ada gunanya.

2.1.2 Klasifikasi Sistem

Sistem dapat di klasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem di klasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*phsycal system*)
2. Sistem di klasifikasikan sebagai sistem alamiah (*nature system*) dan sistem buatan manusia (*human made system*)
3. Sistem di klasifikasikan sebagai sistem tentu (*Deterministic system*) dan sistem tak tentu (*Probabilistic System*)
4. Sistem di klasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*opened system*)

2.2 Konsep Dasar Informasi

"Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerimanya" **(Jogiyanto H.M., 2001 : 8)**

Sumber informasi adalah data. Data merupakan fakta atau kenyataan yang menggambarkan suatu kejadian-kejadian atau *event* nyata, kemudian dirumuskan

ke dalam sekelompok simbol atau lambang-lambang yang teratur yang menunjukkan kualitas, tindakan atau hal-hal lain.

Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi, sehingga informasi sangat penting di dalam suatu organisasi. Karena informasi dapat berguna bagi suatu organisasi atau seseorang dalam mengambil suatu keputusan menurut JOG [6]. Informasi (*information*) dapat didefinisikan sebagai berikut :

"Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya."

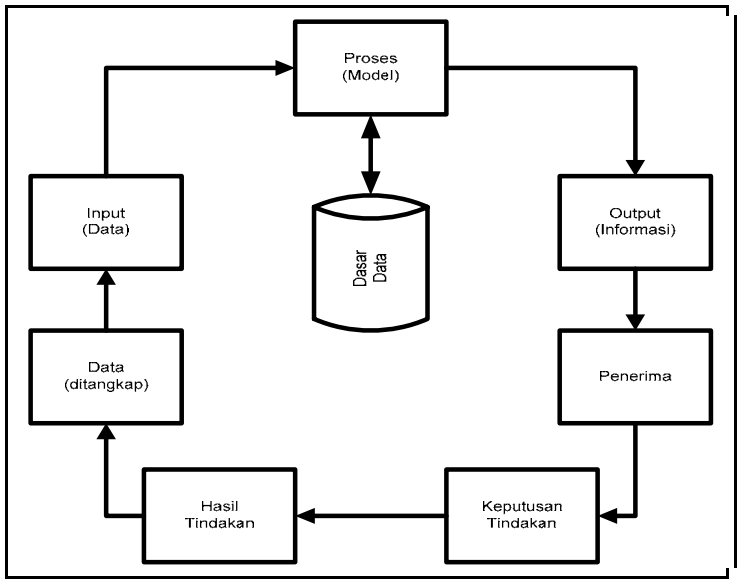
Sedangkan menurut AGS [1] informasi dapat didefinisikan sebagai berikut:

"Informasi adalah suatu kerangka kerja dimana SDM (manusia, komputer) dikoordinasikan untuk mengubah masukan (data) menjadi keluaran (informasi) guna mencapai sasaran perusahaan."

2.2.1 Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses yang tertentu. Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu

model dan seterusnya membentuk suatu siklus. Siklus ini oleh John Burch disebut dengan siklus informasi (*information cycle*). Siklus ini disebut juga dengan siklus data (*data processing cycles).*



Gambar 2.1. Siklus Informasi

(Sumber : JOG)

2.2.2 Kualitas Informasi

Informasi yang baik adalah informasi yang berkualitas. Informasi yang berkualitas ditentukan oleh hal-hal sebagai berikut:

1. Relevan (*Relevance*)

   Informasi tersebut mempunyai manfaat untuk pemakainya dan harus sesuai dengan yang dibutuhkan.

2. Dapat dipercaya (Realibility)

   Informasi yang akan diberikan dapat dipercaya kebenarannya dan mempunyai data-data yang lengkap dan jelas sumber-sumber datanya.

3. Tepat pada waktunya (*Timeliness*)

   Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak mempunyai nilai lagi, oleh karena itu informasi harus *up to date*.

4. Akurat (*Accurate*)

   Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan informasi, harus jelas mencerminkan maksudnya. Komponen-komponen data yang akurat adalah sebagai berikut:

   a. *Completeness*, yaitu informasi yang dihasilkan atau yang dibutuhkan memiliki kelengkapan yang baik, karena bila informasi yang dihasilkan sebagian-sebagian tentunya akan mempengaruhi dalam pengambilan keputusan atau menentukan tindakan secara keseluruhan.
   b. *Correctness*, yaitu kebenaran informasi dapat dipertanggungjawabkan dan mempunyai bukti-bukti dan fakta yang kuat.
   c. *Security* atau keamanan, dalam hal ini informasi yang dikirimkan ke setiap orang yang membutuhkannya perlu pengawasan karena struktur pengecekan dapat memutuskan jika informasi yang sensitif ditujukan kepada pemakai yang tidak sah kepada pihak yang salah.

5. Ekonomis

   biaya pembuatan informasi murah dan memberikan manfaat yang besar bagi pemakai.

2.3 Konsep Dasar Sistem Informasi

Sebuah sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut.

Selain itu sistem informasi dapat didefinisikan sebagai berikut :

1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi.
3. Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Informasi merupakan hal yang sangat penting dalam pengambilan keputusan, permasalahannya adalah dimana informasi tersebut didapat. Informasi dapat diperoleh dari sistem informasi. Robert A Leitch dan K. Roscoe Davis mendefinisikan sistem informasi sebagai berikut:

"Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan

pihak luar tertentu dengan laporan-laporan yang diperlukan." **(Jogiyanto H.M., 2001 : 8)**

Komponen-komponen sistem informasi adalah sebagai berikut:

1. Perangkat keras *(hardware)*

   Perangkat keras *(hardware)* terdiri dari komputer

2. Perangkat lunak *(software)*

   Perangkat lunak *(software)* berupa program-program aplikasi yang akan digunakan, yaitu merupakan kumpulan dari perintah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.

3. Data

   Data merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

4. Prosedur

   Prosedur merupakan dokumentasi prosedur atau proses sistem, tata cara atau penuntun operasional (aplikasi) dan teknis

5. Manusia

   Manusia adalah pengguna dari sistem informasi.

2.4 Sistem Temu-kembali Informasi

Ledakan informasi menyebabkan masyarakat akan mengalami kesulitan mendapatkan informasi yang cepat, padat dan relevan dengan kebutuhannya. Untuk mengatasi hal tersebut diperlukan suatu sistem temu-kembali informasi.

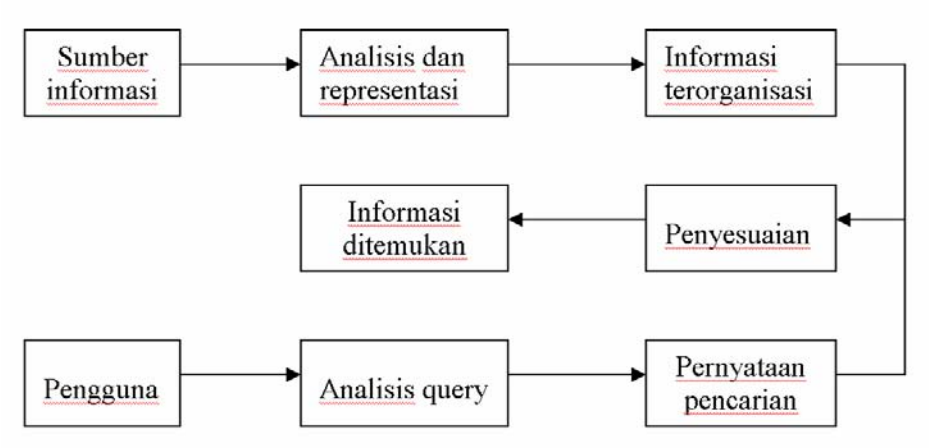
Menurut Davies & Weeks (1995) dalam Stolt (1997), tahun 1982 pertumbuhan informasi meningkat dua kali lipat setiap 5 tahun. Tahun 1988 diprediksi informasi meningkat dua kali lipat setiap 2.2 tahun dan tahun 1992 berubah lagi menjadi setiap 1,6 tahun. Kecendrungan ini akan selalu berubah dan saat ini terjadi peningkatan informasi dua kali lipat setiap satu tahun.

Kecepatan perubahan dan penambahan informasi menyebabkan dibutuhkannya suatu sistem yang dapat mengakses dan menyediakan berbagai informasi tersebut. Saat ini telah banyak dari berbagai informasi tersebut dapat diakses secara elektronik melalui WWW atau *internet* dengan menggunakan berbagai mesin penelusur (*search engine*). Perbedaan mesin penelusur yang satu dengan yang lain sangat tergantung pada teknik temu-kembali informasi dan teknik pengindeksan yang dipakai.

Sistem temu-kembali informasi pada prinsipnya adalah suatu sistem yang sederhana. Misalkan ada sebuah kumpulan dokumen dan seorang *user* yang memformulasikan sebuah pertanyaan (*request* atau *query*). Jawaban dari pertanyaan tersebut adalah sekumpulan dokumen yang relevan dan membuang dokumen yang tidak relevan. Secara matematis hal tersebut dapat dituliskan sebagai berikut :

$Q \xrightarrow{2^n} D$, dimana Q = pertanyaan (queri), D = dokumen, n = jumlah dokumen, 2^n = jumlah kemungkinan himpunan bagian dari dokumen yang ditemukan. Sistem temu-kembali akan mengambil salah satu dari kemungkinan tersebut.

Sistem temu-kembali informasi pada dasarnya dibagi dalam dua komponen utama yaitu sistem pengindeksan (*indexing*) yang menghasilkan basis data sistem dan temu-kembali yang merupakan gabungan dari *user interface* dan *look-up-table*. Pada bagian selanjutnya akan dijelaskan berbagai macam sistem pengindeksan dan teknik-teknik temu kembali informasi yang telah dikembangkan.



Gambar 2.2. Garis Besar Sistem Temu Kembali Informasi

2.5 Sistem Temu-kembali Informasi Berbasis Hiperteks

Pada awalnya, hiperteks dan temu-kembali informasi merupakan bidang penelitian yang berbeda satu dengan yang lain. Hiperteks berkisar pada masalah *user-disorientation*, strategi pengembangan dokumen hiperteks, dan mekanisme konversi dokumen tekstual menjadi bentuk hiperteks (Ellist,1996). Sedangkan temu kembali informasi bergerak pada topik manipulasi kueri, konsep basis data tekstual, dan relevansi dokumen terhadap kueri (Bodhitama,1997). Penggabungan kedua bidang ini dapat memecahkan masalah-masalah dalam bidang temu

kembali informasi. Misalnya, sistem temu kembali informasi yang didasarkan pada penggunaan operator *Boolean,* mengandalkan kemampuan pemakai dalam memformulasikan kueri. Hal ini sering mempersulit pengguna. Dengan adanya sistem hiperteks, hal ini dapat di permudah dengan penyediaan antar muka yang memakai pencarian dengan metode *browsing*.

Smeaton (1991) di dalam *Ellist* (1996) juga menyatakan bahwa hiperteks dan temu-kembali informasi itu saling berkomplemen satu sama lain. Hiperteks membutuhkan lebih banyak *searching* sedangkan temu-kembali informasi membutuhkan lebih banyak *browsing*. Hal yang dimaksud adalah hiperteks akan semakin baik jika disertai dengan fasilitas *search,* dan temu-kembali informasi membutuhkan *browsing* dalam melakukan pencarian yang efisien. Adapun maksud dari *searching* adalah berusaha mendapatkan atau mencapai tujuan spesifik sedangkan *browsing* adalah mengikuti suatu path sampai mencapai suatu tujuan. Menurut Brown(1988) didalam Agosti(1993), *browsing* itu bisa diibaratkan dengan **From Where to What**. Maksudnya adalah kita tahu dimana posisi kita dalam *database* dan kita ingin tahu apa yang ada disana (*database*). Sedangkan *Searching* bisa diibaratkan dengan **From What to Where**. Maksudnya adalah kita tahu apa yang kita inginkan dan kita ingin menemukan dimana dia didalam *database*.

Penggabungan sistem temu-kembali kedalam basis hiperteks lebih dikenal dengan nama search engine, dimana sistem ini dapat dibagi kedalam dua kategori berdasarkan sumber informasinya yaitu:

1. *Worldwide Search Engine*

   Worldwide Search Engine adalah suatu sistem temu kembali informasi yang mengambil data-data dari berbagai *server* di seluruh penjuru dunia. Data-data tersebut diambil melalui program yang disebut dengan “robot” atau “bot”. Program inilah yang melakukan pencarian data pada setiap *server*, yang kemudian dikirim ke *server* pusat pada selang waktu tertentu.

2. *Local Search Engine*

   Local search engine adalah suatu sistem temu kembali informasi yang mengambil data-data dari *server* tertentu saja. Kata “*local*”, yang berarti lokal atau setempat, memberi penekanan akan lokasi sumber data yang akan digunakan. *Local search engine* tidak dirancang untuk mengarungi belantara internet seperti *worlwide search engine*. Tujuan implementasi *local search engine* dimaksudkan untuk pencarian pada objek spesifik dan lebih kecil lingkupnya dibandingkan internet sendiri.

Mengenai pemilihan penerapan sistem temu-kembali berbentuk *local search engine* atau *worlwide search engine* tergantung kepada masalah atau jenis informasi yang akan kita sediakan. Penerapan kedua kategori ini hanya akan mempengaruhi cara sistem pengindeksan dari temu kembali. Sedangkan teknik *retrieval* dan rancangan penerapan teknik pada hiperteks akan sama saja, baik pengindeksannya secara *local search engine* ataupun *worldwide search engine*.

2.6 Vektor Space Model

Vector space model adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu *query*. Pada model ini, *query* dan dokumen dianggap sebagai vektor-vektor pada ruang n-dimensi, dimana n adalah jumlah dari seluruh *term* yang ada dalam leksikon. Leksikon adalah daftar semua *term* yang ada dalam indeks.

Salah satu cara untuk mengatasi hal tersebut dalam vektor space model adalah dengan cara melakukan perluasan vektor. Proses perluasan dapat dilakukan pada vektor query, vektor dokumen, atau pada kedua vektor tersebut. Vektor space model juga mempunyai beberapa kriteria sebagai berikut:

a. *Keyterm* berdasarkan model
b. Memberikan penyesuaian bagian dan pengurutan dokumen
c. Prinsip-prinsip dasar

Adapun beberapa prinsip-prinsip dasar tersebut sebagai berikut:

1) Dokumen direpresentasikan melalui vektor *keyterm*
2) Ruang dimensi t direntangkan dengan 1 set *keyterms*
3) Query direpresentasikan dengan oleh vektor *keyterms*
4) Kesamaan dokumen *keyterm* dikalkulasikan berdasarkan pada jarak vektor

d. Keperluan

Adapun beberapa keperluan vektor space model sebagai berikut:

a) Untuk *keyterms* dalam mempertimbangkan vektor-vektor dokumen
b) Untuk *keyterms* dalam mempertimbangkan pertanyaan

c) Untuk ukuran jarak vektor *keyterms* dengan dokumen

e. Kinerja vektor space model

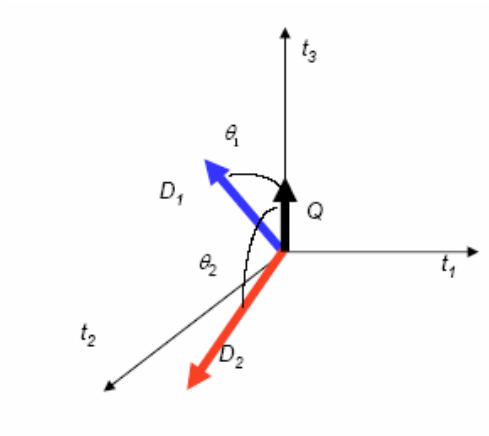
Adapun beberapa kinerja dari vektor space model adalah sebagai berikut:

(a) Efisien

(b) Mudah dalam representasi

(c) Dapat diimplementasikan pada *document-matching*

Teknik vektor space model space ini adalah menghitung nilai cosinus sudut dari dua vektor, yaitu W dari tiap dokumen dan W dari kata kunci.



Dimana: t = kata di database

D = dokumen

Q = kata kunci

2.7 Teknik *Boolean*

Teknik *Boolean* merupakan suatu cara dalam mengekspresikan keinginan pemakai ke sebuah kueri dengan mamakai operator-operator *Boolean* yaitu : “and”, “or”, dan “not”. Adapun maksud dari operator “and” adalah untuk menggabungkan istilah-istilah kedalam sebuah ungkapan, dan operator “or” adalah untuk memperlakukan istilah-istilah sebagai sinonim, sedangkan operator “not” merupakan sebuah pembatasan. Pada teknik *Boolean* sederhana, kueri diproses sesuai dengan operator yang digunakan dan menampilkan dokumen berdasarkan urutan dokumen ditemukan. Sedangkan pada teknik *Boolean* berperingkat, dokumen diperingkat berdasarkan bobot dari dokumen. Adapun pembobotan dari masing-masing dokumen berdasarkan aturan sebagai berikut :

A and B $\rightarrow$ $D_{1A\cap B}, D_{2A\cap B}, \ldots \rightarrow d_{1A\cap B} > d_{2A\cap B} >$

dengan $d_{A\cap B} = \min(d_A, d_B)$

A or B $\rightarrow$ $D_{1A\cup B}, D_{2A\cup B}, \ldots \rightarrow d_{1A\cup B} > d_{2A\cup B} >$

dengan $d_{A\cup B} = \max(d_A, d_B)$

Not A $\rightarrow$ $U - d_A$

Dimana d_A menyatakan bobot istilah A pada dokumen D. Bobot istilah ini didapat dari hasil proses *Indexing*. $\text{Min}(d_A, d_B)$ berarti bahwa sebuah dokumen di *retrieve* dengan bobot sebesar nilai terkecil dari bobot-bobot istilah yang dipunyainya. $\text{Max}(d_A, d_B)$ berarti bahwa sebuah dokumen di *retrieve* dengan bobot sebesar nilai terbesar dari bobot-bobot istilah yang dipunyainya. Berikut dibawah ini contoh dengan menggunakan teknik *Boolean* berdasarkan rumus savoy.

Tabel 2.1. Hasil kueri “citra and komputer” teknik *Boolean*

	Indeks 1		Indeks 2
1. S048	2.000000	1. S005	0.099570
2. S005	1.000000	2. S048	0.039120
3. S006	1.000000	3. T044	0.031300
4. S030	1.000000	4. S006	0.026080
5. S067	1.000000	5. T005	0.022350
6. T005	1.000000	6. S030	0.013040
7. T044	1.000000	7. S067	0.013040

Rumus $W_{ik} = ntf_{ik} * nidf_k$,

dimana $ntf_{ik} = \frac{tf_{ik}}{Max_j tf_{ij}}$ dan $nidf_k = \frac{\log\left[\frac{n}{df_k}\right]}{\log(n)}$

Keterangan:

1) W_{ik} adalah bobot istilah k pada dokumen i.
2) tf_{ik} merupakan frekuensi dari istilah k dalamdokumen i.
3) n adalah jumlah dokumen dalam kumpulan dokumen
4) df_k adalah jumlah dokumen yang mengandung istilah k
5) $Max_j\ tf_{ij}$ adalah frekuensi istilah terbesar pada satu dokumen.

Hasil penemuan dokumen dan bobotnya pada Tabel 2.1, dapat dijelaskan sebagai berikut :

1. Perhatikan dokumen S048 pada hasil kueri dengan indeks 1 dan indeks 2. Pada indeks 1, dokumen S048 mempunyai bobot 2 atau mempunyai nilai RSV yang tertinggi. Pada hasil indeks 2, dokumen S048 mempunyai bobot 0.039120 atau mempunyai RSV dengan urutan peringkat ke-dua yang lebih kecil dari S005. Walaupun dokumen S048 dihasilkan oleh istilah yang mempunyai frekuensi yang tinggi pada sebuah dokumen dengan memakai indeks 1, tetapi pada sistem pengindeksan Savoy(1993) (indeks 2), istilah yang mempunyai frekuensi tinggi yang menghasilkan dokumen S048 belum menjamin bahwa dokumen S048 itulah yang lebih relevan dengan kueri yang di masukkan. Hal ini disebabkan karena sistem pengindeksan Savoy(1993) tidak hanya mempertimbangkan nilai frekuensi saja, tetapi mempertimbangkan posisi relatif dari istilah tersebut. Dengan kata lain rumus Savoy(1993) mempertimbangkan seberapa besarnya frekuensi istilah jika dibandingkan dengan frekuensi istilah terbesar pada dokumen, dan juga mempertimbangkan posisi dokumen yang mengandung istilah dimaksud terhadap jumlah dokumen keseluruhannya.
2. Kalau dilihat dari hasil indeks 1, dokumen urutan 2 sampai 7 tidak dapat diperingkat dan dianggap sama , karena masing-masing dokumen mempunyai frekuensi kemunculan istilah yang sama. Kalau dilihat pada hasil indeks 2, dokumen-dokumen yang tidak dapat diurutkan pada

indeks 1, dapat diperingkat berdasarkan ketinggian bobotnya. Hal ini dapat dilakukan karena rumus Savoy(1993) tidak hanya berdasarkan frekuensi kemunculan istilah saja, tetapi juga mempertimbangkan jumlah istilah terbesar dalam dokumen dan jumlah dokumen dalam kumpulan dokumen yang ada.

Pada teknik pembobotan ini, bobot istilah telah dinormalisasi. Dalam menentukan bobot suatu istilah tidak hanya berdasarkan frekuensi kemunculan istilah di satu dokumen, tetapi juga memperhatikan frekuensi terbesar pada suatu istilah yang dimiliki oleh dokumen bersangkutan. Hal ini untuk menentukan posisi relatif bobot dari istilah dibanding dengan istilah-istilah lain di dokumen yang sama. Selain itu teknik ini juga memperhitungkan jumlah dokumen yang mengandung istilah yang bersangkutan dan jumlah keseluruhan dokumen. Hal ini berguna untuk mengetahui posisi relatif bobot istilah bersangkutan pada suatu dokumen dibandingkan dengan dokumen-dokumen lain yang memiliki istilah yang sama. Sehingga jika sebuah istilah mempunyai frekuensi kemunculan yang sama pada dua dokumen belum tentu mempunyai bobot yang sama.

Hal yang sama berlaku pula untuk operator “OR” seperti terlihat pada Tabel 2 dan “NOT”.

Tabel 2.2. Hasil Kueri “citra or komputer” Teknik *Boolean*

	Indeks 1		Indeks 2
1. S091	14.000000	1. S006	0.497870
2. S030	12.000000	2. S030	0.497870
3. T039	9.000000	3. S041	0.497870
4. T042	8.000000	4. S091	0.497870
5. S041	7.000000	5. T044	0.398300
6. S006	6.000000	6. S048	0.311170
7. S040	6.000000	7. T005	0.284500
8. T032	6.000000	8. S040	0.248940
9. S048	5.000000	9. T026	0.199150
10. S055	5.000000	10. S093	0.156480
11. S093	5.000000	11. T032	0.156480
12. T002	5.000000	12. T034	0.149360
13. S005	4.000000	13. S055	0.130400
14. S085	4.000000	14. S005	0.125180
15. S086	4.000000	15. S085	0.089420

Dari hasil kueri berdasarkan Tabel 2.2, terlihat perbedaan bobot dokumen yang ditemukan antara indeks 1 dan indeks 2. Hasil dari indeks berdasarkan frekuensi (indeks 1), nilai *Retrieval Status Value* ditemukan mempunyai banyak persamaan. Hal ini disebabkan karena banyaknya persamaan frekuensi dari istilah-istilah pada indeks. Kebanyakan dari istilah yang mempunyai frekuensi tinggi tersebut adalah istilah-istilah yang berhubungan dekat dengan domain dari kumpulan dokumen, dalam hal ini adalah domain komputer.

Sebenarnya sebuah indek yang baik itu tidak mempunyai frekuensi yang terlalu tinggi dan juga tidak mempunyai frekuensi yang terlalu rendah. Akan tetapi sejauh mana menentukan batas atas dan batas bawah dari frekuensi istilah yang baik dijadikan indeks, Pemakaian sistem pengindeksan berdasarkan frekuensi (indeks 1), lebih sesuai bagi pemakai yang menginginkan dokumen yang ditemukan itu adalah dokumen yang mengandung paling banyak istilah yang

terdapat pada kueri. Selain itu pemakai sudah mengetahui persis bahwa informasi yang diinginkannya itu biasanya mengandung suatu istilah yang pasti dan sering terdapat pada informasi yang diinginkannya tersebut. Sistem pengindeksan berdasarkan rumus Savoy(1993), pada dasarnya merupakan pengembangan lebih lanjut dari sistem pengindeksan berdasarkan frekuensi. Sistem pengindeksan ini dapat memberikan bobot istilah yang baik terhadap sebuah dokumen. Walaupun suatu istilah mempunyai frekuensi yang sama, tetapi sistem pengindeksan ini dapat memberikan bobot yang berbeda, dengan cara menambah perhitungan dengan faktor lain seperti jumlah dokumen yang mengandung istilah tersebut, atau jumlah frekuensi istilah terbesar. Bobot dokumen yang dihasilkan lebih variatif, dan juga tidak menutup kemungkinan bahwa bobot sebuah istilah pada beberapa dokumen sama. Sistem pembobotan Savoy(1993) ini, lebih sesuai diterapkan pada dokumen-dokumen yang mempunyai kecendrungan jumlah frekuensi istilah. Sistem pembobotan ini mampu untuk memberikan bobot yang lebih spesifik pada dua dokumen yang punya frekuensi istilah yang sama, sehingga mudah diperingkat.

2.8 Teknik *Extended Boolean*

Teknik *Extended Boolean* berdasarkan p-norm model merupakan pengembangan lebih lanjut dari model *Boolean.* Teknik ini memakai operator yang dikomputasi berdasarkan rumus sebagai berikut :

Query	*Retrieval Status Value* (RSV)
A OR <p> B	$\sqrt[p]{\frac{W_{ia}^{\,p} + W_{ib}^{\,p}}{2}}$
A AND <p> B	$1 - \sqrt[p]{\frac{(1 - W_{ia})^{p} + (1 - W_{ib})^{p}}{2}}$
NOT A	$1 - W_{ia}$

Dimana :

1. p adalah nilai p-norm yang dimasukkan pada kueri.
2. W_{ia} adalah bobot istilah A dalam indeks pada dokumen D_i.
3. W_{ib} adalah bobot istilah B dalam indeks pada dokumen D_i.

Pemeringkatan yang dipakai bisa dua cara :

1. Langsung mengurutkan dokumen (dari besar ke kecil) berdasarkan bobot dokumen yang didapat dengan rumus RSV (*retrieval status value*) di atas.

2. Memakai rumus *Learning Scheme*.

$$RSV(D_i) = RSV_{init}(D_i) + \sum_{k=1}^{r} \propto_{ik} \text{norm} * RSV_{init}(D_k) \quad \text{untuk } i= 1, 2,...., n,$$

Dimana :

a. $RSV_{init}(D_i)$ merupakan *retrieval status value* dari dokumen *i* yang dikomputasi berdasarkan rumus teknik *retrieval* P-norm model.

b. $\propto_{ik}$ merupakan bobot keterhubungan antara dokumen *i* dan *k*. Bobot keterhubungan ini didapat dari nilai *relevance link* yang merupakan hasil dari proses pembelajaran.

Prinsip utama dari model Extended Boolean adalah :

a) Dokumen direpresentasikan dalam ruang term berdimensi n

b) Koordinat x, y dan z ditentukan dengan menggunakan bobot term

c) Tergantung pada *conjunction* atau *disjunction* :

i. Menentukan vektor jarak dari (0,0)

ii. Menentukan vektor jarak dari (1,0)

d) Menghitung jarak

iii. Menggunakan konsep p-norm

iv. Perluasan karakteristik dari extended boolean

Pada Tabel 3, dapat dilihat hasil dokumen *retrieval* dengan menggunakan teknik *Boolean* biasa, *Boolean* berperingkat, dan teknik P-norm, dengan menggunakan sistem pengindeksan Savoy(1993) memakai operator “and”.

Tabel 2.3. Perbandingan Teknik *Boolean* biasa, *Boolean* peringka dan P-norm

Citra and Komputer – Bool. Biasa-		Citra and Komputer -Bool. Peringkat-		Citra and Komputer P-norm, p=10		Citra and Komputer P-norm, p=5000	
1.S005		1.S005	0.099570	1.T032	0.177624	1.S005	0.099695
2.S006		2.S048	0.039120	2.S093	0.177624	2.S048	0.039253
3.S030		3.T044	0.031300	3.S086	0.160106	3.T044	0.031434
4.S048		4.S006	0.026080	4. S013	0.145057	4.S006	0.027710
5.S067		5.T005	0.022350	5.S006	0.114821	5.T005	0.022486
6.T005		6.S030	0.013040	6.S005	0.111546	6.S030	0.016120
7.T044		7.S067	0.013040	7.S017	0.107115	7.S067	0.013177
				8.S021	0.106112	8.S091	0.004531
				9.S043	0.104643	9.S041	0.004531
				10.S030	0.104018	10.T032	0.000755
				11.S048	0.100304	11.S093	0.000755
				12.T044	0.095401	12.S086	0.000755
				13.S091	0.093214	13.S013	0.000709
				14.S041	0.093214	14.S021	0.000601
				15.T005	0.083877	15.S017	0.000601

Berdasarkan Tabel 2.3, hal-hal yang dapat diamati adalah sebagai berikut:

a. Perbedaan antara *Boolean* biasa dengan *Boolean* peringkat terlihat dari bobotnya. Pada *Boolean* biasa tidak mempunyai bobot dokumen karena teknik ini hanya menemukan dan menampilkan dokumen berdasarkan

urutan kata yang ditemukan pada dokumen. Dari hasil dokumen yang ditemukan jika dibandingkan dengan teknik *Boolean* berperingkat terdapat perbedaan yang mendasar dari segi urutan dokumen yang ditampilkan. Pada teknik *Boolean* biasa dokumen yang ditampilkan paling atas belum tentu mempunyai tingkat relevan yang lebih baik dari dokumen dibawahnya karena teknik ini hanya mempertimbangkan ada atau tidaknya kata-kata kueri pada koleksi dokumen dan tidak mengukur urutan tingkat kerelevanan dokumen tersebut dengan kueri yang dimasukkan.

b. Pada teknik *Boolean* berperingkat, telah ada perbaikan dari hasil temu-kembali dimana dokumen yang ditemukan telah diberi bobot dan diperingkat sesuai dengan bobotnya. Ini berarti bahwa pemakai telah diberi kemudahan untuk memilih dokumen yang benar-benar relevan dari dokumen-dokumen hasil yang ditampilkan.

c. Perhatikan hasil kueri operasi teknik p-norm, dengan nilai **p**=10 dan **p**=5000. Pada saat nilai **p**=5000, maka terdapat penurunan bobot yang cukup tajam seperti dokumen S093 dan T032. Dokumen-dokumen yang nilai bobotnya tidak terlalu jauh perbedaan mempengaruhi peringkatnya pada saat nilai **p**=10 adalah dokumen S006, S030 dan S048. Ketiga dokumen ini juga terdapat pada dokumen-dokumen yang dihasilkan oleh operasi *Boolean* dengan operator *"and"*, di mana artinya bahwa ketiga dokumen ini mengandung semua istilah yang ada pada kueri. Sedangkan dokumen-dokumen yang peringkatnya turun

adalah dokumen yang mengandung salah satu istilah yang ada pada kueri. Ada juga dokumen yang peringkatnya naik seperti dokumen S048, T044 dan T005. Naiknya peringkat dokumen ini karena dokumen ini juga mengandung semua istilah pada kueri dan peringkatnya naik seiring dengan makin besarnya nilai **p**. Kalau melihat kembali ke salah satu teori yang mengatakan bahwa jika (T1 AND <p> T2) di mana nilai **p** mendekati ∞ , maka sebuah dokumen akan ditemukan jika kedua istilah T1 dan T2 ada pada dokumen tersebut. Maksudnya adalah jika semakin besar nilai **p**-nya maka dokumen-dokumen yang dihasilkan mempunyai bobot yang semakin kecil (mendekati 0), di mana penurunan bobot bagi dokumen yang mempunyai semua istilah yang ada pada kueri akan sedikit dan sebaliknya dokumen yang tidak mengandung semua istilah pada kueri maka penurunan bobotnya akan tinggi, sehingga dokumen yang diperoleh nantinya akan terpisah antara dokumen-dokumen yang mengandung semua istilah dengan dokumen-dokumen yang tidak mengandung semua istilah. Dokumen-dokumen yang mengandung semua istilah pada kueri akan diurutkan sama dengan dokumen-dokumen yang ditemukan pada teknik *Boolean* (lihat Tabel 3). Hal ini disebabkan karena bobot istilah mempunyai nilai dalam rentang [0,1], sehingga jika dilakukan pemangkatan dengan suatu bilangan yang semakin besar (nilai **p**) maka akan menghasilkan suatu bilangan yang semakin kecil (mendekati 0), dan hal ini menyebabkan bobot istilah yang paling kecil dalam sebuah kueri terlebih dahulu akan

mencapai nilai nol, dan hasil pemangkatan dari rumus RSV dari teknik p-norm akan dipengaruhi oleh bobot istilah yang terbesar. Pada kasus dengan operator "*and*", *inverse* dari hasil pemangkatan yang dipengaruhi oleh maksimal dari bobot-bobot istilah adalah minimal dari bobot-bobot istilah, sehingga hal ini sama dengan perhitungan peringkat dari teknik *Boolean* yaitu bobot dokumen yang didapat berdasarkan minimal dari bobot istilah pada kueri untuk operator "*and*".

Selanjutnya kita coba lihat karakteristik dari masing-masing teknik dengan operator "or", berdasarkan Tabel 2.4 berikut ini.

Tabel 2.4. Perbandingan Teknik *Boolean* peringkat dan p-norm

Citra or komputer (indeks 2) –B.Per--		Citra or <1> komputer (indeks 2)		Citra or <100> komputer (indeks 2)		Citra or <405> komputer (indeks 2)	
1. S006	0.497870	1.S006	0.345678	1.S091	0.659241	1. S091	0.662691
2. S030	0.497870	2.S030	0.339882	2.S041	0.659241	2. S041	0.662691
3. S041	0.497870	3.S091	0.334087	3.S030	0.659241	3. S030	0.662691
4. S091	0.497870	4.S041	0.334087	4.S006	0.659241	4. S006	0.662691
5. T044	0.398300	5.T032	0.244873	5.T032	0.486363	5. T044	0.397619
6. S048	0.311170	6.S093	0.244873	6.S093	0.486363	6. S048	0.310638
7. T005	0.284500	7.S086	0.216117	7.S086	0.429249	7. T005	0.284014
8. S040	0.248940	8.T044	0.214800	8.T044	0.395549	8. S040	0.248514
9. T026	0.199150	9.S013	0.193461	9.S013	0.384250	9. T026	0.198809
10. S093	0.156480	10.S048	0.175145	10.S048	0.309021	10.T047	0.000000
11. T032	0.156480	11.T005	0.153425	11.T005	0.282535	11.T046	0.000000
12. T034	0.149360	12.S017	0.140196	12.S017	0.278455	12.T042	0.000000
13. S055	0.130400	13.S021	0.139094	13.S021	0.276266	13.T039	0.000000
14. S005	0.125180	14.S043	0.137278	14.S043	0.272660	14.T038	0.000000
15. S085	0.089420	15.S040	0.124470	15.S040	0.247220	15.T036	0.000000

Dari hasil kueri dengan operator "*or*" di atas, jika nilai **p**-nya semakin besar maka bobot dari masing-masing dokumen akan semakin tinggi (mendekati 1), dan dokumen-dokumen teratas yang ditemukan merupakan dokumen yang mempunyai maksimal bobot dari bobot istilah-istilah yang ada pada kueri, dan

dokumen-dokumen yang ditemukan akan sama dengan dokumen-dokumen yang ditemukan pada teknik *Boolean.* Kasus operator "*or*" ini sama dengan operator "*and*" yaitu karena adanya pemangkatan dengan suatu bilangan yang semakin besar (nilai **p**), dan bilangan yang dipangkatkan mempunyai nilai dalam rentang [0,1], sehingga hasil pemangkatan dari rumus RSV berdasarkan rumus teknik p-norm untuk operator "*or*" akan dipengaruhi oleh bobot istilah yang terbesar, dan hal ini akan sama dengan cara pembobotan dokumen dengan teknik *Boolean* peringkat yaitu berdasarkan maksimal dari bobot istilah yang ada pada kueri untuk operator "*or*". Untuk lebih jelasnya berdasarkan Tabel 4 di atas, dapat kita amati bahwa teknik p-norm mulai dari nilai p=1 sampai nilai p=405, dokumen yang ditemukan berangsur-angsur seiring dengan penambahan nilai p-nya akan sama diperingkat dengan dokumen yang ditemukan pada teknik *Boolean* dengan operator "or".

Untuk kasus kueri citra or komputer dengan nilai p=405, terdapat dokumen yang mempunyai bobot 0.0 adalah disebabkan karena nilai p-nya yang semakin besar, sedangkan nilai bobotnya dalam rentang [0,1], maka sebelum nilai di akar-kan, nilai bobot yang dipangkatkan dengan nilai p telah menjadi nol terlebih dahulu (lihat rumus teknik p-norm dengan operator "or"), sehingga hasil RSV dari dokumen adalah 0.

Pada bagian sebelumnya telah disinggung bahwa teknik p-norm dengan memakai sistem pengindeksan berdasarkan frekuensi tidak menghasilkan dokumen ter-*retrieve* lebih baik dibandingkan dengan memakai teknik Savoy(1993) yang dimilikinya sendiri. Sedangkan teknik *Boolean* peringkat yang

sebelumnya mempunyai sistem pengindeksan berdasarkan frekuensi, setelah menggunakan sistem pengindeksan Savoy(1993), dapat menghasilkan dokumen ter-*retrieve* yang baik, ditandai dengan dapatnya mengurutkan dokumen-dokumen yang ditemukan, dimana sebelumnya tidak dapat dilakukan (lihat Tabel 1).

Untuk lebih jelasnya tentang teknik p-norm dengan sistem pengindeksan berdasarkan frekuensi menggunakan operator “and” dapat di lihat pada hasil kueri Tabel 2.5 berikut ini.

Tabel 2.5. Teknik p-norm dengan memakai sistem pengindeksan frekuensi

Citra and <5> komputer (indeks 1)		Citra and <10> komputer (indeks 1)		Citra and <100> komputer (indeks 1)	
1. T047	1.000000	1. T046	0.066967	1. T046	0.006908
2. T036	1.000000	2. T038	0.066967	2. T038	0.006908
3. S045	1.000000	3. T025	0.066967	3. T025	0.006908
4. S032	1.000000	4. T009	0.066967	4. T009	0.006908
5. T046	0.129449	5. S089	0.066967	5. S089	0.006908
6. T044	NaN	6. S088	0.066967	6. S088	0.006908
7. T042	NaN	7. S083	0.066967	7. S083	0.006908
8. T039	NaN	8. S078	0.066967	8. S078	0.006908
9. T038	0.129449	9. S076	0.066967	9. S076	0.006908
10. T034	NaN	10. S067	0.066967	10. S067	0.006908
11. T032	NaN	11. S066	0.066967	11. S066	0.006908
12. T031	1.000000	12. S065	0.066967	12. S065	0.006908
13. T026	NaN	13. S060	0.066967	13. S060	0.006908
14. T025	0.129449	14. S053	0.066967	14. S053	0.006908
15. T023	NaN	15. S046	0.066967	15. S046	0.006908

Berdasarkan Tabel 2.5 di atas, dapat diamati bahwa teknik p-norm dengan memakai sistem pengindeksan berdasarkan frekuensi kurang baik menghasilkan dokumen ter-*retrieve*. Hal ini dapat dilihat dari bobot dokumen (RSV) yang didapat. Hal ini disebabkan karena teknik p-norm itu mengharuskan bahwa bobot dari indeks istilah tersebut harus dalam rentang [0,1]. Sedangkan bobot dari

indeks berdasarkan frekuensi adalah besar dari satu, sehingga RSV dokumen yang di-*retrieve* menyalahi kaedah dari teknik p-norm itu sendiri, dimana RSV/bobot dokumen yang didapat tidak bermakna.

2.9 Aplikasi Search Engine dan Permasalahannya

Beberapa tahun yang lalu ketika sebuah konferensi internasional berjudul "Bridging the Gap between Information Technology and Business" diselenggarakan oleh Harvard Business School di San Fransisco, Yahoo! memperkenalkan konsep searching engine-nya kepada para peserta. Tujuannya cukup sederhana, yaitu mencari investor yang mau menanamkan uangnya di perusahaan tersebut karena kebanyakan peserta konferensi adalah para investor kelas kakap. Yang terjadi adalah bahwa setelah konferensi yang dilaksanakan selama 3 (tiga) hari usai, tidak seorang investor-pun mengerti mengenai konsep bisnis yang ditawarkan oleh Yahoo!, walaupun secara intensif telah diterangkan pada setiap kesempatan yang ada. Lain dahulu lain sekarang. Saat ini terlihat bagaimana orang-orang di seluruh dunia berlomba-lomba memburu saham perusahaan yang mengklaim dirinya telah memiliki lebih dari 50 juta pelanggan ini.

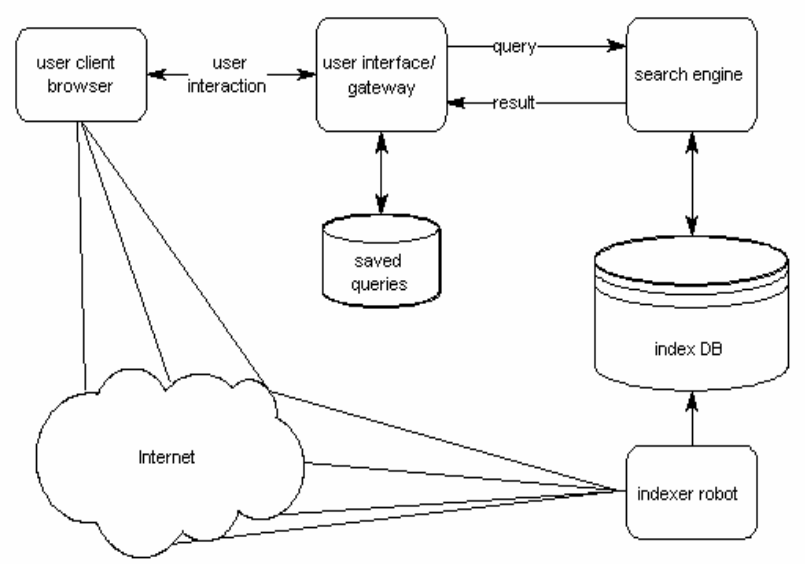
Internet merupakan suatu tempat dimana berjuta-juta situs dapat diakses oleh berjuta-juta orang setiap harinya, tanpa mengenal batasan ruang dan waktu. Situs yang dikembangkan oleh berbagai orang dan perusahaan sangat beragam sifatnya, mulai dari yang hanya berisi data dan informasi ringkas mengenai profil sebuah organisasi sampai dengan yang dapat dipergunakan sebagai sarana untuk

melaksanakan transaksi electronic commerce. Permasalahan yang timbul adalah bahwa perkembangan internet yang sangat cepat (beberapa pakar mengatakan bahwa pertumbuhan jumlah situs di internet bergerak secara eksponensial) telah mengakibatkan terjadinya banjir data dan informasi (information overloaded) sehingga sangat menyulitkan pengguna (user) dalam mencari data dan informasi yang diinginkan. Analogikan dengan sebuah jaringan televisi kabel yang memiliki satu juta channel yang berbeda. Bagaimana seseorang dapat mengetahui apakah ada channel yang menayangkan film favoritnya lengkap dengan jadwal dan lokasinya? Atau bayangkan sebuah perpustakaan negara yang memiliki koleksi satu milyar buku. Bagaimana seseorang dapat tahu buku-buku mana saja yang membahas subjek-subjek tertentu yang diinginkan?

Fenomena inilah yang kemudian berkembang menjadi ide untuk membuat suatu program yang dapat membantu para user internet dalam usahanya untuk mencari data maupun informasi spesifik dalam waktu yang relatif sangat singkat (dalam hitungan detik). Mulailah perusahaan-perusahaan semacam Altavista.com, Excite.com, Yahoo.com, AskJeeves.com, dan lain sebagainya berlomba-lomba untuk membuat mesin pencari (searching engine) yang terbaik. Secara prinsip, tujuan dari sebuah program searching engine adalah **menemukan dokumen atau arsip elektronis di internet yang sesuai dengan kebutuhan atau permintaan pengguna dalam waktu yang sesingkat-singkatnya**. Kedua hal inilah, yaitu kualitas hasil temuan dan waktu pencarian, yang kemudian menjadi pengukur baik tidaknya kinerja sebuah searching engine.

2.10 Arsitektur Mesin Pencari

Penelitian mengenai penerapan sistem temu-kembali berbasis hiperteks telah mulai dilakukan seiring dengan perkembangan internet akhir-akhir ini. Penelitian yang dilakukan Yuwono(1995), menggunakan rancangan/ arsitektur seperti terlihat pada Gambar.



Gambar 2.3 Arsitektur sistem temu-kembali (Yuwono,1995)

Arsitektur yang dirancang ini terdiri dari dua komponen utama yaitu: *Index Builder* dan *Search Engine*. *Index builder* merupakan sebuah sistem pengindeksan yang memanfaatkan "robot" yang berkomunikasi dengan menggunakan HTTP (*Hypertext Transfer Protocol*) untuk mencari informasi yang akan di indeks. Sedangkan *Search engine* merupakan teknik dari temu-kembali dalam menemukan dokumen dan sekaligus mengeksekusi algoritma peringkat dalam menampilkan dokumen. Sedangkan komunikasi antara pemakai dan *search engine* dalam memformulasikan kueri dilakukan melalui *User Interface.* Setelah pemakai

menemukan dokumen yang relevan dengan kueri, dapat langsung melakukan *browsing* ke sumber informasi dalam hal ini adalah alamat tempat *www*.

2.10.1 Segmentasi

Dalam penulisannya, Bahasa Indonesia menggunakan huruf latin. Sebagian besar dari dokumen-dokumen berbahasa Indonesia yang ditemui di web menggunakan karakter ASCII. Dalam Bahasa Indonesia modern juga tidak dikenal adanya huruf-huruf beraksen (contohnya: é atau ê), kecuali untuk beberapa kata-kata yang diserap dari bahasa asing dalam bentuk aslinya. Meskipun demikian, tanda hubung '-', angka dua '2', dan tanda pangkat '^{2}' memiliki fungsi tersendiri sehingga memerlukan penanganan khusus. Bentuk jamak dalam Bahasa Indonesia dituliskan dengan cara mengulangi kata bentuk tunggalnya. Kata yang diulang tersebut dihubungkan dengan tanda hubung '-', contohnya 'buku-buku'. Pengulangan adalah bentuk yang resmi dalam Bahasa Indonesia, namun demikian, khalayak umum juga terbiasa untuk menuliskan bentuk jamak tersebut dengan menggunakan angka dua maupun tanda pangkat, contohnya 'buku2' dan 'buku^{2}'. Bentuk penulisan tersebut juga banyak ditemui dalam dokumen-dokumen berbahasa Indonesia di Internet, terutama di media-media dan situs-situs informal (forum diskusi, e-mail, situs remaja, dsb). Bentuk pengulangan ini juga kadang-kadang mengindikasikan pekerjaan yang dilakukan berulang-ulang bahkan membentuk kata baru, contohnya mata-mata. Ditambah lagi aturan yang menyatakan bahwa kata ulang tersebut dapat diberi imbuhan (contoh: berjalan-jalan) dan kata imbuhan dapat pula diulang (contoh: petani-

petani). Hal ini menambah rumit proses segmentasi kata, karena jika ditemui kata ulang, kita harus menentukan apakah kata tersebut dapat dianggap setara dengan betuk tunggalnya (contoh: apakah ‘petani-petani’ setara dengan ‘petani’). Terlebih lagi, karena bentuk ulang bisa juga berimbuhan, apakah kata ulang berimbuhan itu dapat dianggap setara dengan kata tunggal berimbuhan. Masalah kesetaraan ini memegang peranan penting dalam proses temu-kembali informasi. Karena melalui konsep “setara” inilah kita dapat mengelompokkan kata-kata yang memiliki arti yang hampir sama. Karena itulah, suatu mekanisme khusus perlu dirancang untuk mendukung keunikan bahasa ini.

2.10.2 Pemenggalan Imbuhan

Kata-kata Bahasa Indonesia kaya akan imbuhan. Kurang lebih ada sekitar 35 imbuhan resmi yang disebutkan. Imbuhan-imbuhan ini dapat berupa prefiks (awalan), sufiks (akhiran), konfiks, maupun infiks (sisipan) yang diserap dari Bahasa Jawa. Satu hal yang unik dari Bahasa Indonesia adalah kecenderungan pemakaian imbuhan secara bebas. Secara gamblang, dapat dikatakan bahwa imbuhan-imbuhan dalam Bahasa Indonesia dapat digunakan pada semua kata dan imbuhan-imbuhan tersebut dapat dikombinasikan satu dengan lainnya.

Penggunaan imbuhan secara besar-besaran ini juga menghasilkan tendensi diantara pemakai Bahasa Indonesia untuk menciptakan suatu imbuhan baru atau aturan-aturan baru. Hal ini dapat disadari karena imbuhan dalam Bahasa Indonesia memang berfungsi untuk menciptakan suatu kata baru yang sedikit banyak berhubungan dengan kata dasarnya. Dalam penelitian kami, imbuhan-imbuhan

yang memiliki aturan resmi kami namakan “Imbuhan Standar” dan imbuhan-imbuhan yang belum resmi namun penggunaannya telah memasyarakat, kami namakan “Imbuhan Tambahan”.

Sepanjang pengetahuan kami, ada beberapa implementasi algoritma pemenggalan imbuhan (*stemming*) untuk kata-kata Bahasa Indonesia. Namun hanya satu yang tersedia sebagai bahan perbandingan, yaitu algoritma dari Universitas Indonesia yaitu algoritma *vektor space mode, metode Boolean* dan lain-lain. Algoritma-algoritma tersebut kesemuanya hanya mendukung Imbuhan Standar dan menggunakan kamus atau daftar kata dasar yang digunakan untuk membantu menentukan imbuhan apa yang harus dihilangkan.

2.11 Mengidentifikasi Dokumen Berbahasa Indonesia

Web Indonesia bukanlah suatu lingkungan yang berdiri sendiri. Web Indonesia merupakan salah satu komponen yang membentuk *World Wide Web* dan berhubungan dengan komponen-komponen lainnya dalam Web. Halaman-halaman Web Indonesia berhubungan dengan halaman-halaman Web berbahasa lain, seperti Inggris, Arab, Belanda, dan sebagainya. Karena kami hanya berkeinginan untuk mengindeks halaman-halaman Web berbahasa Indonesia, kami memerlukan sarana pengidentifikasian bahasa yang dapat mendeteksi apakah suatu dokumen ditulis dalam Bahasa Indonesia atau tidak. Masalah pengidentifikasian bahasa adalah masalah yang telah selesai. Algoritma-algoritma yang ada dapat mendeteksi bahasa dari suatu dokumen hampir tanpa melakukan kesalahan sama sekali. Dari semua algoritma yang ada, yang dipandang sebagai

algoritma terbaik adalah algoritma yang mengeksploitasi informasi dari frekwensi *n-gram* dari suatu bahasa. Namun demikian, semua algoritma yang ada hanya mampu mengidentifikasi bahasa dari suatu dokumen dari sekumpulan bahasa-bahasa yang telah ditentukan sebelumnya. Contohnya, jika algoritma-algoritma tersebut kita beri informasi bahwa bahasa-bahasa yang ada adalah Indonesia, Belanda, dan Inggris, maka ketika algoritma itu menemukan dokumen dalam Bahasa Russia, ia hanya dapat menentukan bahwa kemungkinan besar dokumen itu ditulis dalam Bahasa Indonesia atau Belanda atau Inggris, namun ia tidak dapat memastikan bahwa dokumen itu tidak ditulis dalam tiga bahasa yang diketahuinya. Untuk aplikasi dalam Web, penggunaan algoritma demikian tidaklah mungkin, karena kita tidak dapat memprediksi bahasa-bahasa apa saja yang akan digunakan di dalam Web.

2.11.1 Peningkatan Kinerja Algoritma Secara Mandiri Dan Berkelanjutan

Pengumpulan contoh-contoh dokumen yang representatif untuk melatih suatu program sangatlah memakan waktu. Adalah lebih baik apabila algoritma/program tersebut dapat mengumpulkan contoh-contoh dokumen untuk digunakan melatih dirinya sendiri. Didasari oleh kinerja algoritma kami diatas yang cukup memuaskan, kami memutuskan untuk mencoba membuat algoritma tersebut supaya dapat menggunakan keputusannya sendiri demi meningkatkan kemampuannya sendiri. Teknik ini kami namakan Belajar Secara Berkelanjutan (*Continous Learning*).

Algoritma tersebut kemudian akan menentukan dokumen-dokumen mana saja yang ditulis dalam Bahasa Indonesia. Dokumen-dokumen yang dianggap berbahasa Indonesia tersebut akan kemudian digunakan untuk melatih kembali algoritma tersebut. Disini kita lihat proses belajar berkelanjutan. Untuk setiap dokumen baru yang diberikan kepada algoritma diatas, ia akan menentukan terlebih dahulu grup mana dokumen tersebut kemungkinan berasal. Setelah keputusan tersebut dibuat, algoritma tersebut akan menganggap bahwa dokumen tersebut benar-benar berasal dari grup tersebut dan menggunakannya sebagai contoh tambahan yang kemudian dipakai untuk memperbaharui parameter-parameter dalam algoritma itu sendiri.

2.12 Pengideksan dan Bahasa Indeks Dalam Sistem Temu Kembali Informasi

Salah satu faktor yang berpengaruh terhadap sistem temu kembali ialah pengindeksan dokumen. Pengindeksan (*indexing*) mencakup proses pencatatan ciri-ciri dokumen, analisis isi, klasifikasi maupun pembuatan *entri* ke dalam bahasa indeks. Tujuan pengindeksan ialah untuk memungkinkan ditemukannya dokumen yang relevan dengan pertanyaan (*query*) dengan tepat.

Kegiatan pengideksan akan menghasilkan indeks. Meadow (1992 : 69-70) mengemukakan bahwa indeks adalah merupakan cantuman dari bermacam-macam *atribut* yang diharapkan dapat digunakan sebagai dasar pencarian dokumen. Jika *atribut* tersebut berupa subjek, maka indeks yang mewakilinya disebut sebagai indeks subjek. Sedangkan bila *atribut* tersebut berupa pengarang,

maka indeks yang mewakilinya disebut sebagai indeks pengarang. Umumnya kegiatan pengindeksan adalah berupa pengindeksan subjek. Dengan demikian fungsi indeks pada *database* pada prinsipnya adalah sama yaitu sebagai sarana temu kembali.

Tujuan utama dari pengindeksan ialah untuk membentuk representasi dari dokumen dalam bentuk yang sesuai untuk dicantuman dalam berbagai tipe *database* (Lancaster, 1998 : 1). Indeks sebagai representasi dari dokumen diharapkan dapat menggambarkan isi atau subjek yang terkandung di dalam dokumen tersebut, sehingga dapat ditemukan kembali melalui istilah (*index term*) yang digunakan.

Pada dasarnya ada dua jenis bahasa indeks yaitu bahasa alamiah (*natural language*) dan kosa kata terkontrol (*controlled vocabulary).* Bahasa alamiah adalah bahasa dari dokumen yang diindeks. Biasanya bahasa tersebut merupakan bahasa yang tidak terkendali (*uncontrolled vocabulary*). Bahasa alamiah ini umum digunakan dalam komunikasi dan penulisan ilmiah. Sedangkan kosa kata terkontrol dapat berupa indeks subjekmaupun tesaurus.

Ditinjau dari sisi sistem temu kembali informasi, tesaurus adalah suatu daftar pengendali (*authority list*) istilah-istilah khusus yang digunakan dalam sistem temu kembali informasi. Akan tetapi bila ditinjau dari segi fungsinya tesaurus adalah sarana pengawasan istilah yang digunakan untuk penerjemahan bahasa alamiah dokumen ke bahasa yang lebih terkendali. Tesaurus berisi sejumlah istilah indeks dengan menggunakan bahasa yang terkendali, sehingga sering disebut juga dengan bahasa terkontrol (*controlled language*). Tujuan utama

tesaurus adalah juga untuk memudahkan temu kembali dokumen, dan untuk mencapai konsistensi dalam pengindeksan dokumen pada sistem simpan dan temu kembali informasi.

Dalam bahasa pengindeksan kosa kata terkontrol seperti tesaurus, istilah yang digunakan untuk menyatakan kandungan atau isi suatau dokumen telah dibakukan dalam suatu daftar indeks yang disusun secara alfabetis, misalnya *Sears List of Subject Heading, Library of Congress Subject Heading, Macro Economics Thesaurus, DDC Index*, dan sebagainya. Sedangkan pengindeksan bahasa alamiah adalah pengindeksan yang dilakukan pada semua istilah baik dari judul, abstrak, maupun dari teks lengkap (*full text*) dokumen, terkecuali *stop word* atau daftar kata umum yang tidak digunakan dalam penelusuran (Rowley, 1992 : 272). Semua istilah indeks yang dihasilkan adalah bergantung kepada bahasa dokumen itu sendiri, dan semuanya itu dapat merupakan representasi dari dokumen itu. Mengingat volume pengindeksan dalam bahasa alamiah ini sangat besar, maka biasanya dilakukan oleh komputer.

Bahasa alamiah dan kosa kata terkontrol adalah dua bahasa hasil dari pengindeksan yang sama-sama dapat dipergunakan sebagai representasi dokumen. Kedua bahasa pengindeksan tersebut digunakan pada waktu pemasukan (*input*) data ke *database*, dan akan digunakan juga pada waktu pencarian atau penelusuran (*output*) informasi dari *database*.

2.13 Mesin Pencari (*Search Engine*)

Satu dekade ini internet berkembang demikian pesat. Jumlah situs tumbuh secara eksponensial dan nyaris tak terkendali. Jutaan topik dan layanan disuguhkan untuk memenuhi kebutuhan manusia, dan hampir tidak satu topik pun yang tidak dimiliki internet.

Dengan melimpahnya sajian di internet, hal itu tidak lantas membuat mudah bagi orang-orang atau tepatnya *user* untuk menemukan apa yang mereka cari. Sering kali, karena begitu banyak pilihan yang ditawarkan, *user* justru menjadi bingung apa yang mesti dilakukan dan dari mana memulai? Kondisi ini lebih merepotkan lagi bagi mereka yang belum bisa berhadapan dengan “Dunia Maya”.

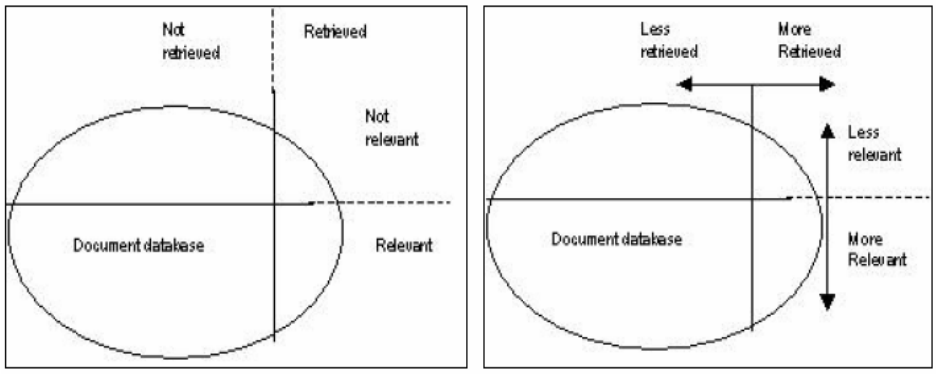
Manusia dengan setumpuk idenya berusaha menemukan jalan keluar dari setiap masalah. Berbagai metode dicoba untuk menemukan solusi yang dapat mengeliminasi kendala-kendala diatas dan lagi-lagi hal ini dimaksudkan agar bisa menguntungkan manusia, atau setidaknya membuat mudah pekerjaan manusia. Dalam konteks di mana manusia harus menemukan satu titik informasi diantara lautan informasi, sudah barang tentu yang dibutuhkan adalah sebuah mesin pencari yang pintar, yang dapat menyuguhkan apa-apa yang *direquest* dalam waktu yang sesingkat-singkatnya dan dengan tingkat akurasi yang dapat diandalkan.

Apabila dilihat dari karakteristik, yang harus mampu melakukan pencarian atas berbagai topik dalam kecepatan tinggi, yang dibutuhkan manusia adalah apa yang disebut “*Search Engine*”. *Search Engine* tidak lain sebuah mesin pencari yang ulet dan teliti, yang melakukan eksplorasi atas informasi-informasi yang

direquest tanpa memandang kapan, di mana dan oleh siapa itu dilakukan. *Search Engine* dirancang oleh insinyur-insinyur teknologi informasi sefleksibel mungkin, mudah digunakan dan dengan konstruksi yang dapat dikostumasi.

Mesin pencari biasanya menggunakan indeks (yang sudah dibuat dan disusun secara teratur) untuk mencari *file* setelah pengguna memasukkan kriteria pencarian. Informasi yang ditampilkan mengandung atau berhubungan dengan suatu istilah spesifik. Lancaster mendefenisikan temu kembali informasi sebagai proses pencarian dokumen dengan menggunakan istilah luas untuk mengidentifikasi dokumen yang berhubungan dengan subjek tertentu. Mengenai efektivitas kinerja dari sebuah *search engine* selalu dikaitkan dengan tingkat relevansi hasil pencarian. Meskipun demikian, defenisi konsep relevansi telah menjadi perdebatan bertahun-tahun dalam ilmu informasi. Schamber menyatakan bahwa konsep relevansi adalah sebuah fenomena yang multidimensional dan dinamis. Sementara itu, Saracevic menjelaskan tentang bagaimana seharusnya konsep relevansi itu dimaknai, apakah relevansi yang dimaksud dilihat dari aspek algoritma, topic, kognitif, situasi, atau motivasi.

Konsep penilaian biner pada hasil pencarian yang mendikotomikan antara yang relevan dan yang tidak relevan, yang ditemukan (*retrived*) dan tidak ditemukan, seperti yang digagas oleh Salton dan McGill kini mengalami pergeseran. Mizarro mengatakan seiring dengan meningkatnya penggunaan *search engine*, maka penilaian biner tidak lagi mencukupi.



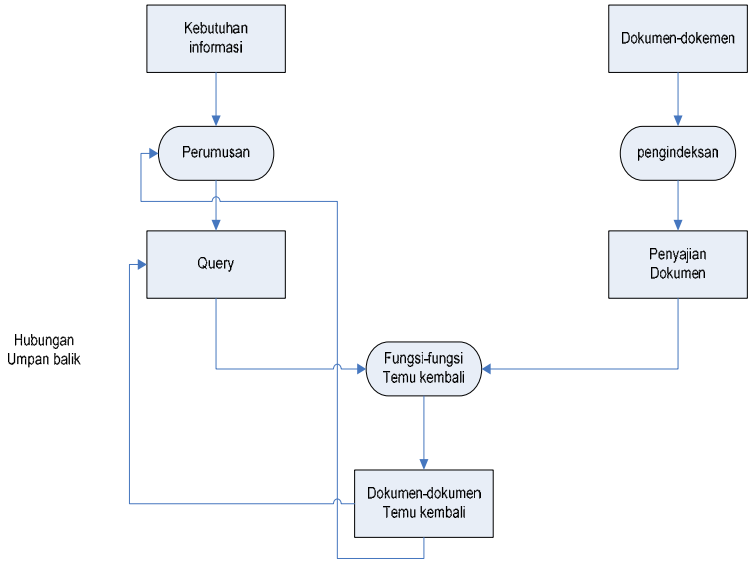
Gambar 2.4. Perubahan konsep penilaian biner ke penilaian kontinu

Dari gambar diatas dapat dilihat bahwa efektivitas kerja *search engine* tidak semata-mata dilihat dari relevan atau tidak relevan atau ditemukan dan tidak ditemukannya informasi yang diinginkan. Akan tetapi, semua hasil pencarian *search engine* akan ditampilkan secara kontinu dari hasil yang lebih relevan sampai yang kurang relevan dengan metode pemeringkatan. Dengan demikian, konsep penilaian biner bergeser menjadi penilaian kontinu. Semakin banyak sumber informasi *online* memunculkan berbagai penelitian tentang efetivitasi sistem temu kembali informasi.

Metode evaluasi dilakukan dengan 3 tahapan. Pertama, mengumpulkan literatur-literatur berhubungan baik tercetak maupun elektronik. Kedua, menyeleksi *search engine* dan menentukan *query* yang akan digunakan untuk penelusuran. Ketiga, penelusuran dengan *search engine*. Setelah semua data diperoleh, langkah selanjutnya adalah melakukan analisis.

2.13.1 Gambaran Umum Mesin Pencari (*Search Engine*)

Mesin pencari umumnya terdiri atas tiga unit utama, yaitu: penjelajah *web*, modul pengindeks dan temu-kembali, serta fasilitas antarmuka untuk pengguna. Penjelajah web, seperti namanya, bertugas untuk menjelajahi web dan mengumpulkan dokumen-dokumen yang diinginkan. Dalam aplikasi penjelajah web ini juga dirancang untuk mampu membedakan dokumen yang ditulis dalam Bahasa Indonesia dari dokumen-dokumen lainnnya. Satu-persatu, dokumen-dokumen yang diinginkan akan diproses lebih lanjut oleh modul pengindeks, yang terlebih dahulu akan mem-*parsing* atau mensegmentasi dokumen itu sehingga diperoleh daftar kata-kata yang ada didalamnya. Daftar kata itu kemudian disaring dengan membuang kata-kata yang ada di daftar *stop-word*. Kata-kata yang tersisa itu kemudian dihilangkan imbuhan-imbuhannya melalui proses *stemming* sehingga didapatkan daftar kata dasar yang dapat mewakili dokumen tersebut. Daftar kata dasar inilah yang kemudian diasosiasikan dengan dokumen dan URL (*Universal Resource Locator*) dari dokumen tersebut. *Query* juga diproses dengan cara yang hampir sama.



Gambar 2.5. Proses Pencarian

Modul temu-kembali akan membentuk daftar dokumen-dokumen yang diperkirakan relevan dengan *query* yang diberikan pengguna. Dokumen-dokumen tersebut kemudian diurutkan berdasarkan bobot kemiripan masing-masing dokumen dengan *query* pengguna.

2.14 Klasifikasi *Web Search Services*

Sebenarnya agak sulit untuk mengklasifikasikan situs-situs mesin pencari. Disamping karena belum adanya referensi format yangdisepakati, kita juga memiliki banyak kriteria untuk membedakan antara *engine* yang satu dengan yang lainnya. Hal ini banyak dipengaruhi oleh corak dan warna yang diperlihatkan oleh *engine-engine* itu sendiri.

Meski demikian, bila kita meninjaunya secara umum dengan mengambil titik tolak dari *content*, fitur-fitur, *desain*, serta kemudahan penggunaannya, setidaknya kita mendapatkan tiga kategori *engine* berikut:

a. Search Engine

Dengan *search engine*, *user* memasukkan *keyword* baik berupa kata, kalimat, angka, kode, atau kombinasi dari semuanya untuk menampilkan daftar dokumen atau alamat *situs* yang berhubungan dengan *keyword* yang di-*input*. Pencarian dalam *search engine* tidak terbatas dan *user* dapat meng-*input query* paling spesifik sekalipun.

b. Directory

Mesin *directory* adalah pilihan terbaik untuk tujuan eksplorasi *situs*, tetapi kurang tepat untuk dokumen. Disini user dapat melakukan pencarian berdasarkan kategori, seperti ekonomi, bisnis, komputer, ilmu pengetahuan, kesehatan, pendidikan, dan lain sebagainya.

c. Library

Online library merupakan direktori dari kemupulan direktori. *Database* yang disuguhkan memuat *file-file* dokumen atau referensi. Umumnya koleksi library dengan sangat hati-hati dipilih dan dievaluasi oleh pakar-pakar kepustakaan dengan sasaran validasi dan kualitas.

2.15 Mengevaluasi Aplikasi *Search Engine*

Sebuah *search engine* akan berhadapan langsung dengan *interface user*, melayani *user* menemukan *resource-resource* spesifik melalui berbagai metode pencaria. Dalam hal ini kebanyakan user tidak ambil peduli dengan apa

sesungguhnya yang dilakukan *search engine* guna memenuhi *request-request* yang masuk kepadanya. Yang penting begitu menekan tombol, *search engine* harus menyodorkan hasilnya dalam satu atau beberapa detik.

Bila kita kaji secara teknis, sebuah alikasi search engine sebetulnya memikul beban kerja yang berat untuk menangani satu buah query saja. Sebagai mana dijelaskan sebelumnya, search engine akan melewati tahapan-tahapan proses yang kompleks untuk menemukan hasil akhir. Disamping itu ia juga memperhatikan faktor-faktor ketepatan, dan ini bukanlah tugas yang ringan. Hanya aplikasi-aplikasi yang cerdas saja yang mampu melakukannya.

2.16 Anatomi *Search Engine*

Setiap *search engine* pasti memiliki fasilitas dimana para pemakai internet (*netter*) dapat mengetikkan kata kunci yang akan menjadi referensi pencarian. Bila kita tinjau dari anatomi dan strukturnya, sebuah aplikasi search engine dibentuk oleh sekumpulan program terotomasi. Mereka dikenal sebagai *spider* atau *crawlers*, yang berfungsi mengambil informasi dari internet. Kesatuan dari fungsi-fungsi ini sering juga disebut *crawling*.

Secara garis besar, crawling search engine pada umumnya terdiri dari lima bagian utama

1. Crawler
2. Spider
3. Indexer
4. Database (the “index”)

5. Result Engine

1) Crawler

Crawler adalah program terotomasi yang memproses *link-link* yang ditemukan dalam halaman-halaman web, yang kemudian menunjukkan *spider* untuk mengunjungi situs-situs tertentu yang baru ditemukan. Saat *spider* mendownload halaman-halaman, ia melakukan 'pengintaian' atas link-link. Mereka dapat dengan mudah melakukannya karena selalu menemukan *item* yang sama. Selanjutnya *crawler* menunjukkan ke mana spider harus pergi (didasarkan link-link dan list URL yang ada). Seringkali link-link baru yang dia temukan saat kunjungan kembali ke sebuah situs kemudian ditambahkan ke dalam list. Saat anda menambahkan sendiri sebuah URL ke search engine, rogram crawler akan mengecek *request* anda dengan mengunjungi situs tersebut.

2) Spider

*Spide*r adalah bagian program otomatis yang berperan untuk *mendownload* dokumen-dokumen yang ditemukan dalam suatu web atas referensi *crawler*.

Program spider bekerja sangat sibuk dan dalam kecepatan tinggi. Layaknya sebuah *browser,* ia melakukan *download* banyak halaman (dalam *environment* yang besar bisa mencapai ratusan ribu). Kebanyakan *spider* tidak melakukan *download* atas *image*, dan tidak diperintahkan untuk mengirim. Jika anda penasaran apa yang dilihat dan diseleksi *spider* saat berkunjung ke sebuah halaman *web*, silahkan klik kanan *button mouse* anda, kemudian pilih *view source*

pada menu yang muncul. Anda akan melihat kode-kode *script* dari halaman *web* tersebut. Inilah yang dipelajari oleh *spider*.

3) Indexer

Program *indexer* memiliki tugas “membaca” halaman-halaman yang telah di download spider. Di sini *indexer* mempelajari tentang apakah subjek dari *site* anda tersebut. Beberapa kata yang terkategori umum akan di *reject* (seperti *and*, *it*, *the*, dan semacamnya). *Indexer* akan memeriksa kode HTML guna menemukan kata-kata penting yang dikandung oleh situs yang dibaca. Kata-kata yang dicetak tebal (*bold*), miring (*italic*), dan *tag-tag header* akan lebih diperhatikan. Analisis juga akan difokuskan terhadap informasi-informasi *meta*, termasuk *tag-tag keyword* dan deskripsi.

4) Database (the index)

Sesuai dengan namanya, *database* adalah suatu ruang dimana informasi-informasi yang diperoleh oleh indexer akan disimpan. Pada prakteknya, volume *database* dari sebuah *Search Engine Internet* senantiasa bertambah dari waktu ke waktu karena disana ada ribuan *webmaster* yang memproduksi halaman baru dalam setiap datiknya! Tidak pelak, untuk sebuah *search engine* sederhana pun, akan dibutuhkan *space disk* yang besar.

5) Result Engine

Sebagai program penutup dan sekaligus berperan dalam menggenerasikan hasil pencarian (dari database) atas setiap *query* yang *diinput user*, program ini adalah bagian terpenting dalam *search engine*.

Result Engine adalah porsi *customer facing*. Oleh sebab itu disini diperlukan usaha optimasi yang maksimal karena ia akan berhadapan langsung dengan *interface user*. *Result Engine* harus mampu memperhatikan *output* yang akurat dan relevan dengan apa yang *direquest user*.

Saat seorang *user* mengetik sebuah *keyword* atau kalimat yang dicari, *result engine* harus memutuskan halaman-halaman mana saja dari sekian ribu halaman yang lebih mendekati dengan keinginan *user*. Metode yang berperan mengolah keputusan ini adalah apa yang disebut “algoritma”. Sebagai informasi tambahan, *spider* dan *crawler* sering dipanggil juga “robots” terutama dalam konteks dokumen-dokumen *official robots exclusion standar*.

2.17 Sejarah Hypertext

Istilah hypertext sendiri sudah digunakan sejak lebih dari 30 tahun yang lalu. Bahkan di tahun 1945, sudah ada tulisan yang memimpikan suatu mesin yang bisa berfungsi sebagai mesin hypertext. Beberapa perkembangan yang dapat dicatat dalam sejarah antara lain sebagai mana dikemukakan oleh *Jacob Nielsen* dalam *Short Hystory of Hypertext*:

1. **1945** Vannevar Bush mengajukan proposal mesin *Memex*

2. **1965** Ted Nelson menggunakan istilah "*hypertext*" dalam buku *Literary Machines*
3. **1967** The *Hypertext Editing System* and FRESS, Brown University, Andy van Dam
4. **1968** Doug Engelbart dan beberapa peneliti mendemokan *NLS* system
5. **1978** *Aspen Movie Map* hypermedia videodisk pertama , Andy Lippman, MIT
6. **1984** Filevision dari Telos; hypermedia database dibuat untuk komputer Macintosh
7. **1985** *Symbolics Document Examiner*, Janet Walker
8. **1985** *Intermedia*, Brown University, Norman Meyrowitz
9. **1986** OWL memperkenalkan *Guide*, hypertext untuk umumpertama
10. **1987** Apple memperkenalkan *HyperCard*, Bill Atkinson
11. **1987** *Hypertext'87* menyelenggarakan konfrensi pertama mengenai hypertext
12. **1991** *World Wide Web* di CERN menjadi global hypertext pertama, Tim Berners-Lee
13. **1992** *New York Times Book Review* , cerita sampul hypertext fiksi
14. **1993** *Mosaic*, National Center for Supercomputing Applications
15. **1993** *A Hard Day's Night* film berformat hypermedia pertama
16. **1993** Hypermedia encyclopedias terjual lebih banyak dari bentuk cetakannya

Nielsen melihat perkembangan hypertext berdasarkan tonggak (mile stone) dimana terjadi perkembangan yang cukup signifikan dalam sejarah. Pengembangan ini berupa alat, teknologi, ataupun penggunaan hypertext itu sendiri. Nielsen menganggap ide dari Vannevar Bush mengenai mesin pintar dengan link-link nya yang bisa tersimpan sebagai ide awal pengembangan hypertext.

Neil Ridgway menganggap ada tiga tokoh utama yang paling penting dibalik pengembangan hypertext. Tokoh-tokoh tersebut adalah *Vannevar Bush*, *Engelbert*, serta *Nelson* dengan ide dan ciptaannya masing-masing.

a) Pertama adalah *Vannevar Bush* dengan mesin Memex-nya. Tahun 1945 Bush sudah memperkirakan akan pertumbuhan literature *sains* yang sangat pesat, dan dia berkeinginan untuk menciptakan suatu cara dimana informasi dalam jumlah besar dapat dilihat (*browse*) sekaligus. Dalam salah satu artikelnya, Bush menjelaskan tentang bagaimana pikiran manusia bekerja dengan merangkai informasi. Dia mengaplikasikan konsep ini menjadi suatu mesin yang disebut Memex, yang memungkinkan pengguna merangkai beberapa potongan yang relevan menjadi suatu informasi, dari dokumen yang berbeda. Ide ini dikenang orang sebagai ide pertama yang menjelaskan *hypertext*.
b) Kedua adalah *Doug Engelbarts* dengan mesin oN Line System (NLS/ Augment) yang dibuatnya. Pada tahun 1963, Engelbarts menjelaskan suatu sistem komputer yang akan memperkaya kemapuan intelektual manusia, dengan memungkinkan pengguna berinteraksi menggunakan

beberapa perangkat kerjasama khusus. Hasilnya adalah peningkatan dalam jumlah informasi yang bisa dikelola secara efektif oleh kemampuan dasar manusia tersebut. NLS ini diimplementasikan 5 tahun kemudian pada Stanford Research Institute. Mesin ini memungkinkan pengguna merelasikan bagian antar dokumen atau dalam dokumen itu sendiri.

c) *Nelson* dengan sistem XANADU nya. Pada saat NLS sedang dibuat, Ted Nelson juga sedang mematangkan sebuah ide mengenai mesin pemerkaya kemampuan tersebut. Sistem yang diciptakan Nelson memungkinkan pengeditan atau perubahan isi dari dokumen yang ada sesuai dengan format aslinya saja. Dengan menggunakan *link* ke belakanag, maka isi dokumen asli dapat diketahui. Dokumen ini disimpan dalam satu media penyimpanan sehingga perubahan-perubahannya bisa dilacak dengan mudah. Sistem yang diciptakan *Nelson* ini memungkinkan penggunan membuat hubungan atau keterkaitan antar bagian dokumen. Nelson menuangkan idenya ini melalui sebuah buku yang diberi judul *Literary Machines*

2.17.1 Pengertian Hypertext

Smeaton(1991) di dalam Ellist(1996) juga menyatakan bahwa hiperteks dan temu-kembali informasi itu saling berkomplemen satu sama lain. Hiperteks membutuhkan lebih banyak *searching* sedangkan temu-kembali informasi membutuhkan lebih banyak *browsing*. Hal yang dimaksud adalah hiperteks akan

semakin baik jika disertai dengan fasilitas *search*, dan temu-kembali informasi membutuhkan *browsing* dalam melakukan pencarian yang efisien. Adapun maksud dari *searching* adalah berusaha mendapatkan atau mencapai tujuan spesifik sedangkan *browsing* adalah mengikuti suatu path sampai mencapai suatu tujuan. Menurut Brown(1988) didalam Agosti(1993), *browsing* itu bisa diibaratkan dengan *From Where to What*. Maksudnya adalah kita tahu dimana posisi kita dalam *database* dan kita ingin tahu apa yang ada disana (*database*). Sedangkan *Searching* bisa diibaratkan dengan *From What to Where*. Maksudnya adalah kita tahu apa yang kita inginkan dan kita ingin menemukan dimana dia didalam *database*.

Dalam terminologi yang diberikan oleh Konsorsium W3, *hypertext* diartikan sebagai suatu teks yang tidak dibatasi oleh *linieritas* (*Text which is not constrained to be linear)*. Definisi ini disamakan dengan *Hypermedia*, dimana *Hypermedia* dinyatakan sebagai Multimedia Hypertext, dan digunakan dengan arti yang sama atau dapat saling dipertukarkan.

Neil Ridgway menyatakan bahwa Hypertext adalah perluasan dari bentuk tradisional atau linier text menjadi text yang tidak linier. *Ridgway* mencontohkan penggunaan sistem komputer yang baru yang memungkinkan penggunanya membuat referensi dari bagian mana saja didalam teksnya ke suatu tempat, baik dalam dokumen atau file yang sama ataupun ke dokumen atau file eksternal. *Hypertext* juga sering disebut sebagai *non linier text*, karena dalam bagian-bagian tertentu bisa merujuk ke bagian lain secara tidak sekuen sesuai dengan alamat rujukan yang diberikan. Rujukan atau *link* ini diantaranya yang membedakan

dengan teks konvensional sebelumnya. Pembaca bisa mengikuti jalur sesuai dengan preferensinya, dan tidak harus melalui jalur yang sama.

Nancy Kaplan dalam tulisannya yang dimuat dalam jurnal *Computer-Mediated Communication Magazine* mendefinisikan *Hypertext* sebagai : *multiple structurations within a textual domain*. Kaplan mencontohkan dengan sebuah buku yang bisa dibaca kapan saja, dimulai dari bagian mana saja, dan bagian-bagian tersebut bisa memiliki hubungan dengan bagian lain. Dia mencontohkan suatu bentuk yang disebutnya sebagai *proto* hypertext dalam bentuk cetakan seperti *Ensiklopedia*, kamus, dan bentuk-bentuk buku manual atau buku panduan. Dimana pembaca bisa mulai dari bagian mana saja sesuai dengan keinginannya, dan setelah itu bisa pergi ke bagian atau halaman mana saja sesuai dengan kebutuhannya. Contoh lain yang diberikan *Kaplan* adalah program pertelivisian di Inggris yang disebut *CEEFAX*, dimana dengan menggunakan beberapa tombol yang ada di remote kontrol kita bisa mengakses atau melihat daftar acara yang disajikan.

George P. Landow mendefinisikan hypertext sebagai : *text composed of blocks of words (or images) linked electronically by multiple paths, chains, or trails in an open-ended, perpetually unfinished textuality described by the terms link, node, network, web , and path.* *Landow* mendefinisikan sebagai satu kumpulan kata-kata atau gambar, yang terhubung secara elektronik dengan banyak jalur, kaitan, atau jejak yang terbuka, secara terus-menerus tidak pernah selesai secara tekstual, yang dijelaskan dengan terminologi link, simpul, jaringan, web dan jalur. Defenisi *Landow* menekankan pada kumpulan kata atau gambar.

Kata ini saling terkait, dan bisa melalui beberapa jalur (bukan satu jalur saja). Kaitan antar kumpulan ini terus berkembang dan kemungkinan tidak ada habisnya. *Landow* juga memberikan penekanan pada terminologi link yang menunjukkan hubungan, *node* yang menunjukkan masing-masing simpul atau bagian, jaringan dan web yang berguna untuk menjadi prasarana penghubung masing-masing simpul.

Darlene Cardillo & Kimberly Kenyon dari University at Albany menjelaskan Hypertext sebagai suatu bentuk presentasi nonlinier dengan banyak pilihan jalur informasi, yang memungkinkan pembaca berinteraksi secara interaktif dengan setiap teks. Lebih jauh dijelaskan hypertext telah membuat bentuk baru dari lingkungan membaca dan menulis yang mendukung pengembangan materi pembelajaran secara interaktif seperti jurnal akademik, ensiklopedia, referensi, serta bentuk elektronik teks lainnya.

Dari beberapa defenisi tersebut, dapat dilihat adanya beberapa persamaan mengenai hypertext. Dimana hypertext dianggap sebagai bentuk teks yang tidak linier, memiliki link atau terhubung ke teks lain, baik internal maupun eksternal, dan dapat dikembangkan secara bersama-sama. Hypertext juga mengandung unsur elektronik, baik dalam pembuatan, pengaksesan, maupun pengembangan. Walaupun ada bentuk cetak dari hypertext, tetapi sebagian besar menekankan hypertext sebagai media yang diakses secara elektronik.

2.18 Sejarah Internet

Pada tahun 1969 ARPA (Advanced Research Project Agency), sebuah bagian dalam kementerian pertahanan Amerika Serikat amemulai sebuah proyek, yang disatu sisi menciptakan jalur komunikasi yang tak dapat dihancurkan dan disisi lain memudahkan kerja sama antar badan riset diseluruh negeri, seperti juga industri senjata. Maka terbentuklah ARPANet.

Proyek ARPANET merancang bentuk jaringan, kehandalan, seberapa besar informasi dapat dipindahkan, dan akhirnya semua standar yang mereka tentukan menjadi cikal bakal untuk pengembangan protokol baru yang sekarang dikenal sebagai TCP/IP *(Tranmission Control Protocol / Internet Protocol)*

ARPANET dibentuk secara khusus oleh empat universitas besar di Amerika, yaitu Stanford Research Institute, University of california at Santa Barbara, University of California at Los Angeles, dan University of Utah, dimana mereka membentuk suatu jaringan terpadu di tahun 1969, dan secara umum ARPANET diperkenalkan pada bulan Oktober 1972. Pada tahun 1981, jumlah komputer yang tergabung dalam ARPANET hanya 213 komputer, kemudian di awal tahun 1986, bertambah menjadi 2308 komputer, dan 1,3 juta komputer pada tahun 1993. Pada awal tahun delapan puluhan seluruh jaringan yang tergabung dalam ARPANET diubah menjadi TCP/IP, karena proyeknya sendiri sudah dihentikan, dan jaringan ARPANET inilah yang merupakan koneksi utama *(backbone)* dari *internet*.

Proyek percobaan tersebut akhirnya dilanjutkan dan dibiayai oleh NSF *(National Science Foundation)*, suatu lembaga pengetahuan seperti LIPI di

Indonesia. NSF lalu mengubah nama jaringan ARPANET menjadi NSFNET dimana *backbone*nya memiliki kecepatan tinggi yang dihubungkan ke komputer-komputer yang ada di universitas dan lembaga penelitian terbesar di Amerika. Setelah itu pemerintah Amerika Serikat memberikan ijin ke arah komersial pada awal tahun 1950. jelaslah bahwa awalnya *internet* dikenal sebagai suatu wadah bagi para peneliti untuk saling bertukar informasi kemudian dimanfaatkan oleh perusahaan-perusahaan komersial sebagai sarana bisnis mereka. Sat ini pengguna *internet* tersebar di seluruh dunia dengan jumlah mencapai lebih dari 250 juta orang.

2.18.1 Fasilitas Internet

Saat ini jika orang berbicara tentang *internet*, yang mereka maksud adalah *word wide web (www).* Pada kenyatannya *internet* memiliki banyak bagian yang lain, yaitu:

1. *World wide web* disingkat dengan web adalah bagian yang paling menarik dari *internet*. Melalui *web* kita bisa mengakses informasi yang tidak hanya berupa teks, tetapi juga gambar-gambar, suara-suara, film, dan lain-lain. Untuk mengakses web dibutuhkan *software* yang disebut *browser*. *Browser* yang terpopuler saat ini adalah microsoft Internet Explorer dan Netscape Communicator.
2. *Electronic Mail* disingkat *E-mail* adalah surat elektronik yang dikirim melalui *internet*. Dengan fasilitas ini kita bisa mengirim atau menerima *email* dari dan ke pengguna *internet* di seluruh dunia. Jika dibandingkan

dengan pos, fasilitas *email* jauh lebih cepat kita bisa mengirim email ke teman yang ada di luar negeri hanya dalam beberapa menit dan tidak perlu khawatir, surat tersebut akan rusak karena hujan atau hal-hal lainnya. Untuk mengirim atau menerima *e-mail* dibutuhkan program *e-mail* yang terdapat di Microsoft Internet Explorer dan Netscape Communication. Yang perlu dibutuhkan hanya mengetik surat tersebut dan mengirimkannya. Di *internet* juga banyak tersedia *mailing list*, fasilitas yang memungkinkan kita untuk saling berbagi informasi mengenai topik-topik tertentu dengan orang lain. Jika kita bergabung dengan salah satu *mailing list*, kita bisa menerima dan mengirim *e-mail* ke semua anggota *mailing list* tersebut.

3. *Telnet*. Dengan menggunakan telnet kita bisa menggunakan komputer untuk berhubungan dengan komputer lain dan mencari atau mengambil informasi-informasi yang ada di komputer tersebut.
4. *File Transfer Protokol* disingkat FTP. Melalui *software* FTP, anda bisa mengirim data atau file di sebuah komputer ke komputer lain. Proses mengirim file dari sebuah komputer ke komputer anda disebut dengan proses *download*, sedangkan proses mengirim file dari komputer anda ke komputer lain disebut dengan *upload*. Jika anda pernah membuat *homepage*, maka kata *upload* tidak asing lagi.
5. *Gopher* adalah sistem dimana pemakai dapat mengakses informasi dari komputer lain. Beda *gopher* dengan *web* adalah tidak bisa menampilkan

gambar, melainkan hanya teks. Oleh sebab itu *gopher* banyak ditingalkan para pemakai *internet* saat ini.

6. *Chat Group* atau *Internet Relay Chat* (IRC) adalah forum dimana pemakai dapat saling berdiskusi atau berbincang-bincang dengan pemakai lain.
7. *News Group* bisa disebut ruang percakapan bagi para anngota yang mempunyai kemungkinan yang sama. Di *internet* tersedia bermacam-macam news group dengan tema yang berbeda-beda. Untuk dapat menikmati fasilitass ini kta harus terkoneksi ke ISP yang menyediakan fsilitas *news group*, karena tidak semua ISP menyediakannya. Adapun fasilitas yang anda gunakan semuanya merupakan bagian dari *internet*. Melalui fasilitas-fasilitas *internet* itu, kita bisa melakukan banyak hal yang akan menambah pengetahuan dan wawasan kita.

2.18.2 Syarat Bergabung Ke Internet

Untuk berhubungan dengan i*nternet* tentunya harus mempunyai komputer. *Internet* adalah jaringan komputer, oleh karena itu jalan satu-satunya untuk berhubungan dengan *internet* adalah melalui komputer. Selain itu juga diperlukan *harddisk* yang berisi *software* untuk berhubungan dengan *Internet Service Provider* dan internet seperti *web browser*, dan *harddisk* untuk menyimpan informasi-informasi yang anda *download*. Untuk mengakses *internet* tidak cukup hanya dengan komputer saja, tetapi ada alat bantunya seperti:

1. ***Modem***. *Modem* adalah perangkat *hardware* tambahan untuk komputer (baik jenis *card internal* maupun *eksternal* yang terletak di luar komputer). Pada dasarnya *modem* adalah alat yang memungkinkan komputer anda untuk berbicara dengan komputer lain melalui kabel telepon. Kata *modem* berasal dari kata Modulasi Demodulasi yang bisa diartikan sebagai proses perubahan denyut elektronis dari komputer menjadi suara (modulasi) sehingga dapat dikirimkan melalui kabel telepon. Modem penerima akan mengubah nada suara tadi menjadi denyut elektronis kembali untuk diproses selanjutnya oleh komputer (demodulasi).
2. **Telepon Modem** memerlukan telepon untuk melakukan tugasnya. Proses pada saat *modem* terhubung dengan telepon dan anda memulai hubungan *internet service privider* pada dasarnya sama dengan proses anda menelepon biasa. Jika ada yang menelepon anda pada saat menggunakan modem, maka dia akan menerima nada sibuk dan tidak bisa menghubungi anda, karena *modem* dan telepon berkaitan erat, kecepatan *modem* juga akan berpengaruh terhadap rekening telepon yang harus anda bayar setiap bulan, sehingga *modem* yang cepat akan menghemat uang daripada *modem* yang lambat.
3. ***Software***. *Software* diperlukan untuk berhubungan dengan *internet*. Tanpa *software*nya sesuai, anda tidak bisa mengakses *internet*. Contoh *software* yang diperlukan adalah:

a. Microsoft Internet Explorer dan Netscape Communicator adalah *browser* untuk mengakses *web* yang paling baik dan terpopuler saat ini. Kedua *browser* ini juga sekaligus berfungsi sekaligus berfungsi sebagai FTP, *mail* dan *news group*.
b. Cute FTP, Mozilla dan WSFTP, adalah *software* khusus untuk FTP
c. Untuk *e-mail* anda bisa menggunakan *Internet Mail/ Outlook express* yang merupakan bagian dari Netscape Communicator.
d. MIRC, Yahoo Messenger, atau ICQ adalah program untuk *chating* yang paling popular saat ini. Dengan program ini anda bisa berbincang-bincang dengan orang lain melalui *keyboard*.
e. Internet Service Provider. Untuk bisa bergabung dengan *internet*, anda harus mempunyai akses dengan cara berlangganan ke penyedia jasa *internet* atau yang lebih sering disebut *Internet Service Provider* (ISP). ISP adalah perusahaan yang menawarkan jasa pelayanan kepada anda untuk berhubungn dengan *internet*. Untuk mengakses, anda perlu cukup menghubungi Internet Service Privider melalui komputer dan modem, selanjutnya *Internet Service Privider* akan mengurus detail-detail yang diperlukan untuk berhubungan dengan internet termasuk biaya SLJJ koneksi tersebut.
f. Sambungan langsung ke Network Anda dapat menggunakan sebuah komputer yang secara langsung mempunyai hubungan ke INTERNET. Sebagai contoh, Anda mungkin menggunakan sebuah

PC yang merupakan bagian dari sebuah jaringan komputer yang mempunyai hubungan ke INTERNET. Dalam kasus ini, sistem Anda menjadi host INTERNET penuh, yaitu mempunyai alamat elektronik tersendiri.

g. Sambungan dengan menggunakan SLIP/PPP. Untuk menggunakan hubungan dial-up telepon, Anda memerlukan sebuah alat untuk mengkonversi sinyal komputer (digital) menjadi sinyal telepon (analog), dan sebaliknya. Alat untuk mengkonversi sinyal digital ke sinyal analog disebut *modulator*. Sedang, alat untuk mengkonversikan sinyal analog ke sinyal digital disebut *demodulator*. Untuk mengakses ke INTERNET melalui hubungan telepon, Anda memerlukan sebuah modem (*modulator-demodulator*). Selain itu diperlukan juga TCP/IP dan software SLIP atau PPP seperti Linux, Warp, dll.

h. Sambungan langsung ke On-line Service seperti BBS, Compuserve. Untuk menjadi sebuah host INTERNET tanpa harus memiliki hubungan full-time ke INTERNET (yang umumnya sangat mahal), ada sebuah cara mensetup sebuah host INTERNET melalui hubungan telepon. Untuk melakukan hal tersebut, Anda perlu mengadakan perjanjian dengan sebuah host INTERNET yang lain yang bertindak sebagai titik hubungan. Selanjutnya, diperlukan sejumlah program yang disebut sebagai PPP (Point to Point Protocol) dan SLIP (Serial Line Internet Protocol) dalam

workstation. Setelah workstation menghubungi host INTERNET melalui jalur telepon, PPP menyediakan kemampuan TCP/IP untuk workstation tersebut.

2.19 Web Server

Web server merupakan salah satu tulang punggung penyedia layanan di Internet atau server di Internet. Apache merupakan web server yang terpopuler saat ini dimana survey netcraft lebih dari 50% situs di Internet menggunakan Apache sebagai Web Server. Selain karena Apache lebih cepat juga luwes karena dapat di implementasikan untuk berbagai system operasi, mudah dikonfigurasi, dapat ditambahkan modul lain melalui module Apache.

Apache menyediakan fasilitas yang kaya, yang sangat dibutuhkan suatu server serius, seperti otentikasi, pengaturan akses direktori, virtual host, kemampuan URL rewritting, dan juga alias. Kemampuan melakukan content negotiation membuat Apache mampu melayani beragam client secara otomatis, baik untuk berbagai browser yang memiliki kemampuan berbeda. Fungsi log yang dimiliki oleh Apache dapat dikirim melalui proses piping, sehingga dapat dilakukan rotasi log, fillter log, serta melakukan pemisahan log secara langsung.

Awalnya Apache dikembangkan berdasakan keinginan untuk memperbaiki Web server yang saat itu populer (NCSA web server). Tetapi akhirnya mengalami perombakan dan penulisan ulang dan menjadi Web server yang berdiri sendiri dan berbeda dengan NCSA. Kini malah mengalahkan kepopuleran NCSA Web server.

Pada tahun 1999 dibentuk Apache Software Foundation untuk mengurus perkembangan Apache ini. Apache telah membuktikan sebagai web server yang cepat, stabil dengan fitur yang paling kaya di antara web server lainnya. Saat ini proyek Apache telah berkembang dan tidak hanya sekedar Web server. Apache dapat memberikan layanan yang cukup banyak bagi penggunakanya karena memiliki program pendukung yang cukup banyak seperti control akses, CGI, PHP dan lain sebagainya.

2.20 Pengenalan HTML

Hypertext Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman website. HTML merupakan pengembangan dari standard pemanfaatan dokumen teks yaitu *Standar Generalized Markup Language* (SGML).

Dokumen HTML sebenarnya hanya berupa *file* teks biasa yang dilengkapi dengan tanda-tanda khusus (tag) yang menentukan bagaimana teks tersebut ditampilkan. Karena itu, dokumen HTML dapat dibuat menggunakan teks editor sederhana seperti Notepad, EditPlus, dan sebagainya. Namun sekarang ini telah banyak visual editor untuk HTML seperti: Microsoft Front Page, Macromedia Dreamweaver, dan lain-lain.

Tag HTML biasanya adalah tag-tag yang berpasangan dan ditandai dengan simbol “<” dan “>”, sedangkan akhir perintah dari sebuah tag ditandai dengan tanda “/”. Dokumen HTML biasanya disimpan dengan extension .HTM atau .HTML. Untuk membuka dokumen HTML ini, diperlukan sebuah Web Browser.

Struktur sebuah dokumen HTML pada dasarnya dibagi menjadi dua bagian besar, yaitu *header* dan *body*. Masing-masing ditandai oleh pasangan *container tag* <*head*> dan <*body*>. Bagian *head* berisikan judul dokumen dan informasi-informasi dasar lainnya sedangkan bagian *body* adalah data dokumennya. Pengaturan format teks dan pembentukan *link* dilakukan terhadap objeknya langsung dengan ditandai oleh *tag-tag* HTML.

HTML diatur oleh konsorsium WWW (W3C) semua perubahan atas standar bahasa HTML harus disahkan terlebih dahulu oleh WWW. Sejauh ini, HTML telah mengalami berbagai revisi sepanjang hidupnya, standar paling akhir yang sekarang diperkenalkan adalah standar HTML 4.0, yang mendukung antara lain CSS *(Cascading Style Sheet)*, *Dynamic Content Positioning* (Penempatan isi secara dinamis).

2.21 Pengenalan CSS

CSS merupakan kependekan dari **Cascading Style Sheet** yang memungkinkan untuk mendesain (style) tampilan dokumen (terutama HTML) dengan memisahkan isi dari dokumen HTML dengan kode untuk menampilkannya (CSS). Jika memiliki banyak file HTML, hanya perlu satu file CSS, sehingga ketika mengganti jenis huruf pada file CSS maka semua file HTML yang berhubungan pada file CSS tersebut akan berubah. CSS distandarisasi oleh W3C (World Wide Web Consortium). CSS dapat dipasang pada dikumen HTML yang telah jadi.

Ada 3 cara untuk memasang CSS pada dokumen HTML yaitu: External

Style Sheet (file CSS berbeda dari file HTML), Internal Style Sheet (Kode CSS dipasang di dalam tag head HTML) dan Inline Style Sheet (Kode CSS langsung dipasang di tag HTML, tidak direkomendasikan). namun biasanya standar yang digunakan adalah External Style Sheet karena lebih mudah dalam mengelolanya.

Kode program dari CSS:

```
<html>
<head>
<title>Belajar CSS</title>
<style type="text/css">
h1 { font-family: verdana; }
</style>
</head>
<body>
<h1>Belajar CSS</h1>
</body>
</html>
```

CSS terdiri dari dua bagian utama yaitu: selector, dalam hal ini H1 dan deklarasi yang berada diantara kurung kurawal {font-family: verdana}. Didalam deklarasi juga terbagi menjadi dua bagian yaitu property dalam hal ini font-family dan value dalam hal ini verdana. Dalam contoh diatas hanya mengubah sebuah tag yaitu tag <h1> menjadi teks dimana jenis hurufnya menjadi verdana. Agar dapat mengkombinasikan berbagai macam style menjadi satu.

2.22 Pengertian PHP

PHP singkatan dari *Personal Home Page Tools*, adalah sebuah bahasa *scripting* yang dibundel dengan HTML, yang dijalankan disisi server. Sebagian besar perintahnya berasal dari C, Java dan dengan beberapa tambahan fungsi khusus PHP. Bahasa ini memungkinkan para pembuat aplikasi web menyajikan halaman HTML dinamis dan interaktif dengan cepat dan mudah, yang dihasilkan server. PHP dapat dijalankan pada berbagai macam Operating System (OS), misalnya Windows, Linux dan Mac OS. Selain Apache, PHP juga mendukung beberapa web server lain, misalnya Microsoft IIS, Caudium, PWS dan lain-lain.PHP dapat memanfaatkan database untuk menghasilkan halaman web yang dinamis. Sistem manajemen database yang sering digunakan bersama PHP adalah MySQL. Namun PHP juga mendukung system manajemen Database Oracle, Microsoft Acces, Interbase, d-Base, PostgreSQL dan sebagainya.

PHP menawarkan solusi yang lebih luwes. Dengan PHP, developer tidak perlu lagi berurusan dengan dua buah file terpisah seperti itu. *Browser web* mengacu secara langsung ke file yang dituju, yang lalu dibaca oleh server sebagaimana file HTML statis biasa. Bedanya, sebelum dikirim balik ke *browser web*, *server web* memeriksa isi file dan menetukan apakah ada kode di dalam file tersebut yang akan dieksekusi. Bila ada, kode-kode tersebut akan dieksekusi. Hasilnya akan dimasukkan di dalam dokumen yang sama. Server web bekerja secara langsung terhadap *file* yang bersangkutan, tidak memanggil *script* terpisah seperti pada metode CGI. Seluruh kode dieksekusi di Server (oleh karena itu disebut *server-side script*).

PHP membuat proses pengembangan aplikasi menjadi mudah karena kelebihan-kelebihannya yaitu:

1. *Script* (kode program) terintegrasi dengan file HTML, sehingga *developer* dapat berkonsentrasi langsung pada penampilan dokumen *web*nya.
2. Tidak ada proses *compiling* dan *linking*.
3. Berorientasi objek (*Object Oriented*).
4. Sintaksis pemogramannya mudah dipelajari, sangat menyerupai C dan Perl.
5. Integrasi yang sangat luas ke berbagai server database.

PHP tidak terbatas untuk hanya menghasilkan keluaran HTML. Ia juga bisa digunakan untuk menghasilkan gambar GIF, atau bahkan sumber gambar GIF yang dinamis.

Hingga kini PHP sudah berkembang hingga versi ke 5. PHP 5 mendukung penuh ***Object Oriented Programing(OOP)***, integrasi XML, mendukung semua ekstensi terbaru MySQL, pengembangan web services dengan SOAP dan REST, serta ratusan peningkatan kemampuan lainnya dibandingkan versi sebelumnya. Sama dengan web server lainnya PHP juga bersifat open source sehingga setiap orang dapat menggunakannya dengan gratis.

2.23 Pengenalan XAMPP

XAMPP merupakan sebuah tool yang menyediakan beberapa paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP tidak

perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis.

XAMPP merupakan pengembangan dari LAMP (Linux Apache, MySQL, PHP and PERL), XAMPP ini merupakan project non-profit yang di kembangkan oleh Apache Friends yang didirikan Kai 'Oswalad' Seidler dan Kay Vogelgesang pada tahun 2002, project mereka ini bertujuan mempromosikan pengunaan *Apache web server.*

XAMPP sendiri menawarkan berberapa paket pengistallan antara lain :

a. Apache 2.2.6
b. MySQL 5.0.45
c. PHP 5.2.4 + PHP 4.4.7 + PEAR
d. PHP-Switch win32 1.0
e. XAMPP Control Version 2.5 from www.nat32.com
f. XAMPP Security 1.0
g. SQLite 2.8.15
h. OpenSSL 0.9.8e
i. phpMyAdmin 2.11.1
j. ADOdb 4.95
k. Mercury Mail Transport System v4.01b
l. FileZilla FTP Server 0.9.23
m. Webalizer 2.01-10

Cara mengaktifkan Xampp:

1. Instal XAMP, lalu aktifkan web server Apache dan MySQL dari control panel XAMPP.
2. Jalankan browser (IE, Mozilla Firefox atau Opera) lalu ketikkan alamat web berikut : http://localhost/phpmyadmin/ pada address bar lalu tekan Enter.
3. Apabila telah nampak interface (tampilan antar muka) phpMyAdmin bisa memulainya dengan mengetikkan nama database, nama tabel dan seterusnya.

2.24 Sekilas MySQL

Pada awal perkembangannya disebut SQL yang merupakan kepanjangan dari Structured Query Language. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. SQL pertama kali didefinisikan oleh American National Standards Institute (ANSI) pada tahun 1986. MySQL adalah sebuah sistem manajemen database yang bersifat open source. MySQL adalah pasangan serasi dari PHP. MySQL dibuat dan dikembangkan oleh MySQL AB yang berada di Swedia.

MySQL dapat digunakan untuk membuat dan mengola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah dan menghapus data yang berada dalam database. MySQL merupakan sisitem manajemen database yang bersifat ***at relational*.** Artinya data-data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat.

MySQL (*My Strukture Query Language*) atau yang biasa dibaca " mai-se-kuel" adalah salah satu jenis *database* server yang sangat terkenal. MySQL sangat popular karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*. MySQL bersifat *free* pada berbagai *platform* (kecuali pada Windows, yang bersifat *shareware* atau anda perlu membayar setelah melakukan evaluasi yang memutuskan untuk digunakan untuk keperluan produksi). Perangkat lunak MySQL sendiri bisa di download dari http://www.mysql.org atau http://www.mysql.com.

MySQL dalam operasi klien-server melibatkan server daemon MySQL di sisi server dan berbagai macam program serta pustaka yang berjalan di sisi klient. MySQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MySQL yaitu TcX, mengaku menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 Gigabyte data. MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar. MySQL juga dapat menjalankan perintah-perintah ***Structured Query Language (SQL)*** untuk mengelola database-database yang ada di dalamnya. Hingga kini, MySQL sudah berkembang hingga versi 5. MySQL 5 sudah mendukung trigger untuk memudahkan pengelolaan tabel dalam database.

2.25 Pengertian Dreamweaver

Dreamweaver dalah suatu bentuk program editor web yang dibuat oleh makromedia. Dengan program ini seorang programmer web dapat dengan mudah

membuat dan mendesain webnya.

Dreamweaver merupakan editor yang komplit, yang dapat digunakan untuk membuat animasi sederhana yang berbentuk layer. Dengan program ini pembuatnya tidak akan susah-susah mengetik script-script format HTML, PHP, ASP maupun bentuk program yang lainnya

Dengan kelebihan ini, seorang programmer dapat langsung melihat hasil buatannya tanbpa harus di browser. Dreamweaver MX juga memiliki dua bentuk layer, yaitu bentuk halaman *Design* dan halaman *Code*. Dua bentuk layer ini akan mempermudah dalam menambahkan *script* yang berbasis PHP maupun javascript. Dreamweaver MX, selain mendukung pembuatan web yang berbasis HTML, juga dapat mendukung program–program web yang lain diantaranya PHP, ASP, Perl, Javascript, dan lain-lain.