LATIHAN ILMIAH TAHUN AKHIR

WXES 3182

dCleanViewer

(Pembersihan data dalam gudang data dan perlombongan data)

DISEDIAKAN OLEH

NUR-AIDAH BT NARAWI

(WEK990367)

PENYELIA:

EN. TEH YIN WAH

MODERATOR:

PN. FARIZA HANUM BT MD. NASARUDDIN

Laporan Latihan Ilmiah ini diserahkan kepada

Fakulti Sains Komputer Dan Teknologi Maklumat

Universiti Malaya, Kuala Lumpur

Bagi memenuhi sebahagian daripada syarat

Penganugerahan Ijazah Sarjana Muda Sains Komputer Dengan kepujian

SESI 2002/2003

ABSTRAK

NUR-AIDAH BT NARAWI: Pembersihan Data Dalam Gudang Data dan Perlombongan Data. Projek Ilmiah Tahap Akhir, Fakulti Sains Komputer dan Teknologi Maklumat, Universiti Malaya, Sesi 2002/2003.

Data dari sumber dunia sebenar biasanya mengandungi ralat, tidak sempurna, dan tidak konsisten, ianya boleh menyebabkan ralat operator, kecacatan implementasi sistem dan sebagainya. Proses pembersihan data telah dikenalpasti sebagai perkara utama bagi mendapatkan data yang berkualiti dalam gudang data dan perlombongan data. Oleh yang demikian, projek pembersihan data dalam gudang data dan perlombongan data ini dilaksanakan dengan menekankan beberapa teknik dalam pembersihan data.

Proses pembersihan data melibatkan proses pengesanan, penyingkiran ralat dan ketidakkonsistenan data bagi meningkatkan kualiti data dalam gudang data. Di dalam perlombongan data pembersihan data melibatkan 3 proses penting iaitu pengekstrakan, tranformasi dan penyatuan.

Objektif utama projek ini adalah untuk melakukan pembersihan data pada pangkalan data-pangkalan data yang telah ditetapkan dengan menggunakan teknik-teknik pembesihan data yang telah dikenalpasti. Projek ini lebih menekankan pada pembersihan data yang tidak konsisten bagi data yang terhasil dari sumber data berganda.

Metodologi yang dipilih adalah Model Air Terjun dan Pemprototaipan. Metodologi pembangunan sistem ini adalah sebagai garis panduan dalam usaha membangunkan projek ini.

Perisian yang digunakakan dalam perlaksanaan projek ini adalah Microsoft Visual Basic 6.0 dan Microsoft SQL Server 7.0

Adalah di harapkan projek yang bakal di hasilkan ini memberi manfaat dalam menyediakan data yang berkualiti bagi proses membuat keputusan.

PENGHARGAAN

Alhamdulillah, dengan limpah dan kurniaNya saya dapat melengkapkan projek ilmiah ini dalam tempoh masa yang telah ditetapkan. Pelbagai dugaan dan cabaran terpaksa ditempuhi, namun dengan berbekalkan doa dan kesabaran maka segalanya dapat diatasi dengan baik.

Pertama sekali, saya ingin menyampaikan ucapan terima kasih saya kepada En. Teh Yin Wah yang banyak memberikan tunjuk ajar serta sokongan sepanjang kursus ini berjalan dan yang paling utama diatas kepercayaan beliau pada keupayaan saya dalam membangunkan projek ini.

Sekalung penghargaan juga diucapkan kepada moderator projek ilmiah ini iaitu Pn. Fariza Hanum Md. Nasaruddin di atas kerjasama yang baik serta cadangan-cadangan yang bernas bagi meningkatkan kualiti sistem yang bakal disediakan.

Tidak lupa juga, ucapan terima kasih yang tidak ternilai buat kedua ibu bapa saya yang merupakan pendorong bagi saya untuk melakukan yang terbaik. Tidak ketinggalan juga buat teman-teman yang telah terlibat secara langsung atau tidak langsung dalam perlaksanakan projek ini. Terima kasih diatas kesudian dan kerjasama yang telah diberikan. Sesungguhnya hanya Tuhan yang dapat membalas segala jasa yang telah diberikan.

Akhir kata, semoga Tuhan akan memberi rahmat dan mengurniakan ganjaran yang sebaik-baiknya kepada semua. InsyaAllah.

Sekian, terima kasih.

Nur Aidah bt. Narawi
Sarjana Muda Sains Komputer
Fakulti Sains Komputer dan Teknologi Maklumat
Universiti Malaya, Kuala Lumpur
Sesi 2002/2003

KANDUNGAN

APENDIKS

BIBILIOGRAFI

SENARAI JADUAL

SENARAI RAJAH

BAB 1 PENGENALAN

1.0 PENGENALAN

Kewujudan perisian bagi pengumpulan data secara automasi dan kematangan dalam teknologi pangkalan data telah menyebabkan jumlah simpanan data dalam pangkalan data meningkat dengan mendadaknya dan membawa kepada wujudnya keperluan pada gudang data. Gudang data mempunyai koleksi data yang berorientasikan subjek, bersepadu, masa-berbeza (*time-varian*), dan tidak cepat berubah (*nonvolatile*).[6] Ianya dikelola sedemikian bagi membantu pihak pengurusan membuat keputusan. Oleh yang demikian, gudang data perlu menyediakan data yang bersih, dipercayai dan berkualiti bagi mengelak terjadinya kesilapan dalam membuat keputusan.

Perlombongan data adalah merupakan suatu tugas pencarian corak yang menarik daripada sejumlah data yang besar yang disimpan dalam pangkalan data dan gudang data. Penemuan pada maklumat baru boleh diperolehi daripada proses perlombongan data. Pembersihan data merupakan proses penting dalam gudang data dan perlombongan data bagi menghasilkan suatu maklumat baru yang berguna. Menyedari kepentingan ini, projek pembersihan data yang dikenali sebagai **dCleanViewer** dibangunkan bagi membersihkan data yang kotor dan masalah pembersihan data yang ditekankan dalam projek ini adalah mengatasi masalah data yang tidak konsisten. Data yang telah bersih ini kemudiannya akan dimuatkan kedalam gudang data. Adalah diharapkan data-data ini dapat membantu pihak pengurusan dalam membuat keputusan.

1.1 OBJEKTIF PROJEK

Objektif **dCleanViewer** adalah untuk membersihkan data-data bagi perlombongan data dan gudang data yang terhasil dari gabungan dua buah pangkalan data yang telah dikenalpasti. Selain itu, ianya adalah untuk mengkaji teknik-teknik yang telah telah dikenalpasti dalam pembersihan data. Ianya juga adalah untuk mengenalpasti dan mengkategorikan ralat-ralat pada data yang terhasil dari sumber berganda (data-data yang terhasil dari pelbagai sumber atau gabungan pangkalan data).

Berikut merupakan senarai objektif-objektif yang telah digariskan:

1) Menyediakan data yang telah bersih dalam gudang data kepada pengguna gudang data atau pihak pembuat keputusan bagi membuat kajian, rumusan dan keputusan.
2) Merekabentuk dan mengimplimentasi aliran data yang efektif dan efisyen bagi menghasilkan data bersih berdasarkan sumber data yang telah dikenalpasti.
3) Mengaplikasikan teknik pembersihan data yang telah dikenalpasti ke dalam sistem.

4) Mengenalpasti ralat yang wujud dari sumber data yang berganda.

5) Mengaplikasikan konsep Antara Muka Penguna (GUI) dalam membangunkan antaramuka sistem.

6) Menyediakan data yang telah bersih dan berkualiti kedalam gudang data dan memudahkan pihak pengurusan membuat kajian, rumusan, dan keputusan.

7) Mempertingkatkan kecekapan pihak pengurusan dalam membuat keputusan berdasarkan data yang telah disediakan.

1.2 SKOP PROJEK

Secara keseluruhannya, projek ini memfokuskan kepada perlaksanaan proses pembersihan data di dalam gudang data berdasarkan penyatuan dua buah pangkalan data yang telah ditetapkan. Pendekatan pembersihan data bagi projek ini menekankan pada pembersihan data yang tidak konsisten. Skop sistem adalah sebagai garis panduan bagi memastikan sitem ini memenuhi keperluan projek. Projek ini dilaksanakan dengan kriteria-kriteria berikut:

1. Projek ini melibatkan pengabungan 2 buah pangkalan data yang telah dikenalpasti.
2. Hasil akhir projek boleh memaparkan data yang telah dibersihkan.
3. Data bersih yang terhasil akan digunakan oleh Pentadbir Gudang Data.
4. Satu antaramuka dibentuk bagi membenarkan capaian pada data yang telah dibersihkan.

Umumnya, projek ini adalah untuk membersihkan data yang terhasil daripada penyepaduan data-data dari sumber berganda dan seterusnya mewujudkan suatu corak data yang baru yang boleh dijadikan maklumat bagi membantu pihak pengurusan membuat keputusan.

1.3 SASARAN PENGUNA

Sebagai permulaan, penguna sasaran **dCleanViewer** hanyalah Pentadbir Gudang Data sesebuah organisasi.

1.4 HASIL YANG DIJANGKAKAN

Adalah diharapkan hasil akhir projek ini dapat memaparkan data-data yang telah dibersihkan berdasarkan teknik pembersihan yang telah dipilih. Melalui projek ini, Pentadbir Gudang Data dapat melihat, menggunakan dan mengesahkan data yang telah dibersihkan dan seterusnya memasukkannya ke dalam gudang data. Data-data yang tersimpan dalam gudang data ini seterusnya di harapkan dapat membantu pihak pengurusan membuat keputusan.

1.5 PENJADUALAN PROJEK

Proses pembangunan sistem ini terbahagi kepada 2 peringkat:

1) Peringkat Awal (WXES 3181-Semester 1)
2) Peringkat Akhir(WXES 3182-Semester 2)

Peringkat awal perlaksanaan **dCleanViewer** bermula pada awal Jun 2002 sehingga September 2002. Peringkat ini terdiri daripada 2 fasa pembangunan iaitu:

1) Fasa Analisi dan Keperluan Sistem
2) Fasa Rekabentuk

Setelah selesai perlaksanaan fasa-fasa diperingkat awal, fasa berikutnya seterusnya dilaksanakan. Peringkat ini merupakan perlaksanaan sebenar sistem seperti yang ditakrifkan pada peringkat awal. Fasa-fasa yang terlibat pada peringkat ini adalah:

3) Fasa Perlaksanaan (Pengkodan)
4) Fasa Pengujian
5) Fasa Penyelenggaraan Sistem.

Jadual 1.0: Fasa-Fasa Pembangunan Sistem

BIL	FASA	AKTIVITI
1	Kajian awal dan analisis sistem	• Menentukan objektif, skop dan kekangan sistem • Proses pencarian maklumat • Menentukan keperluan sistem • Memilih perisian dan perkakasan yang sesuai bagi sistem • Menyediakan perancangan projek • Memilih dan menentukan mod pembangunan sistem
2	Rekabentuk sistem	• Merekabentuk pangkalan data • Merebentuk antaramuka
3	Perlaksanaan-pengkodan	• Menggunakan perisian Visual Basic dan SQL Server 7.0 • Pengkodan
4	Pengujian sistem	• Menguji modul-modul sistem
5	Penyelenggaraan sistem	• Melakukan perubahan terhadap sistem sekiranya terdapat permasalahan
6	Dokumentasi dan laporan	• Menyediakan laporan projek

Jadual 1.1: Jadual Perancangan

Bil	Fasa	Jun 02	Jul 02	Ogos 02	Sept 02	Okt 02	Nov 02	Dis 02	Jan 03
1.	Kajian analisis keperluan								
2.	Analisis Sistem								
3.	Rekabentuk Sistem								
4.	Pembangunan Sistem								
5.	Pengimplementasian dan Pengujian								
6.	Dokumentasi								

2.0 ORGANISASI BAB

Bab 1 Pengenalan

Bab ini memberikan gambaran awal keseluruhan projek dengan memberi penerangan ringkas tentang definisi projek, objektif, skop, pengguna sasaran, skedul projek, jadual perancangan, hasil yang dijangkakan serta organisasi bab.

Bab 2 Kajian Literasi

Bab ini mengulas tentang kajian permasalahan yang dijalankan sebelum sistem dilaksanakan. Kajian literasi meliputi kajian serta analisa ke atas sistem-sistem terdahulu, kajian berkenaan teknik yang akan digunakan serta kajian terhadap domain bagi sistem.

Bab 3 Metodologi

Satu huraian yang mendalam tentang kaedah penyelidikan dan teknik yang digunakan bagi menyelesaikan masalah sistem yang dikemukakan.

Bab 4 Analisis Keperluan Sistem

Bab ini memaparkan anlisis terhadap sistem yang akan dibangunkan, keperluan fungsian, keperluan bukan fungsian, keperluan perkakasan dan perisian berdasarkan teknik-teknik pengumpulan maklumat yang telah dijalankan.

Bab 5 Rekabentuk Sistem

Bab ini mengandungi rekabentuk skrin atau antaramuka, carta alir serta carta struktur yang terlibat dalam sistem.

Bab 6 Pembangunan Sistem

Bab ini menghuraikan tentang pembangunan sistem yang meliputi beberapa prototaip bagi memenuhi keperluan pengguna.

Bab 7 Pengimplementasian dan Pengujian

Bab ini menerangkan tentang pengujian yang dilakukan terhadap prototaip yang telah disediakan. Ia bertujuan bagi menganalisa setiap keperluan pengguna di dalam sistem ini.

Bab 8 Perbincangan

Bab ini merangkumi aspek penilaian terhadap sistem yang meliputi kelebihan dan kelemahan sistem yang dibangunkan, masalah dan penyelesaian, cadangan serta kesimpulan bagi projek yang dijalankan.

BAB 2 KAJIAN LITERASI

2.0 PENGENALAN

Pembangunan sebuah sistem yang boleh di percayai dan konsisten memerlukan kajian yang terpeinci sebagaimana seperti perlunya membuat perancangan yang baik. Bagi memastikan pemahaman yang baik pada sistem yang akan dibangunkan, satu kajian intensif telah dijalankan pada beberapa kawasan dan sumber seperti Perpustakaan Utama, Perpustakaan FSKTM, dan melayari Internet.

Antara sebab penting kajian literasi adalah untuk memastikan pembangun sistem mempunyai pengetahuan dan maklumat yang mencukupi untuk membangunkan sistem. Ianya merupakan cabaran yang besar sebelum keputusan dibuat untuk membangunkan sistem. Ianya juga dapat membantu pembangun sistem memilih peralatan yang sesuai bagi membangunkan sistem.

Dalam proses pembangunan **dCleanViewer**, kajian telah dibuat bagi memahami beberapa konsep mengenai sistem yang akan dibangunkan. Kajian dan pemerhatian juga dibuat dengan melayari internet bagi mendapatkan maklumat yang berkaitan dengan projek yang akan dibangunkan.

2.1 GUDANG DATA

Koleksi data yang berorientasikan subjek, bersepadu, masa-berbeza(*varian*), dan tidak cepat berubah (*non-volatile*). Gudang data merupakan tempat penyimpanan data daripada sumber berganda bagi tempoh jangka panjang, ianya dikelolakan sedemikian supaya mempermudahkan pihak pengurusan membuat keputusan.[6]

i) Berorientasikan subjek

Disusun dalam subjek-subjek major seperti pelanggan, produk, dan jualan. Ianya memfokuskan pada pemodelan dan analisis data yang membantu proses membuat keputusan dan tidak memfokuskan pada pemprosesan transaksi dan operasi harian.

ii) Bersepadu

Dibina hasil daripada penyepaduan sumber yang berganda dan pelbagai. Selain itu, ianya mengaplikasikan teknik pembersihan data integrasi data.

iii) Masa-berbeza(*varian*)

Julat masa bagi gudang data adalah berbeza berbanding sistem operasian.

- Pangkalan data operasian : nilai data semasa
- Gudang data : menyediakan maklumat dari perspektif sejarah (contoh : 5-10 tahun sebelumnya)

iv) Data tidab berubah (*non-volatile*)

Data yang telah dipindahkan dan diubah disimpan secara berasingan daripada persekitaran operasian. Pengemaskinian data dalam persekitaran operasian tidak memberi kesan dalam persekitaran gudang data.

Langkah-langkah dalam gudang data [17]:

1. perolehan : data diperoleh dari pelbagai sumber
2. penyepaduan : data yang diperoleh dikumpulkan dan kemudiannya disepadukan kepada satu set data melalui kaedah integrasi skima
3. perubahan : data yang telah disepadukan kemudiannya diubah kepada format asas bagi tujuan pembersihan data
4. pembersihan : paras sebenar pembersihan data. Contoh : lewahan data, ralat masukan dan sebagainya
5. persiapan : lanjutan untuk pembersihan data kini disediakan dalam format seperti yang dikehendaki oleh gudang data
6. pindah : set data akhirnya dimuatkan ke dalam gudang data dan seterusnya sedia untuk digunakan

2.2 PERLOMBONGAN DATA

Perlombongan data merupakan suatu tugas pencarian corak yang menarik daripada jumlah data yang besar, yang mana data dapat disimpan dalam pangkalan data, gudang data atau mana-mana tempat simpanan maklumat yang lain. Ia merupakan satu bidang disiplin yang baru yang terhasil daripada bidang-bidang seperti sistem pangkalan data, gudang data, statistik, mesin pembelajaran, pemaparan data, capaian semula maklumat, dan persembahan perkomputeran yang tinggi.[6]

2.3 PEMBERSIHAN DATA

Pembersihan data berkait dengan pengesanan dan penyingkiran ralat pada data bagi meningkatkan kualiti data yang seterusnya menghasilkan maklumat yang berguna. Antara ciri-ciri data yang berkualiti adalah tepat, sempurna , berwibawa dan tidak basi.

Pembersihan data diperlukan terutamanya apabila menyepadukan sumber data yang banyak, yang mana ianya perlu dinyatakan bersama dengan skima penukaran data terhubung. Didalam gudang data, pembersihan data merupakan bahagian utama bagi proses Pengekstrakan, Penyepaduan dan Penyatuan.

2.3.1 PENGENALAN

Masalah kualiti data wujud dalam koleksi data tunggal, seperti fail-fail dan pangkalan data-pangkalan data. Contohnya, di dalam gudang data, kehilangan data atau data tidak sah wujud disebabkan oleh kesalahan ejaan ketika memasukkan data. Keperluan pada pembersihan data meningkat apabila data dari sumber yang pelbagai perlu disepadukan. Ini adalah kerana sumber-sumber berkenaan mungkin mengandungi lewahan data yang sama dalam bentuk perwakilan yang berbeza. Bagi menyediakan capaian data yang tepat dan konsisten, gabungan perwakilan data yang berbeza dan penyingkiran lewahan maklumat menjadi keperluan.

Petunjuk:

⇨	Aliran Metadata	1,3	Ciri-ciri ketikaan (data sebenar)	4	Pemetaan diantara sumber dan skima sasaran
→	Aliran Data	2	Peraturan perubahan	5	Peraturan penapisan dan penyatuan

Rajah 2.0: Langkah-Langkah Dalam Membina Gudang Data: Proses Pengekstrakan, Penyepaduan, dan Penyatuan [14]

Gudang data memerlukan dan menyediakan sokongan yang meluas bagi pembersihan data. Ianya memuatkan secara berterusan sejumlah data yang besar dari sumber yang pelbagai dan ini membawa kepada kemungkinan sebahagian daripada sumber tersebut mungkin mengandungi 'data kotor' adalah tinggi. Tambahan lagi, gudang data adalah digunakan bagi tujuan membuat keputusan, data yang betul dan tepat adalah penting bagi mengelakkan terjadi nya kesilapan dalam membuat keputusan dan kesimpulan.[9] Sebagai contoh, perulangan atau hilang maklumat akan menghasilkan suatu statistik yang tidak betul dan tepat (*'garbage in, garbage out'*). Disebabkan oleh jarak yang besar bagi kemungkinan ketidakkonsistenan data dan ketelusan saiz data, pembersihan data menjadi salah satu masalah utama di dalam gudang data. Semasa proses yang dikenali sebagi proses ETL (Pengekstrakan, Penyepaduan, Penyatuan), seperti ilustrasi dalam **Rajah 2.0 : Langkah-Langkah Dalam Membina Gudang Data: Proses Pengekstrakan, Penyepaduan, dan Penyatuan,** semua pembersihan data biasanya dilaksanakan dalam peringkat data yang berasingan sebelum memuatkan data yang telah diubah kedalam gudang data. Sejumlah peralatan yang besar bagi fungsian yang berbeza diperlukan bagi menyokong tugas-tugas tersebut, tetapi biasanya bahagian penting bagi pembersihan dan penukaran tugas perlu dilakukan secara manual atau menggunakan program tahap-rendah yang sukar untuk ditulis dan dikekalkan.

Data adalah tidak pra-integrasi seperti gudang data tetapi perlu untuk diekstrak, ditukarkan dan dicantumkan semasa masa larian dari sumber berganda. Komunikasi maklum balas dan pemprosesan lewat boleh menjadi penting, yang menjadikan ia sukar

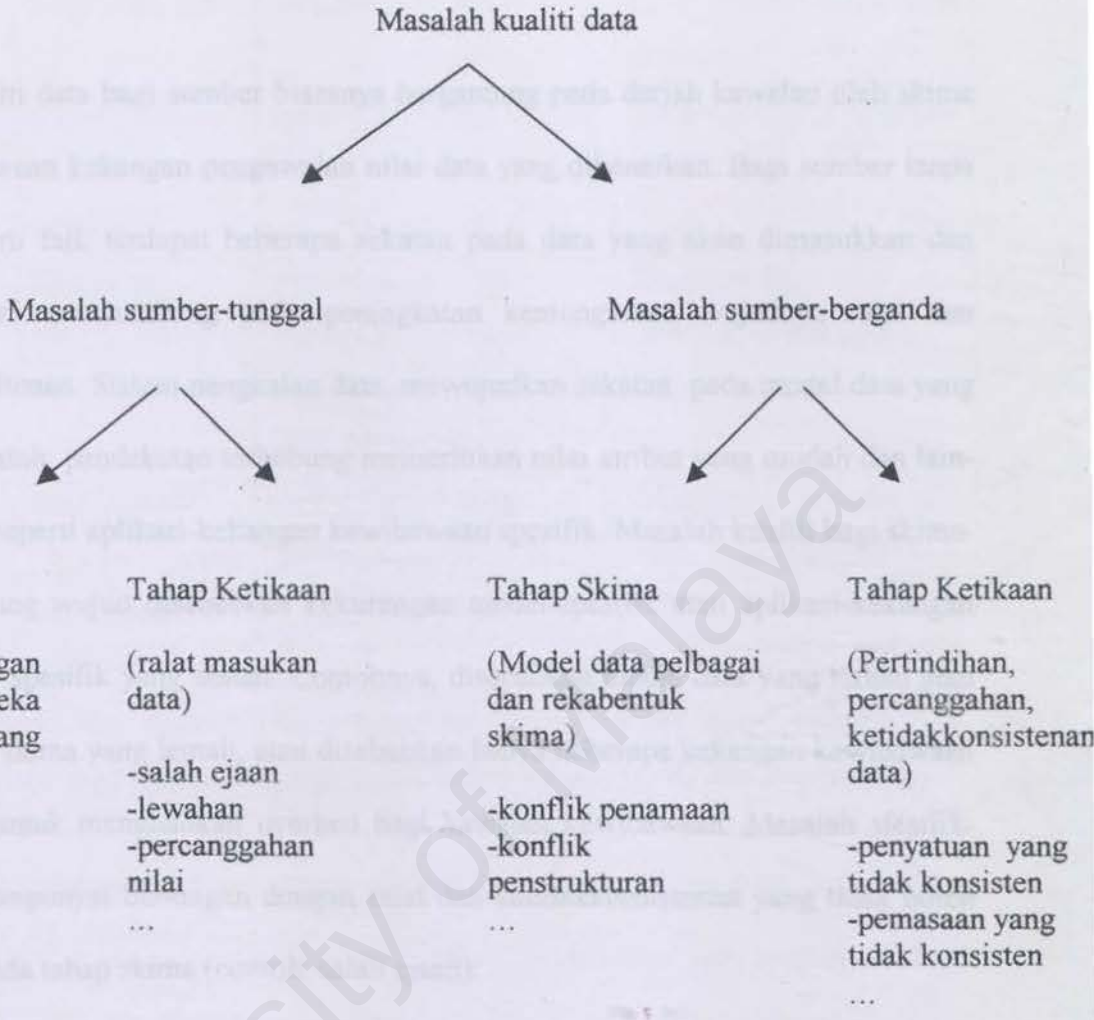
dicapai dalam masa maklum balas yang sepatutnya. Usaha diperlukan dalam pembersihan data semasa pengekstrakan dan integrasi bagi meningkatkan masa maklum balas.

Penyelesaian atau pendekatan dalam pembersihan data perlu memenuhi beberapa keperluan. Pertamanya, ianya perlu dapat mengesan dan menyingkirkan kesemua ralat utama dan ketidakkonsistenan bagi kedua –dua sumber data individu dan ketika mengintegrasikan sumber yang berganda. Penyelesaian perlu disokong dengan peralatan bagi menghadkan penyemakan manual dan pengaturcaraan. Tambahan lagi, pembersihan data tidak boleh di laksanakan secara berasingan tetapi bersama-sama dalam skima – transformasi data terhubung berdasarkan metadata yang menyeluruh. Pemetaan fungsian bagi pembersihan data dan transformasi data yang lain perlu ditentukan dalam bentuk yang diisytiharkan dan boleh diguna semula bagi sumber data yang lain. Terutamanya bagi gudang data, satu infrastruktur aliran kerja perlu disokong untuk melaksanakan semua langkah transformasi data bagi sumber yang pelbagai dan set data yang besar dengan cara yang efisyen dan boleh dipercayai.

2.3.2 ANALISIS MASALAH DALAM PEMBERSIHAN DATA

Bahagian ini mengelaskan masalah kualiti data yang utama yang perlu diselesaikan melalui pembersihan data dan transformasi data. Peralihan data adalah diperlukan untuk menyokong mana-mana perubahan dalam struktur, perwakilan atau kandungan data. Penukaran ini menjadi perlu apabila kebanyakan situasi, contohnya untuk berhubung dengan skima evolusi, migrasi sistem pewarisan kepada satu sistem maklumat yang baru, atau apabila sumber data yang berganda di integrasikan.

Seperti ditunjukkan dalam **Rajah 2.1: Pengelasan Masalah Kualiti Data Dalam Sumber Data**, perbezaan secara kasar dibuat antara masalah sumber–tunggal dan sumber-berganda. Masalah tahap-skima juga ditunjukkan dalam tahap ketikaan. Masalah tahap ketikaan, pula merujuk pada ralat dan ketidakkonsistenan dalam kandungan data sebenar yang mana ianya tidak dilihat pada tahap skima. Ianya adalah fokus utama bagi pembersihan data. Rajah 2.2 juga menunjukkan masalah yang biasanya wujud dalam beberapa kes. Masalah sumber-tunggal juga wujud dalam kes masalah sumber-berganda, disamping masalah khusus yang wujud dalam masalah sumber-berganda itu sendiri.



Rajah 2.1: Pengelasan Masalah Kualiti Data Dalam Sumber Data

2.3.2.1 MASALAH SUMBER TUNGGAL

Kualiti data bagi sumber biasanya bergantung pada darjah kawalan oleh skima dan kewibawaan kekangan pengawalan nilai data yang dibenarkan. Bagi sumber tanpa skima, seperti fail, terdapat beberapa sekatan pada data yang akan dimasukkan dan disimpan, ini menyumbang pada peningkatan kemungkinan wujudnya ralat dan ketidakkonsitenan. Sistem pangkalan data, mewujudkan sekatan pada model data yang spesifik (contoh: pendekatan terhubung memerlukan nilai atribut yang mudah dan lain-lain), sama seperti aplikasi-kekangan kewibawaan spesifik. Masalah kualiti bagi skima-data terhubung wujud disebabkan kekurangan model-spesifik atau aplikasi-kekangan kewibawaan spesifik yang sesuai. Contohnya, disebabkan model data yang terhad atau reka bentuk skima yang lemah, atau disebabkan hanya beberapa kekangan kewibawaan ditafsirkan untuk menghadkan overhed bagi kawalan kewibawaan. Masalah sfesifik-ketikaan mempunyai hubungan dengan ralat dan ketidakkonsistenan yang tidak boleh dielakkan pada tahap skima (contoh: salah ejaan).

Skop/masalah		Data kotor	Sebab
Atribut	Nilai tidak sah	Bdate=30.13.70	Nilai adalah diluar julat domain
Rekod	Kebersandaran atribut yang tidak sah	umur=22,bdate=12.02.70	umur=(tarikh semasa-tarikh lahir)
Jenis rekod	Keunikan (*uniqueness*) yang tidak sah	emp_1=(nama="John Smith",NKS="123456") emp_2=(nama="Albert Tan",NKS="123456")	Keunikan bagi NKS (nombor keselamatan sosial) tidak sah
Sumber	Kewibawaan rujukan yang tidak sah	emp=(nama="John Smith", nojabatan=127)	Rujukan jabatan(127) tidak ditakrifkan

Jadual 2.0 Contoh Masalah Sumber-Tunggal Pada Tahap Skima (Kewibawaan Kekangan Yang Tidak Sah (*Violated Integrity Constraints*))

Bagi kedua-dua masalah tahap - skima dan tahap ketikaan boleh membezakan perbezaan skop masalah: atribut (medan), rekod, jenis rekod dan sumber. Didapati kekangan keunikan yang dinyatakan pada tahap skima tidak mencegah dari wujudnya lewahan data, contoh, sekiranya maklumat pada entiti sebenar yang sama dimasukkan dua kali dengan nilai atribut yang berbeza. (Rujuk **Jadual 2.1: Contoh Masalah Sumber-Tunggal Pada Tahap Ketikaan**).

Skop/masalah		Data kotor	Sebab
Atribut	Nilai hilang	Phone=999-9999	Data yang tidak wujud semasa kemasukan data (nilai nol)
	Salah ejaan	City="Kota Baru	Ralat fonetik
Jenis rekod	Nilai rahsia, Meringkaskan perkataan (*Abbreviation*)	pengalaman="B" pekerjaan="DB Prog"	
	Nilai tertanam (*embedded*)	Nama="J.smith 12.02.70 Kuala Lumpur"	pelbagai nilai dimasukkan kedalam satu atribut
	Salah nilai	bandar="Seremban"	
Rekod	Kebersandaran atribut yang tidak sah	bandar="Kelang",poskod=59100	Bandar dan kod poskod harus sepadan
Jenis rekod	Transformasi perkataan	$Nama_1$="J. Smith", $nama_2$="Miller P."	Biasanya dalam bidang(medan) bentuk bebas
	Lewahan rekod	emp_1=(nama="John Smith",..); emp_2=(nama=" John Smith",..);	Pekerja yang sama diwakilkan dua kali disebabkan ralat pada masukan data
	Percanggahan rekod	emp_1=(nama="John Smith", bdate=12.02.70); emp_2=(nama="John smith", bdate=12.12.70)	Nilai entiti sebenar yang sama diwakilkan dengan perwakilan berbeza
Sumber	Salah rujukan	emp=(nama="John smith", nojabatan=17)	

Jadual 2.1: Contoh Masalah Sumber-Tunggal Pada Tahap Ketikaan

2.3.2.2 MASALAH SUMBER BERGANDA

Masalah yang wujud dalam sumber-tunggal menjadi lebih buruk apabila data dari sumber - berganda perlu di integrasikan. Setiap sumber mungkin mengandungi data kotor dan data dari setiap sumber mungkin diwakilkan dalam bentuk yang berbeza, bertindih atau bercanggah. Ini adalah kerana sumber-sumber biasannya dibangunkan, dilaksanakan dan diselenggarakan tanpa kebersandaran bagi memenuhi keperluan yang spesifik. Ianya berlaku pada darjah yang besar bagi kepelbagaian sistem pengurusan data, model data, rekabentuk skima, dan data sebenar.

Pada tahap skima, perbezaan model data dan rekabentuk skima masing-masingnya dihubungkan oleh langkah-langkah terjemahan skima dan integrasi skima. Masalah utama dalam reka bentuk skima adalah penamaan dan konflik penstrukturan. Konflik penamaan meningkat apabila nama yang sama digunakan pada objek yang berlainan (homonim) atau nama yang berbeza digunakan pada objek yang sama (sinonim). Konflik penstrukturan muncul dalam kebanyakan variasi dan merujuk pada perwakilan yang berbeza bagi objek yang sama dalam sumber yang berlainan, contoh, perwakilan atribut vs jadual, struktur komponen yang berbeza, jenis data yang berbeza, kekangan kewibawaan yang berbeza, dan sebagainya.

Sebagai tambahan pada konflik tahap-skima, kebanyakan konflik muncul hanya pada tahap ketikaan (konflik data). Semua masalah dari kes sumber-tunggal boleh muncul dengan perwakilan yang berbeza dalam sumber yang berbeza (contoh: perulangan rekod, percanggaan rekod). Tambahan lagi, walaupun terdapat atribut dan jenis data yang sama, ianya mungkin mewakili nilai yang berbeza (contoh: bagi status

perkahwinan) atau penafsiran nilai yang berbeza (contoh: ukuran unit Dollar *vs* Euro) antara sumber-sumber. Seterusnya, maklumat dalam sumber mungkin disediakan pada tahap penjumlahan yang berbeza (contoh: jualan per produk *vs* jualan per kumpulan produk) atau merujuk pada titik masa yang berlainan (contoh: jualan semasa selepas satu hari bagi sumber 1 *vs* jualan semasa selepas satu minggu bagi sumber 2).

Masalah utama dalam pembersihan data dari sumber berganda adalah untuk mengenalpasti pertindihan data, terutamanya pemadanan rekod-rekod yang merujuk pada entiti dunia sebenar yang sama (contoh: pelanggan). Masalah ini juga dirujuk sebagai masalah identiti objek, penyingkiran perulangan atau masalah gabungan. Biasanya, hanya sebahagian maklumat akan bertindih dan setiap sumber mempunyai kelebihan tersendiri dengan menyediakan maklumat tambahan pada entiti masing-masing. Maklumat yang berulang ini perlu dicantum dan digabungkan bagi membentuk gambaran yang konsisten pada entiti dunia yang sebenar.[13][14]

Customer (sumber 1)

CID	Name	Street	City	Sex
11	Kristen Smith	2 Hurley P1	South fork, MN 48503	0
24	Brandon Smith	Hurley St 2	S Fork MN	1

Client (sumber 2)

Cno	LastName	FirstName	Gender	Address	Phone/Fax
24	Smith	Howard	M	23 Harley St, chicago IL,60633	333-222-6542/ 333-222-6599
493	Smith	Kris L.	F	2 Hurley Place Fork MN,48503	444-555-6666

Customer (Penyepaduan sasaran dengan data bersih)

N	Lname	Fname	Gender	Street	City	State	ZIP	Phone	Fax	CID	Cno
1	Smith	Kristen	F	2 Hurley	South	MN	4850	444-555-666		11	493
2	Smith	Brendon	M	2 Hurley	South	MN	4850			24	
3	Smith	Howard	M	23 Harley Street	Chicago	IL	6063	333-222-542	333-226-599		24

Jadual 2.2: Contoh Masalah Sumber-Berganda Pada Tahap Skima Dan Ketikaan

2.3.3 ANALISIS PENGENALPASTIAN DAN PENGELASAN RALAT YANG MUNGKIN WUJUD DALAM DATA PADA GUDANG DATA

Sumber data yang dibina secara bebas biasanya mempunyai skima yang berbeza. Malah, kebanyakan daripadanya mungkin menggunakan model data yang berbeza. Sebahagian dari tugas gudang data adalah untuk melaksanakan skima integrasi, dan menukarkan skima integrasi sebelum disimpan. Sebagai keputusan, data yang disimpan didalam gudang data bukan sahaja salinan bagi sumber, malah ianya boleh dianggap sebagai gambaran simpanan (gambaran hasil) data pada sumber.

Seperti yang dinyatakan dalam masalah yang wujud dalam sumber-tunggal dan sumber-berganda, ralat-ralat yang dikenalpasti bagi pembersihan data adalah dikelaskan seperti berikut:[18]

- **Tidak lengkap (*incomplete*)**
- **Tidak betul (*incorrect*)**
- **Tidak dapat difahami (*incomprehensible*)**
- **Tidak konsisten (*inconsistent*)**

2.3.3.1 RALAT~TIDAK LENGKAP (*INCOMPLETE*)

Ianya merangkumi:

- Kehilangan rekod

  Ini bermakna rekod yang sepatutnya tersimpan dalam sumber tidak wujud. Biasanya, ianya disebabkan oleh pengaturcara tidak membersihkan fail secara menyeluruh.

- Kehilangan medan

  Medan yang sepatutnya tersimpan dalam sumber tidak wujud. Biasanya, salah anggapan bahawa sistem sumber memerlukan kemasukan medan.

- Rekabentuk rekod atau medan tidak direkodkan

  Yang mana, kecuaian ketika rekabentuk, data yang perlu disimpan dalam gudang data tidak direkodkan. Terbahagi kepada 3 kategori. Pertama, mungkin terdapat dimensi jadual atribut yang akan direkodkan tetapi tidak terdapat dalam mana-mana sistem yang membekalkan gudang data. Sebagai contoh, pengguna pemasaran mungkin mempunyai skima pengelasan tersendiri bagi produk yang menunjukkan darjah bagi barangan yang mana telah dipromosikan. Kedua, sekiranya jenis data yang sama dibekalkan dari sistem yang pelbagai, didapati salah satu dari sistem sumber tidak merekod medan yang pengguna perlukan untuk disimpan dalam gudang data. Ketiga, mungkin terdapat 'transaksi' yang perlu disimpan dalam gudang data tidak disimpan dalam bentuk luaran.

2.3.3.2 RALAT~TIDAK BETUL (*INCORRECT*)

- Lewahan rekod

  Biasanya melibatkan dua situasi. Pertama, terdapatnya lewahan rekod dalam satu sistem yang membekalkan data pada gudang data. Kedua, terdapatnya maklumat yang berulang dalam sistem yang pelbagai yang membekalkan jenis maklumat yang sama. Sebagai contoh, mungkin data dibekalkan dari dua sistem iaitu sistem masukan tempahan bagi produk dan sistem masukan tempahan bagi perkhidmatan. Dari kedua-dua kes, perulangan mungkin tidak berlaku sekiranya data telah dikumpulkan atau dijumlahkan kedalam gudang data.

- Masukan maklumat yang salah kedalam sumber sistem

  Kadangkala sumber sistem mengandungi data dari masukan yang salah kedalam sistem. Sebagai contoh, salah masukan data 6/9/96 sebagai 9/6/96. Biasanya, langkah diambil dengan membetulkan sumber sistem. Walaubagaimanapun, kadangkala, atas sebab-sebab tertentu, sumber sistem tidak boleh diperbetulkan. Didapati, sekiranya terdapat banyak ralat dalam sumber sistem yang tidak boleh diperbetulkan, isu yang lebih besar adalah kebolehpercayaan 'rekod sistem'.

2.3.3.3 RALAT~TIDAK DAPAT DIFAHAMI (*INCOMPREHENSIBLE*)

Berikut merupakan jenis-jenis keadaan yang menyebabkan sumber data sukar dibaca.

- Lewahan medan dalam satu medan
  Situasi ini berlaku apabila sumber sistem mempunyai satu medan yang mengandungi maklumat yang mana gudang datanya menggunakan pelbagai medan/bidang. Sehingga kini, kemunculan yang biasa bagi masalah ini adalah apabila nama penuh, contoh: "Joe E. Brown", disimpan dalam satu medan dalam satu sistem sumber dan adalah perlu untuk menghuraikannya kepada tiga medan dalam gudang data.

- Pemformatan yang ganjil untuk memelihara ruang cakera
  Ianya berlaku apabila pengaturcara sistem sumber membuat isihan menggunakan skima luar biasa untuk menyimpan ruang cakera.

- Kod yang tidak dikenali
  Kebiasaannya, 99% maksud kod diketahui. Walaubagaimanapun, terdapat sebahagian rekod dengan kod yang tidak dikenali.

2.3.3.4 RALAT~TIDAK KONSISTEN (*INCONSISTENT*)

Masalah data tidak konsisten merangkumi sebahagian besar daripada masalah yang wujud. Biasanya data yang hampir sama dari sistem yang berbeza boleh menyebabkan ketidakkonsistenan. Walaubagaimanapun, data dalam satu sistem boleh tidak konsisten apabila merentasi lokasi, unit laporan, dan masa.

- Perbezaan kod yang tidak konsisten
  Kebanyakan literasi gudang data memberikan satu contoh sistem yang menggunakan "M' dan"F" dan satu sistem lain yang menggunakan "0' dan "1" untuk membezakan gender.

- Makna kod yang tidak konsisten
  Biasanya ianya menjadi isu apabila definisi entiti organisasi berubah mengikut masa. Sebagai contoh, katakan pada tahun 1995, terdapat pelanggan A,B,C, dan D. Pada 1996, pelanggan A membeli pelanggan B. Pada tahun 1997, pelanggan A membeli pelanggan C. Pada tahun 1998, pelangan A menjual sebahagian dari A dan C kepada pelanggan D. Apabila gudang data dibina pada tahun 1999, berdasarkan jenis analisis perniagaan yang dijalankan, dilema akan wujud tentang bagaimana untuk mengenalpasti jualan kepada pelanggan A,B,C dan D pada tahun sebelumnya.

- Kod yang bebeza dengan makna yang sama.
  Sebagai contoh, sesetengah rekod mungkin menyatakan warna ungu dan sesetengah pula menyatakan warna indigo. Pengguna gudang data mungkin mahu melihat warna ini sebagai satu perwakilan yang sama.

- Nama dan alamat yang tidak konsisten

- Peraturan perniagaan yang tidak konsisten

  Biasanya, kemungkinan untuk memuatkan pengiraan nombor kedalam gudang data adalah dielakkan, tetapi kadangkala wujudnya situasi yang mana ianya perlu dilakukan. Seperti yang telah dinyatakan, data perlu dibekalkan kedalam gudang data hanya untuk membuat semakan pengiraan. Ini juga bermakna hubungan bukan aritmetik antara dua medan tidak diikuti secara konsisten (contoh, sekiranya sebahagian nombor suffix adalah XXX, maka kod kategori adalah samada A,B, atau C)

- Pengumpulan/penjumlahan yang tidak konsisten
- Butiran maklumat yang tidak konsisten
- Pemasaan yang tidak konsiten
- Penggunaan atribut yang tidak konsisten

  Sebagai contoh, sistem masukan tempahan mungkin mempunyai medan berlabel arahan penghantaran. Medan mungkin mengandungi nama bagi agen pembelian bagi pelanggan, alamat e-mel pelanggan, dan lain-lain. Situasi yang lebih sukar adalah apabila polisi perniagaan digunakan untuk *populate* medan. Sebagai contoh,terdapat jadual dengan nombor akaun lejer. Didapati entiti A menggunakan akaun '1000' untuk perbelanjaan pentadbiran, sementara entiti B menggunakan '1500' untuk perbelanjaan pentadbiran.

- *Cut-off* tarikh yang tidak konsisten

  Berlaku apabila data disepadukan dari dua sistem yang diikuti oleh polisi yang berbeza untuk tarikh transaksi.

- Penggunaan nol, ruang, nilai kosong, dan lain-lain yang tidak konsisten

- Kewibawaan rujukan yang tidak konsisten

  Kebanyakan sumber sistem dibina tanpa semakan asas ini.

2.3.3.5 KONFLIK SKIMA

Konflik skima boleh meningkat disebabkan oleh perbezaan perwakilan objek dalam skima yang berbeza. 2 jenis konflik adalah konflik penamaan dan konflik penstrukturan.[17]

- Konflik penamaan : orang dari kawasan aplikasi berbeza bagi organisasi yang sama merujuk kepada data yang sama menggunakan terminologi dan nama yang berbeza. Ini akan menyebabkan kemungkinan ketidakkonsistenan dalam komponen skima. 2 jenis masalah yang mungkin wujud adalah: Homonim dan sinonim.

  Homonim muncul apabila nama yang sama digunakan pada dua konsep yang berbeza.

  Contoh : PERALATAN, pada satu skima merujuk kepada 'Komputer' sementara skima yang lain merujuk kepada 'Perabot'.

  Sinonim muncul apabila konsep yang sama digambarkan dengan dua nama atau lebih.

  Contoh : CLIENT vs CUSTOMER

  Homonin boleh dikesan dengan membandingkan konsep dengan nama yang sama dalam skima yang berbeza, sinonim hanya boleh dikesan selepas sfesifikasi luaran.

  Jenis homonim meningkat apabila konsep yang sama terdapatnya kesamaan pada nama tetapi berbeza set yang sepadan. Ianya boleh muncul pada tahap abstraks yang berbeza. Sebagai contoh, pada tahap atribut, saiz merujuk kepada saiz baju (kod integer tunggal) dalam satu skima, dan merujuk pada saiz seluar (interger berpasangan) pada skima

yang lain. Pada tahap entiti, PELAJAR merujuk kepada semua pelajar dalam pangkalan data yang disimpan di pejabat pendaftaran, dan merujuk pada pelajar yang telah berkahwin pada pangkalan data perkahwinan.

- Konflik penstrukturan : berlaku disebabkan pilihan berbeza bagi binaan pemodelan atau kekangan kewibaan. Pengelasan adalah seperti berikut:

    - Konflik Jenis

      Apabila konsep yang sama diwakilkan oleh binaan model yang berbeza dalam skima yang berbeza. Contohnya, satu kelas objek diwakilkan sebagai entiti pada satu skima dan atribut pada skima yang lain.

    - Konflik Kebersandaran

      Apabila sekumpulan konsep mempunyai hubungan antara satu sama lain dengan kebersandaran skima yang berbeza. Satu hubungan mungkin 1:1 (Pekerja Jabatan) pada satu skima dan m:n (Pekerja Jabatan Sejarah) pada skima yang lain.

    - Konflik Kunci

      Kunci yang berbeza diwakilkan pada konsep yang sama dalam skima yang berbeza. Contohnya, Pno an Pid mungkin merupakan kunci Pekerja dalam dua komponen skima.

2.4 KAEDAH PEMBERSIHAN DATA

Bahagian ini akan menunjukkan ralat-ralat yang boleh dibuang menggunakan peralatan. Secara umumnya, pembersihan data melibatkan beberapa fasa.

- Analisis data : untuk mengesan jenis ralat dan ketidakkonsistenan yang mana yang perlu di singkirkan, suatu analisis data yang lengkap diperlukan. Sebagai tambahan kepada penyemakan manual bagi data atau sampel data, program analisis perlu digunakan bagi mendapatkan metadata berkenaan dengan ciri-ciri data dan mengesan masalah kualiti data.
- Definisi aliran transformasi dan peraturan pemetaan : bergantung kepada bilangan sumber data, darjah perbezaan dan "kekotoran" data, transformasi dalam jumlah yang besar dan langkah pembersihan yang perlu dilaksanakan. Kadangkala, terjemahan skima digunakan untuk memetakan sumber kepada model data biasa; bagi gudang data, perwakilan berhubungan biasanya digunakan. Permulaan langkah pembersihan data boleh membetulkan masalah ketikaan sumber-tunggal dan menyediakan data untuk disepadukan. Langkah seterusnya akan berhubung dengan skima/ integrasi data dan pembersihan masalah ketikaan sumber-pelbagai, contohnya adalah lewahan data. Bagi gudang data, kawalan dan aliran data bagi transformasi dan langkah pembersihan ini perlu dientukan diantara aliran kerja yang menakrifkan proses ETL

(rujuk **Rajah 2.1: Langkah-Langkah Dalam Membina Gudang Data: Proses Pengekstrakan, Penyepaduan, dan Penyatuan**)

Skima-transformasi data terhubung seperti juga langkah pembersihan perlu ditentukan dengan pengisytiharaan pertanyaan dan bahasa pemetaan, bagi membolehkan penjanaan secara automatik terhadap kod transformasi. Sebagai tambahan, ianya mungkin boleh menghasilkan kod pembersihan tulisan-pengguna. Langkah transformasi mungkin memerlukan maklum balas pengguna pada data ketikaan yang mana ianya tidak mempunyai binaan dalaman bagi logik pembersihan.

- Pengesahan : ketepatan dan keberkesanan bagi aliran transformasi dan definisi transformasi perlu diuji dan dinilai, contoh, pada sampel atau salinan data sumber, meningkatkan definisi adalah suatu keperluan. Perulangan langkah–langkah bagi analisis, reka bentuk, dan pengesahan mungkin diperlukan, contoh, sesetengah ralat hanya kelihatan selepas proses transformasi.
- Transformasi : perlaksanaan langkah transformasi samada dengan melarikan aliran kerja bagi proses Pengekstrakan, Penyepaduan, dan Penyatuan(ETL) bagi memuatkan dan mengemaskini gudang data atau semasa menjawab pertanyaan bagi sumber pelbagai.

- Aliran balik data bersih : selepas ralat (sumber-tunggal) disingkirkan, data yang bersih perlu menggantikan data kotor dalam sumber sebenar bagi memberikan aplikasi pewarisan data yang telah tingkatkan dan juga untuk mengelakkkan kerja pembersihan berulang bagi pengekstrakan data pada masa hadapan.

2.4.1 ANALISIS DATA

Metadata yang ditunjukan dalam skima biasanya tidak mencukupi untuk mencapai kualiti data bagi sumber, terutamanya sekiranya hanya beberapa kekangan kewibawaan diwujudkan. Adalah penting untuk menganalisa ketikaan sebenar bagi mendapatkan metadata sebenar (kejuruteraan semula) pada ciri-ciri data atau corak nilai luar biasa. Metadata ini membantu dalam mencari masalah berkaitan dengan kualiti data. Tambahan lagi, ianya boleh menyumbang secara berkesan untuk mengenalpasti kesamaan pada atribut diantara skima sumber (persamaan skima), berdasarkan pada transformasi data automatik yang mana yang boleh diterbitkan.

Terdapat dua pendekatan yang berkaitan bagi analisis data, iaitu profil data dan perlombongan data. Profil data memfokuskan pada analisis ketikaan bagi atribut individu. Ianya menerbitkan maklumat seperti jenis data, panjang, nilai julat, nilai diskrit dan frekuensi, perbezaan, keunikan, kejadian nilai nol,

corak rentetan yang biasa (contoh, nombor telefon), dan sebagainya menyediakan gambaran sebenar bagi beberapa aspek kualiti bagi atribut.

Jadual 2.3: Contoh Kegunaan Kejuruteraan Semula Metadata Bagi Menunjukkan Masalah Kualiti Data menunjukkan bagaimana metadata membantu dalam mengesan masalah kualiti data.[14]

Masalah	Metadata	Contoh/Heuristik
Nilai tidak sah	kekardinalan	contoh, kekardinalan (gender)>2 menandakan masalah
	Max, min	max, min tidak berada diluar julat yang dibenarkan
	perbezaan, lencongan	perbezaan, lencongan nilai statistik tidak melebihi sempadan (*threshold*)
Salah ejaan	Nilai atribut	Isihan pada nilai biasanya membawa kepada nilai yang salah bersebelahan dengan nilai yang betul
Nilai hilang	Nilai nol	peratus/nombor bagi nilai nol
	nilai atribut + nilai lalai	kehadiran nilai lalai mungkin menunjukkan nilai sebenar hilang
Perbezaan perwakilan nilai	Nilai atribut	perbandingan set nilai atribut bagi kolum satu jadual dengan yang lain
Lewahan	kekardinalan/keunikan	kekardinalan atribut=# kandungan pada baris
	Nilai atribut	isihan nilai dengan bilangan kemunculan; muncul lebih dari sekali menunjukkan perulangan

Jadual 2.3: Contoh Kegunaan Kejuruteraan Semula Metadata Bagi Menunjukkan Masalah Kualiti Data

Antara kategori ralat yang difokuskan dalam bab ini adalah

i) Data yang tidak konsisten

ii) Integrasi skima

2.4.1.1 DATA YANG TIDAK KONSISTEN

Jadual 2.2: Contoh Masalah Sumber-Berganda Pada Tahap Skima Dan Ketikaan menunjukkan data bersih yang disasarkan yang terhasil dari penyepaduan data dari sumber berganda.

Pada tahap skima, terdapat konflik nama (sinonim *Customer/client*, *CID/Cno*, *Sex/Gender*) dan konflik penstrukturan (perwakilan yang berbeza bagi nama dan alamat).

Pada tahap ketikaan, terdapat perwakilan gender yang berbeza ("0"/"1" vs "F"/"M") dan dengan anggapan lewahan rekod (Kristen Smith). Pemerhatian seterusnya turut mendedahkan walaupun *CID/Cno* adalah sumber-pengenalpasti spesifik, tetapi kandungan kedua sumber tidak dapat dibandingkan; nombor yang berbeza (11/493) boleh merujuk pada orang yang sama sementara orang yang berlainan boleh dirujuk dengan nombor yang sama (24). Penyelesaian pada masalah ini memerlukan integrasi antara kedua skima dan pembersihan data; penyelesaian yang mungkin ditunjukkan dalam **Jadual 2.2: Contoh Masalah Sumber-Berganda Pada Tahap Skima Dan Ketikaan**. Didapati konflik skima perlu diselesaikan terlebih dahulu bagi membenarkan pembersihan data, terutamanya pengesanan pada perulangan berdasarkan kepada bentuk perwakilan nama dan alamat, dan penyamaan nilai *Gender/Sex*.

Proses transformasi data biasanya memerlukan jumlah metada data yang besar seperti skima, tahap ketikaan, ciri-ciri data, pemetaan trasformasi, definisi aliran kerja dan sebagainya. Bagi keselarasan, kefleksibelan, dan kemudahan untuk diguna semula, metadata perlu dikekalkan di dalam DBMS. Bagi menyokong kualiti kualiti data, maklumat terpeinci mengenai proses transformasi perlu direkodkan, iaitu pada kedua gudang data dan transformasi ketikaan.

Suatu pendekatan yang lebih umum dan fleksibel menggunakan Bahasa Pertanyaan Berstruktur (SQL) untuk melaksanakan tranformasi data dan untuk memungkinkan penggunaan aplikasi-spesifik bahasa terluas, yang mana fungsian takrifan penguna (*User defined Functions* (UDFs)) disokong dalam SQL:99. UDFs boleh dilaksanakan dalam SQL atau bahasa pengaturcaraan tujuan-umum dengan penyatuan penyataan SQL. Ianya membenarkan pelaksanaan yang lebih meluas bagi data transformasi dan menyokong kemudahan guna semula untuk transformasi berbeza dan tugas pemprosesan pertanyaan. Tambahan lagi, pelaksanaannya melalui DBMS boleh mengurangkan kos capaian data dan meningkatkan persembahan.

```
CREATE VIEW    Customer2 (Lname,Fname,Gender,Street,City,State,ZIP,CID) AS
SELECT         LastNameExtract(Name), FirstNameExtract(Names), Sex,
               Street, CityExtract(City), StateExtract(City),
               ZIPExtract(City), CID
FROM           Customer
```

Rajah 2.2: Contoh Definisi Langkah Transformasi

Rajah 2.2: **Contoh Definisi Langkah Transformasi** menunjukkan langkah transformasi yang dinyatakan dalam SQL:99. contoh merujuk kepada **Jadual 2.2: Contoh Masalah Sumber-Berganda Pada Tahap Skima Dan Ketikaan** dan meliputi sebahagian daripada data trasformasi yang diperlukan untuk diaplikasikan kepada sumber pertama. Transfomasi ini menakrifkan gambaran pada pemetaan yang mana yang boleh dilaksanakan seterusnya. Transformasi membentuk penstrukturan skima dengan tambahan atribut dalam gambaran yang diperolehi dengan mengasingkan atribut nama dan alamat bagi sumber. Pengekstrakan data yang diperlukan dapat dicapai dengan UDFs (ditunjukan dengan **boldface**). Implementasi UDF boleh mengandungi logik pembersihan, contohnya untuk menyingkirkan salah ejaan pada nama bandar atau menyediakan kod zip yang hilang.[14]

UDF mungkin membayangkan usaha implementasi yang tinggi dan tidak menyokong kesemua keperluan skima transformasi. Terutamanya, fungsian yang mudah dan kerap seperti pengasingan atribut atau percantuman adalah tidak disokong secara umumnya tetapi perlu selalu diimplementasikan semula dalam aplikasi-variasi spesifik. Penstrukturan skima yang lebih kompleks (contoh, atribut *folding* dan *unfolding*) tidak disokong langsung. Untuk sokongan umum transformasi skima-terhubung, bahasa terluas seperti cadangan SchemaSQL adalah diperlukan.[9][10]

Hasil daripada pemerhatian yang dibuat, berikut(**Jadual 2.4: Penghuraian Konflik**) merupakan ringkasan kesimpulan bagi penyelesaian ralat sumber berganda.[13]

Jadual 2.4: Penghuraian Konflik

1. perbezaan jenis objek yang sepadan
   - -nama : hamonim dan sinonim
   - -kunci: atribut berbeza
   - -atribut: set atribut berbeza
     konflik atribut yang sepadan
   - -kaedah: set kaedah yang berbeza
     konflik kaedah yang sepadan
   - -kekangan kewibawaan
   - -kebenaran capaian

2. perbezaan atribut yang sepadan
   - -nama : hamonim dan sinonim
   - -skop: jenis tempatan bagi satu pangkalan data dan jenis global pada pangkalan data yang lain
   - -struktur : -mudah, kompleks
     kekardinalan : nilai mono dan nilai berganda, pilihan dan mandotori
   - -jenis data : skala, unit, nilai lalai, operasi berbeza
   - -kekangan kewibawaan
   - -kebenaran capaian

3. perbezaan kaedah
   - -nama : hamonim dan sinonim
   - -pengenalan : berbeza set parameter, jenis parameter, keputusan

Jadual 2.5: Penyelesaian Bagi Konflik

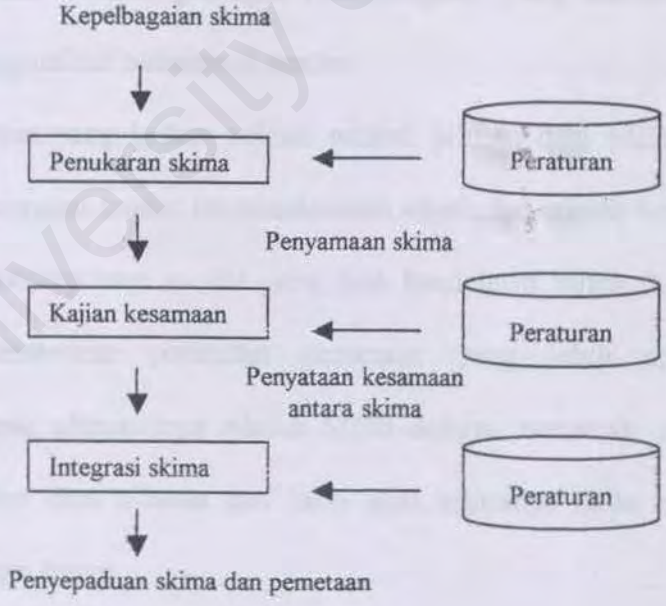
Konflik	Penyelesaian
Nama	
Hamonim	Menambahkan imbuhan pada nama
Sinonim	Setkan satu nama
Kunci berbeza	Menggunakan fungsi penukaran atau menggunakan jadual pemadanan
Set atribut berbeza	Satukan set atribut(menggunakan pendekatan-atribut mempunyai takrifan yang hampir sama disatukan)
Konflik atribut yang Sepadan	Menggunakan fungsi penukaran atau menggunakan jadual pemadanan

2.4.1.2 INTEGRASI SKIMA

Apa yang difokuskan disini merupakan konsep asas dan altenatif bagi mengatasi ralat yang dihadapi. Antara langkah-langkah yang terlibat didalam pembangunan integrasi [13]:

- pra-integrasi yang mana skima input diubah bagi menjadikannya sama jenis dan bentuk (kedua-duanya dari segi sintaksis dan semantik)
- identifikasi kesamaan, tumpuan kepada identifikasi dan penghuraian hubungan antara skima
- integrasi, langkah akhir yang menyelesaikan konfli antara skima dan menyatukan item yang sama di dalam skima yang disepadukan

Rajah 2.3: Proses Integrasi Global



2.4.1.2.1 LANGKAH PRA INTEGRASI

Menerbitkan satu kefahaman umum bagi data yang telah wujud merupakan pra keperluan bagi kejayaan integrasi pangkalan data. Kebanyakan metodologi peraturan integrasi juga mencadangkan bagi mendapat kejayaan pengubahsuaian skima input, adalah dengan menggunakan set bagi operator-operator. [12]

Penyelidik dalam integrasi pangkalan data biasanya menganggap skima input dinyatakan dalam model data yang sama, yang biasanya dikenali model data biasa (MDB). Langkah penterjemahan menjadi pra-keperluan untuk penyepaduan dan ditangani sebagai masalah yang berbeza. Malangnya, model penterjemahan data semasa adalah kurang dari segi peralatan bagi perterjemahan automatik. Pembangunan kini memfokuskan perterjemahan di antara model berorientasikan objek dan model hubungan. Sesetengah penyelidik juga menekankan perterjemahan operasi, yang mana diperlukan bagi sistem bahasa pelbagai (multilingual) yang membenarkan setiap pasangan untuk menggunakan bahasanya sendiri.

Satu perdebatan yang belum selesai adalah pilihan bagi MDB. Kebanyakan penyelidik lebih menyukai model berorientasikan objek. Ini adalah kerana model ini mempunyai semua konsep bagi model yang lain kaedahnya boleh digunakan untuk untuk mengimplimentasikan peraturan pemetaan yang lebih spesifik. Untuk memudahkan integrasi, altenatifnya adalah MDB dengan semantik yang minimum, yang mana perwakilan data dibawa dari fakta asas sekiranya tiada model altenatif, seperti model hubungan binari.

2.4.1.2.2 LANGKAH PENGENALPASTIAN PERSAMAAN

Langkah seterusnya adalah pengenalpastian persamaan. Kandungan pangkalan data mewakilkan fakta dunia sebenar (contoh: objek dan hubungan). Sebelum pangkalan data di integrasikan, penekanan adalah kepada apa yang diwakilkan oleh pangkalan data tersebut berbanding dengan bagaimana pangkalan data tersebut diwakilkan. Oleh itu, dua pangkalan data dikatakan mempunyai suatu persamaan sekiranya fakta dunia sebenar yang diwakilkan oleh pangkalan data-pangkalan data tersebut mempunyai elemen-elemen yang sama. Satu kaedah yang boleh digunakan bagi skima integrasi adalah dengan menggunakan kamus data dan nilai domain sebagai panduan integrasi.

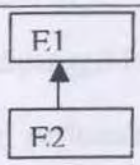
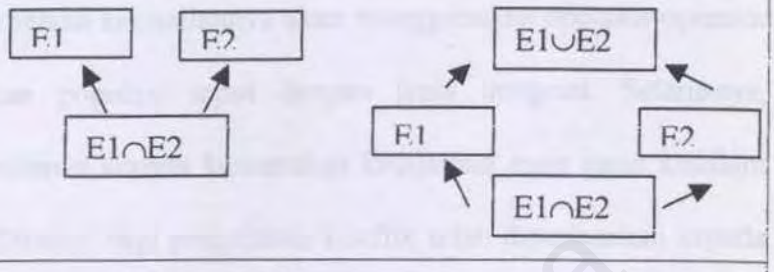
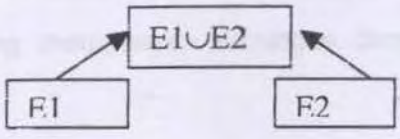
Bagi membina skima integrasi, hubungan dalam dunia sebenar bagi elemen-elemen hubungan tambahan perlu diketahui. Jenis hubungan tambahan dilihat sebagai set-set elemen dunia sebenar. Setiap penyataan persamaan antara skima menentukan set hubungan biasa adalah : sama ($\equiv$), persilangan ($\cap$),bezaan ($\neq$), subset($\supseteq$).

Apabila persamaan telah dikenalpasti, proses integrasi kemudiannya sampai ke satu titik yang mana integrasi sebenar dibuat. Setiap penyataan persamaan antara skima di analisa bagi menentukan perwakilan yang mana diantara elemen-elemen terhubung perlu dimasukkan dalam skima integrasi dan untuk menentukan pemetaan di antara skima integrasi dengan skima input. Integrasi mungkin dibuat secara manual oleh Pentadbir Pangkalan Data, dengan menggunakan beberapa prosedur[12], pengisytiharaan atau bahasa manipulasi logikal. Sistem membantu Pentadbir Pangkalan Data dengan menjana pemetaaan. Apabila penyataan persamaan antara skima

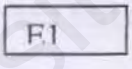
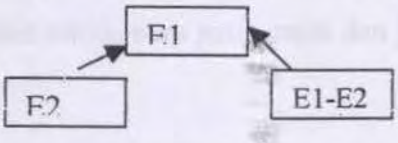
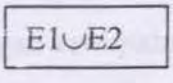
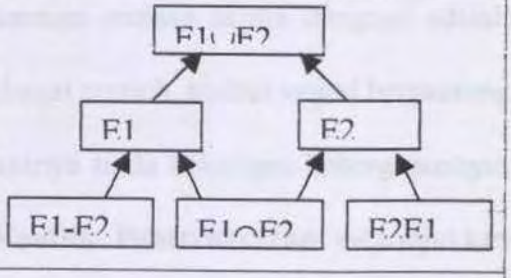
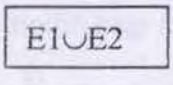
menjelaskan jenis-jenis yang sepadan adalah serupa (sama perwakilan dan sama hubungan tambahan), integrasi antara skima tersebut adalah secara terus. Walaubagaimanapun, biasanya jenis-jenis yang sepadan akan menunjukkan perbezaan dalam perwakilan dan hubungan tambahan.

Kebanyakkan dari kertas cadangan membuat cadangan yang melibatkan penyelesaian konflik bagi pengelasan dan penghuraian, dan hanya sedikit yang membuat cadangan menggunakan konflik metadata, perstrukturan, kepelbagaian, dan konflik data. Tiada kaedah integrasi adalah lengkap. Matlamat integrasi yang tepat biasanya sukar dilakukan. Malah kebanyakannya memberi penekanan yang lebih kepada ringkasan dan kebolehbacaan yang menghasilkan bilangan kelas yang minimum.

Konflik pengelasan meningkat apabila jenis-jenis data yang sepadan menghuraikan set-set yang berbeza bagi elemen dunia sebenar. Sebagai contoh, Pengarang bagi Pangkalan Data 1 merujuk kepada Pengarang bagi Jurnal dan Kertas persidangan, manakala Pengarang bagi Pangkalan Data 2 merujuk kepada Pengarang bagi Jurnal sahaja. Konflik pengelasan dilaksanakan dalam penyataan persamaan antara skima menggunakan set hubungan $\cap$, $\neq$ atau $\supseteq$[12]. Bagi mencapai sasaran skima integrasi, satu penyelesaian piawai bagi konflik-konflik yang wujud semasa mengintegrasikan skima-skima input dibina dalam bentuk hirarki umum yang bersesuaian (seperti ditunjukkan dalam **Jadual 2.6: Penyelesaian Piawai : Tidah Mengubah Jenis Tempatan**). Pemetaan diantara skima integrasi dan pangkalan data input mengurang pada fungsi identiti.

Konflik	Integrasi skima
$E1 \supseteq E2$	
$E1 \cap E2$	
$E1 \neq E2$	

Jadual 2.6: Penyelesaian Piawai : Tidak Mengubah Jenis Tempatan

Konflik	Integrasi skima	
	Teknik Gabungan	Teknik Menyeluruh
$E1 \supseteq E2$		
$E1 \cap E2$		
$E1 \neq E2$		

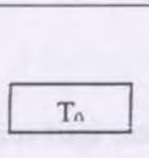
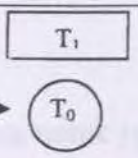
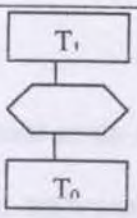
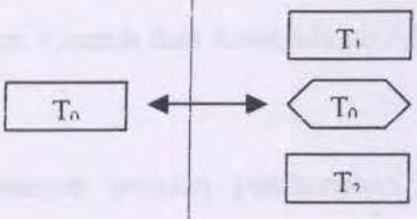
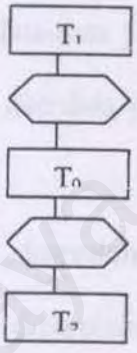
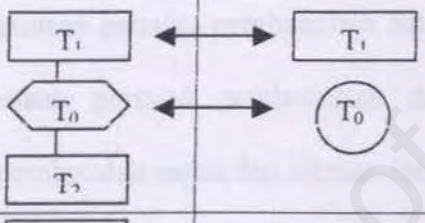
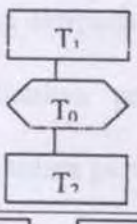
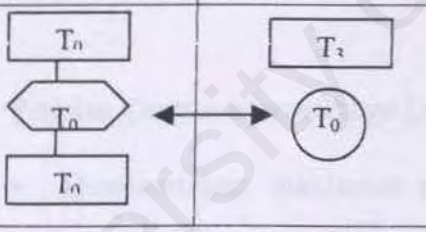
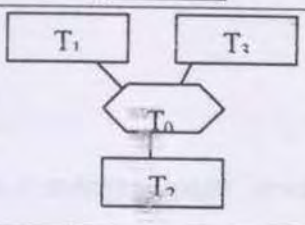
Jadual 2.7: Penyelesaian Altenatif

Jadual 2.6: Penyelesaian Piawai : Tidah Mengubah Jenis Tempatan dan **Jadual 2.7: Penyelesaian Altenatif** merujuk kepada pengelasan penyelesaian konflik. Sebagai altenatif, satu prinsip mudah dipanggil ke dalam integrasi skima bagi jenis yang unik yang menerangkan kesatuan bagi perluasan (teknik gabungan dalam **Jadual 2.7: Penyelesaian Altenatif**). Pemetaan kemudiannya akan menggunakan operator-operator pilihan bagi menghubungkan populasi input dengan jenis integrasi. Seterusnya, keseluruhan prinsip ini membawa kepada kemasukan kedua-dua jenis input kedalam skima yang telah disepadu. Strategi bagi pengelasan konflik telah diperluaskan kepada integrasi hirarki umum yang mempunyai hubungan dengan penyataan persamaan antara skima yang pelbagai.

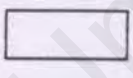
Hirarki-hirarki adalah digabungkan dengan mengambil setiap kelas dari satu hirarki dan kesepadanan kelas-kelas ini dengan hirarki yang lain ditentukan. Penempatan kelas-kelas yang sepadan ini mengambil kira semantik bagi kelas.

Konflik penstrukturan meningkat apabila jenis-jenis data yang sepadan dihuraikan dengan perwakilan yang berbeza: objek dan entiti, atau jenis entiti dan jenis hubungan di antara entiti.

Kekangan-kekangan yang biasanya ditekankan semasa skima integrasi adalah kekardinalan dan kebergantungan kewujudan. Sebagai contoh, atribut wujud bergantung kepada pemiliknya (objek), manakala objek biasanya tiada kekangan kebergantungan kewujudan. **Rajah 2.4: Penyelesaian Bagi Konflik Penstrukturan** menunjukkan binaan integrasi yang mungkin dipilih bagi konflik-konflik penstrukturan.

Kes	Skima 1	Skima 2	Skima Integasi
1.			
2.			
3.			
4.			

Gambarajah ini adalah dalam model data umum yang mana :



Mewakili jenis objek (kelas objek, jenis entiti, hubungan, atau jenis rekod.

Mewakili hubungan diantara jenis objek (atribut, jenis hubungan)

Mewakili atribut

Rajah 2.4: Penyelesaian Bagi Konflik Penstrukturan

2.5 KAJIAN LITERASI PADA PERISIAN PERMBERSIHAN DATA SEDIA ADA

Kini perisian bagi integrasi data untuk jumlah data yang besar mula mendapat permintaan yang tinggi. Di samping menghadapi masalah integrasi tradisional seperti integrasi skima dan pemetaan skima tempatan ke skima global, data-data yang kotor juga perlu ditangani. Contoh data kotor adalah ralat masukan data dan data yang tidak konsisten.

Terdapat banyak perisian pembersihan pangkalan data diperolehi dengan melayari Internet. Pelbagai maklumat baru dan berguna boleh diperolehi melalui demonstrasi penggunaan perisian pembersihan data ini. Kebanyakan daripada web-web yang mempromosikan perisian pembersihan data adalah melibatkan pembaikan, pemiawaian, dan pembetulan nama dan alamat. Berikut merupakan antara perisian yang telah diperolehi:

1) Perisian Centrus Merge/Purge Library [1]
   - Membersihkan maklumat pelanggan : mengenalpasti lewahan rekod
   - Fasa pemadanan algoritma
     i) kunci indeks berganda ditakrifkan berdasarkan satu medan atau berganda medan (dengan menggunakan fungsi transformasi)
     ii) rekod-rekod dipadankan beberapa kali berdasarkan satu kunci setiap kali pemadanan,

hanya rekod yang mempunyai kunci yang sama di bandingkan

- Pasangan kunci yang ditemui kemudiannya dihubungkan antara satu sama lain bagi membentuk kumpulan lewahan
- Kriteria pemadanan medan : apakah huraian pasangan medan dan jenis medan yang dibandingkan
- Algoritma pemadanan medan berganda menggunakan perbandingan yang sama : *soundex*, jarak papan kekunci, jarak penyuntingan, perbandingan rentetan, perbandingan penomboran ciri-ciri frekuensi algoritma
- Membenarkan penggunaan takrifan pengguna mengenai algoritma pemadanan
- Ketika penyepaduan data, keutamaan boleh disetkan bagi setiap senarai rekod
- Kemungkinan untuk mendapat jadual output berganda dengan : rekod-rekod yang unik, rekod-rekod yang terbaik (dimiliki oleh senarai yang mempunyai keutamaan yang paling tinggi), lewahan dimiliki oleh beberapa senarai dan penapisan.
- Capaian terus pada Oracle dan DBMS yang lain
- Ditulis dalam pengaturcaraan C dan C++

2) Perisian Classification Based on Associations(CBA) [2]

- Merupakan peralatan perlombongan data yang dibangunkan oleh Universiti Kebangsaan Singapura
- Algoritma yang digunakan oleh perisian ini dipersembahkan sebagai kertas cadangan "*Integrating Classification and Association Rule Mining* " di Persidangan Antarabangsa ke–4 yang bertajuk *Knowledge Discovery and Data Mining* (KDD 98), di New York City, USA.
- Antara ciri-ciri unik sistem ini dalam bahagian pembersihan data adalah
    - Menggunakan format fail data yang berbeza bagi jenis perlombongan yang berbeza
    - Penukaran dan pembersihan data menggunakan jadual pengelasan dan gabungan
    - Semasa proses pembersihan data, sistem akan menyemak kekonsistenan data, mengesan ralat dan seterusnya menghapuskan ralat.
- Antara ralat-ralat yang boleh dibersihkan adalah
    - Bilangan atribut dalam rekod data adalah berbeza daripada bilangan atribut nama fail yang disenaraikan
    - Nilai bukan penomboran muncul dalam atribut berturutan
    - Kelas atribut mempunyai huraian 'berterusan' dalam fail nama

- Terdapat kata simpan dalam nama dan fail data. Ianya perlu digantikan dengan bukan kata simpan. Set kata simpan adalah : .(){}:,=><+-
- Pengguna dapat melihat langkah-langkah pembersihan data

3) Perisian matchIT [3]

- Menggunakan antaramuka yang boleh diklik
- Ditulis dalam bahasa pengaturcaraan C++ dan Visual Fox Pro
- Membetulkan alamat berdasarkan Fail Alamat Pos Surat (*Royal Mail's Postal Address File*)
- Pengguna menyenarai kunci yang sepadan (kombinasi fungsi-fungsi diaplikasikan pada medan seperti kunci fonetik dan sebagainya)
- Hanya rekod yang mempunyai kunci yang sama dibandingkan; semakan yang berbeza bagi kunci yang berbeza
- Membenarkan pengguna menakrifkan medan yang akan dibandingkan kepentingannya melalui matriks
- Menjana markah pemadanan (perbandingan medan ke medan) bagi setiap pasangan rekod yang dibandingkan
- Lewahan data yang mempunyai kunci yang sama dikumpulkan bersama

- Penyepaduan dilakukan pada paras rekod
- Diaplikasikan pada satu atau dua jadual

4) Perisian StyleList and Personator [4]
   - Mengasingkan nama, alamat, bandar dan poskod
   - Menyokong beberapa kod pemadanan : ketepatan, fonetik, padanan yang hampir sama dan lain-lain
   - Menyokong kod pemadanan yang ditakrifkan oleh pengguna
   - Beberapa ciri-ciri pemadanan digunakan serentak, contoh poskod + nama keluarga + nama jalan
   - Pilihan rekod antara lewahan yang ditemui : dengan fail keutamaan, kandungan medan, pilihan rawak, atau mengambil nombor yang sama bagi setiap sumber yang dibandingkan
   - Diaplikasikan lebih dari satu sumber fail
   - ODBC membuat capaian pada data
   - Corak senarai : format (bingkai, perwakilan kependekan bagi perkataan, pembubuhan tanda baca) nama dan alamat

5) Perisian Data Tools twins [5]

- Menyediakan huraian bagi alamat, negara, poskod dan nama individu
- Menggunakan pangkalan data PAF sebagai jadual panduan bagi mempiawaikan alamat
- Pra-takrifan teknik pemadanan : mencari lewahan perseorangan (orang yang sama/ alamat yang sama), lewahan perniagaan (perniagaan yang sama/ alamat yang sama), pemadanan telefon, nombor rujukan, pemadanan *e-mail*.
- Membenarkan penyuntingan lewahan data yang dijumpai seperti potong, salin dan tampal secara manual bagi mengasingkan rekod
- Kemungkinan untuk menakrifkan matriks dan ciri-ciri kualiti bagi membolehkan secara automatik hanya satu rekod dari setiap lewahan kumpulan dibentuk
- Membenarkan pembersihan data pada satu atau dua jadual
- ODBC dihubungkan ke DBMS

BAB 3 METODOLOGI

3.0 PENGENALAN

Metodologi dapat ditakrifkan sebagai koleksi prosedur, teknik-teknik, peralatan dan dokumentasi. Metodologi dapat membantu pembangun perisian untuk memepercepatkan dan mempermudahkan proses pembangunan perisian. Dengan adanya metodologi, ia membantu untuk merancang, mengurus, mengawal dan membuat penilaian terhadap projek sistem maklumat. Terdapat beberapa metodologi dalam pembangunan sistem di mana setiap satunya mempunyai objektif, kelebihan dan kekurangan tersendiri bergantung kepada jenis sistem yang akan dibangunkan. [7]

Metodologi dijalankan bagi memastikan keperluan sebenar sistem yang akan dibangunkan akan lebih mudah dilaksanakan dan menjimatkan masa.

3.1 KAJIAN MODEL – METODOLOGI

3.1.1 METODOLOGI PEMBANGUNAN SISTEM

Dalam pembangunan sistem, pemilihan metodologi yang bersesuaian adalah perlu sebagai panduan dalam menghasilkan sebuah system dengan memenuhi kesemua keperluan yang telah dijangkakan.

Untuk membangunkan **dCleanViewer**, metodologi yang digunakan ialah Model Air Terjun dengan Prototaip.

Bagi model ini, dalam setiap peringkat, pembangun sistem boleh kembali kepada peringkat sebelumnya sekiranya terdapat kesalahan. Tetapi sekiranya kesalahan yang terdapat pada sistem lambat dikesan, kos yang mahal dipelukan untuk membaikinya. [7]

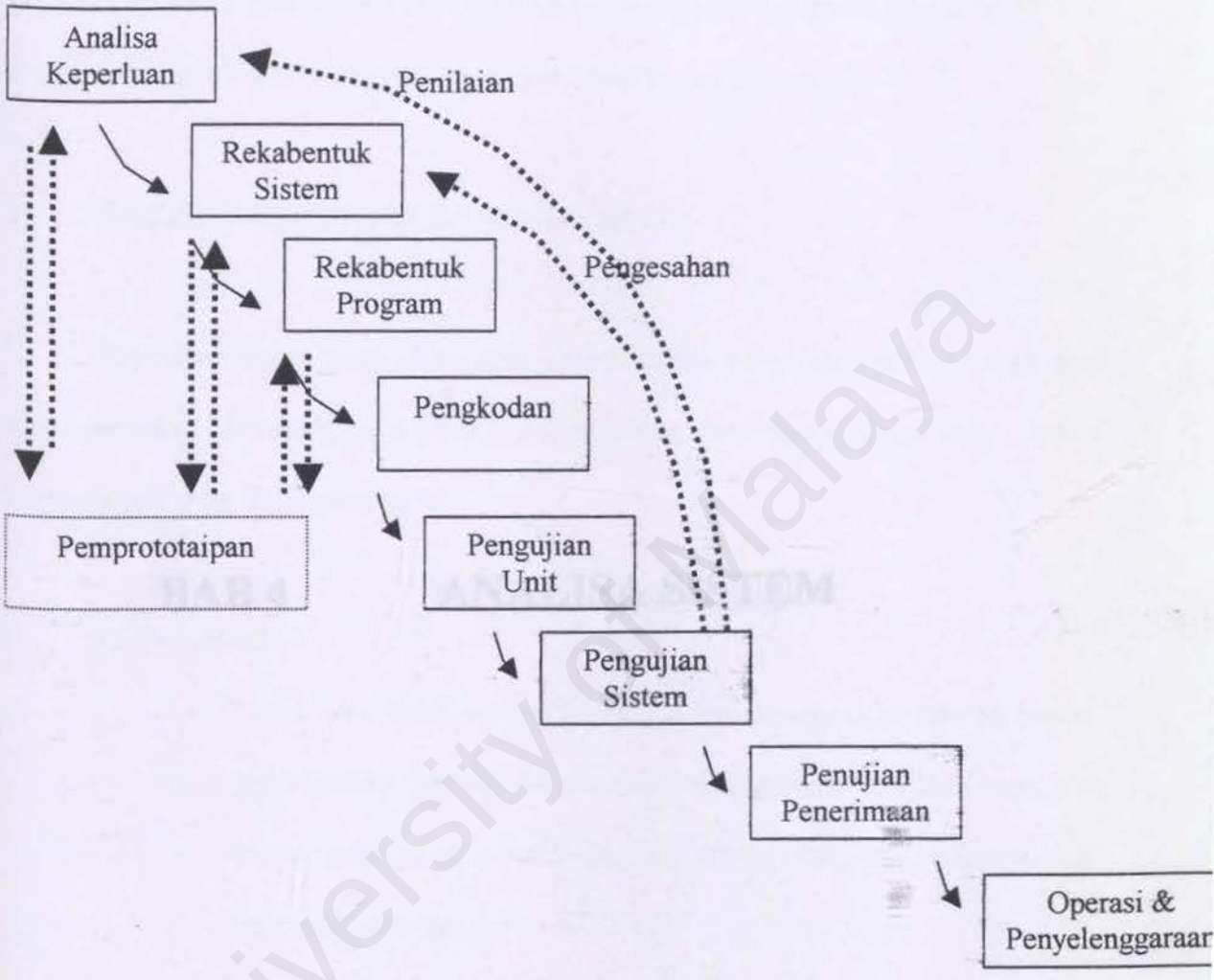
3.1.2 KELEBIHAN MENGGUNAKAN PROTOTAIP DALAM PEMBANGUNAN SISTEM

Dengan menggunakan prototaip, masa yang dihasilkan antara penentuan keperluan maklumat dan perlaksanaan sistem dapat dipendekkan. Ianya juga dapat mengatasi masalah seperti tidak memenuhi keperluan pengguna, dengan mengenalpasti keperluan maklumat pengguna dengan tepat. Pengguna juga boleh melihat apa yang mungkin dan bagaimana keperluan yang di kehendaki dialih ke dalam bentuk perkakasan atau perisian. Selain itu, prototaip juga berpotensi untuk mengubah suai sistem diperingkat awal pembangunan di mana ia berkesempatan untuk menghentikan pembangunan ke atas sistem yang tidak diperlukan. Sistem yang direkabentuk juga dapat memenuhi keperluan dan jangkaan pengguna.

3.1.3 SEBAB MODEL AIR TERJUN DENGAN PROTOTAIP DIPILIH

Model Air Terjun dengan Prototaip dipilih dalam membangunkan **dCleanViewer** ini kerana ia dapat memberi lebih pemahaman kepada pembangun sistem mengenai aktiviti yang sebenarnya berlaku dalam pembangunan sistem. Selain itu, terdapat beberapa proses dalam fasa pembangunan projek yang cukup sekadar ditunjukkan dengan model Air Terjun, tapi terdapat juga sebilangan yang tidak jelas dan harus ditunjukkan dengan prototaip.

Rajah 3.0 Model Air Terjun dengan Prototaip [7]



BAB 4 ANALISA SISTEM

4.0 ANALISA SISTEM

Analisa sistem adalah teknik penyelesaian masalah yang membahagikan sistem kepada komponen-komponen kecil untuk tujuan mengkaji bagaimana satu-satu komponen bekerja dan berinteraksi bagi mencapai tujuan sebenar. [15]

4.1 TEKNIK PENGUMPULAN MAKLUMAT

Beberapa teknik telah dilakukan dalam usaha mendapatkan maklumat bagi membangunkan **dCleanViewer** bagi perlombongan data dan gudang data. Antara teknik-teknik yang digunakan ialah :

a) Penyelidikan

- Untuk mendapatkan maklumat-maklumat yang tepat dan berkualiti, penyelidikan dan kajian dilakukan menggunakan majalah, buku dan kertas kerja yang membincangkan isu-isu berkaitan dengan sistem yang akan dibangunkan.

b) Melayari Internet

- Perbincangan dan rujukan secara bertulis mengenai projek ini adalah terhad kerana ia merupakan suatu bidang baru. Internet merupakan media yang paling efektif bagi mendapatkan maklumat dan kajian mengenai pembersihan data dalam perlombongan data. Kajian-kajian

mengenai sistem sedia ada diperolehi daripada laman-laman web tertentu yang mempromosikan perisian dan membicarakan isu-isu semasa berkaitan sistem seperti ini. Selain itu, banyak kertas kerja yang membincangkan mengenai tajuk-tajuk yang berkaitan juga boleh didapati di internet.

c) Perbincangan dengan Penyelia

- Perbincangan dengan penyelia adalah penting kerana dengan perbincangan ini, skop dan keperluan mengenai sistem yang akan dibangunkan dapat dikenalpasti.

4.2 KEPERLUAN SISTEM

Keperluan adalah suatu ciri sistem atau penerangan tentang sesuatu yang boleh dilakukan oleh sistem bagi memenuhi tujuan sistem tersebut. Terdapat dua jenis keperluan iaitu keperluan fungsian dan keperluan bukan fungsian. [7]

4.2.1 KEPERLUAN FUNGSIAN

Keperluan fungsian menyatakan tentang fungsi-fungsi yang ditawarkan oleh sistem iaitu bagaimana sistem bertindakbalas terhadap sesuatu input dan juga cara kelakuan sistem dalam keadaan-keadaan tertentu. [7]

Keperluan fungsian bagi (**dCleanViewer**) :

a) Menu *File*

- Menu *File* ini disetkan dengan fungsian piawai seperti aplikasi windows yang lain. Sub menu yang terdapat dalam menu *File* adalah *Exit*. Fungsi ini membenarkan pengguna keluar daripada sistem.

b) Menu *Settings*

- Menu *Settings* akan mengandungi sub menu seperti *Database Type*, *NT Integrated Security*, dan *Recordset Type*. Menu *Database Type* membolehkan pengguna memilih jenis pangkalan data yang ingin dicapai samada Microsoft Access atau SQL Server. *NT Integrated Security* pula berfungsi sekiranya pengguna memilih untuk membuat capaian pada pangkalan data SQL Server. Ianya membolehkan pengguna memilih untuk mencapai pangkalan data SQL Server menggunakan capaian *default* NT *domain account* ataupun sebaliknya. Menu *Recordset Type* pula membolehkan pengguna untuk edit pangkalan data yang dicapai.

c) Menu *About*

- Menu *About* mengandungi sub menu *Help* dan *E-Mail*. Menu *Help* ini diperlukan oleh pengguna pertama kali menggunakan sistem. Manakala menu *E-Mail* membolehkan pengguna *E-Mail* kepada pembangun sistem bagi melaporkan sebarang masalah mengenai sistem yang dibangunkan.

d) *Frame Clean*

- *Frame Clean* akan mengandungi butang-butang yang akan memaparkan kepada pengguna pengkalan data yang telah dikenalpasti untuk dibersihkan dan memaparkan data bersih kepada pengguna dalam bentuk jadual.

4.2.2 KEPERLUAN BUKAN FUNGSIAN

Keperluan bukan fungsian merujuk kepada cirri-ciri lain yang perlu ada pada sistem serta had-had ataupun halangan terhadap fungsi yang ditawarkan oleh sistem. Ini termasuklah had-had yang wujud pada proses pembangunan sistem dan had masa. [7]

Keperluan bukan fungsian bagi sistem ini disenaraikan seperti berikut :

a) Kebolehgunaan Antaramuka

- Antaramuka yang direka mesti mempunyai ciri-ciri kebolehgunaan yang tinggi. Metafor yang digunakan mestilah membolehkan pengguna memilih menu dan butang yang diperlukan dengan berkesan.

b) Rekabentuk dan Kestabilan Paparan

- Mengelakkan konflik capaian papan kekunci dengan menyediakan dua cara capaian yang sama untuk dua fungsi yang berlainan.
- Menyediakan capaian papan kekunci dan tetikus supaya pengguna dapat memilih cara interaksi yang bersesuaian dengan kehendak mereka.
- Menyediakan penerangan nama untuk setiap komponen antaramuka dan objek yang menggunakan grafik sebagai pengganti tulisan.

c) Ketepatan Dialog

- Mengelakkan penggunaan singkatan yang tidak difahami kerana ia akan menjejaskan kebolehbacaan sistem
- Mengelakkan penggunaan bahasa komputer jargon.
- Menggunakan ayat yang ringkas tetapi jelas maksudnya.

d) Kekonsistenan

- Kekonsistenan adalah perlu supaya pengguna tidak keliru tentang kedudukan mereka samada masih berada di dalam sistem yang sama atau sebaliknya. Kekonsistenan dikekalkan dengan penggunaan tulisan dan warna yang sama bagi setiap antaramuka.

e) Masa Lengahan

- Masa maklumbalas pada capaian data yang telah bersih adalah dalam jangka masa yang cepat.

f) Jelas dan Mudah Difahami

- Antaramuka, butang dan slider perlu jelas maksudnya untuk disampaikan kepada pengguna.

g) Ketepatan

- pengguna membuat capaian pada data yang telah dibersihkan dan data tersebut haruslah tepat dan bebas daripada kesalahan

h) Kewibawaan

- data yang telah dibersihkan adalah boleh dipercayai,mempunyai sifat keaslian dan berautoriti. Sistem membenarkan pengguna melihat pangkalan data sumber sebelum digabungkan bagi membolehkan pengguna menilai kewibawaan data yang telah dibersihkan.

4.3 ANALISA ALATAN PEMBANGUNAN

4.3.1 PERISIAN

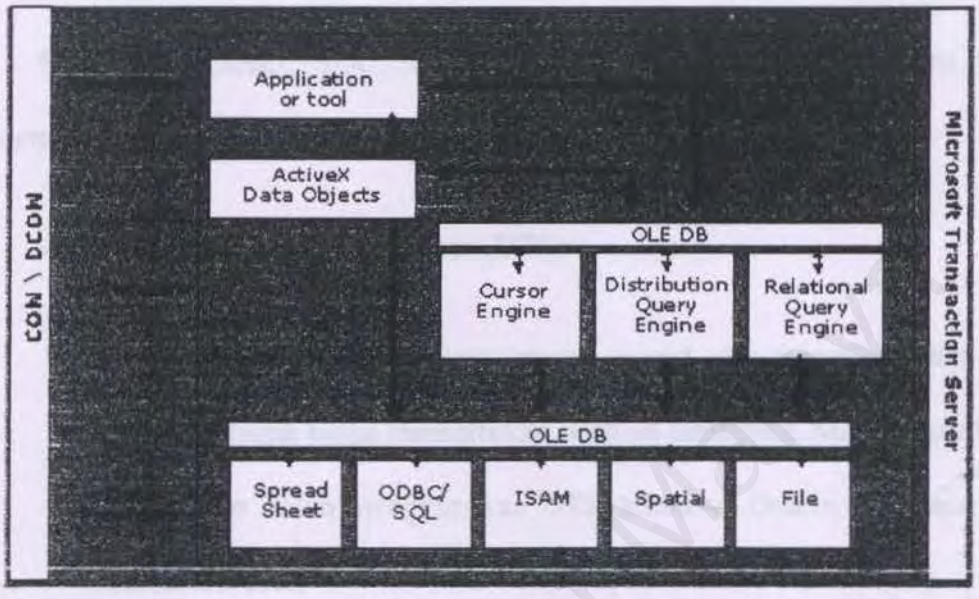
Perisian yang digunakan untuk membangunkan **dCleanViewer** ini ialah :

a) Microsoft Visual Basic 6.0

- Perisian ini dikatakan visual kerana ianya dapat melukis tetingkap, butang, kotak teks dan komponen-komponen lain bagi skrin-skrin dalan sesebuah aturcara. Basic pula merujuk kepada kod aturcara yang ditulis dengan menggunakan BASIC.
- Ianya dipilih berdasarkan beberapa keistimewaan berikut :
    - Visual Basic 6.0 adalah berdasarkan antaramuka pengguna bergrafik (GUI)
    - Visual Basic 6.0 boleh diintegrasikan dengan pangkalan data Microsoft SQL Server 7.0.
    - Ianya sesuai dengan sistem pengdalian windows.
    - Menggunakan konsep pengaturcaraan bermodul. Pengesanan ralat lebih mudah dengan hanya memfokuskan kepada modul yang bermasalah sahaja. Modul-modul lain boleh dilarikan tanpa sebarang masalah.
    - Menyediakan skrip yang dipanggil VBScript untuk mengawal dan menghasilkan antaramuka yang interaktif.
    - Menyokong pengaturcaraan berorientasikan objek (OOP)

b) Microsoft SQL Server 7.0

Microsoft SQL Server 7.0 adalah berasaskan senibina UDA (*Universal Data Access*). Senibina UDA adalah seperti **Rajah 4.1: Senibina UDA**.



Rajah 4.0: Senibina UDA

Microsoft SQL Server 7.0 yang berdasarkan senibina UDA, yang mana ianya diimplimentasikan oleh OLE DB. OLE DB merupakan spesifiasi antaramuka yang menyediakan capaian data teragih tanpa mengambil kira format asal data yang dicapai. Oracle, sebaliknya, menggunakan pendekatan server universal, yang mana semua data perlu wujud dalam gudang data tunggal dan hanya boleh dicapai menggunakan satu bahasa tunggal.

Microsoft SQL Server 7.0 mempunyai kelebihan diatas keupayaan yang wujud melalui senibina UDA, yang mana ianya membenarkan data wujud dalam format yang pelbagai dan boleh dicapai dengan kaedah yang berbeza. Microsoft

SQL Server 7.0 merupakan pangkalan data terhubung (RDBMS) yang berkuasa, dan merupakan mekanisma pengumpulan simpanan maklumat berbeza dan untuk mempersembahkan data secara konsisten tanpa perlu menyepadukan atau menukarkan (*convert*) data yang pelbagai pada satu simpanan data.

Sebagai tambahan, SQL Server 7.0 menyediakan teknologi baru yang membolehkannya bekerja dengan data dalam persekitaran yang pelbagai.

- Data Transformation Services (DTS)

  SQL Server 7.0 membenarkan *import*, *export*, dan mengubah data dari sumber pelbagai tanpa memerlukan perisian tambahan. Mana-mana OLE DB *provider* boleh menggunakan DTS, termasuk Oracle, Informix, dan Microsoft Acces.

- Sokongan pada pertanyaan teragih

  SQL Server 7.0 membenarkan hubungan antara server-server yang jauh (*remote*) dan menggunakan data dari pertanyaan (*queries*) yang datang dari sumber yang pelbagai. Langkah ini adalah tersembunyi kepada pengguna program, yang mana jadual dilihat sebagai jadual SQL Server yang asal, dan meningkatkan rangkaian trafik kerana enjin pertanyaan akan melaksankan sebanyak mungkin kerja yang boleh pada komputer yang jauh. Sebagai tambahan, data tidak perlu dipindahkan dan kekal dalam bentuk simpanan asal.

- *Heterogeneous replication*

  Mana-mana *Open Database Connectivity* (ODBC) *driver* atau OLE DB *data provider* boleh menyertai replikasi SQL Server 7.0. ODBC bermaksud rangkaian pangkalan data terbuka. Teknologi ini membolehkan program berasaskan *Windows* membuat capaian pada pangkalan data dengan menggunakan *driver*.

- Sokongan penyepaduan (*Integrated*) Gudang data.

  Gudang data atau *data marts* adalah mudah untuk dibentuk dari pelbagai pangkalan data terhubung, termasuklah SQL Server, Oracle, and Informix.

4.3.2 PERKAKASAN

Keperluan Perkakasan	Yang Dicadangkan
CPU	Pentium III dan ke atas
RAM	128 MB
Peranti Keluaran	Printer
Peranti Masukan	Papan Kekunci, Tetikus
Cakera Keras	10 GB
Sistem Pengendalian	Windows 2000

Jadual 4.0: Perkakasan

BAB 5 REKABENTUK SISTEM

5.0 REKABENTUK SISTEM

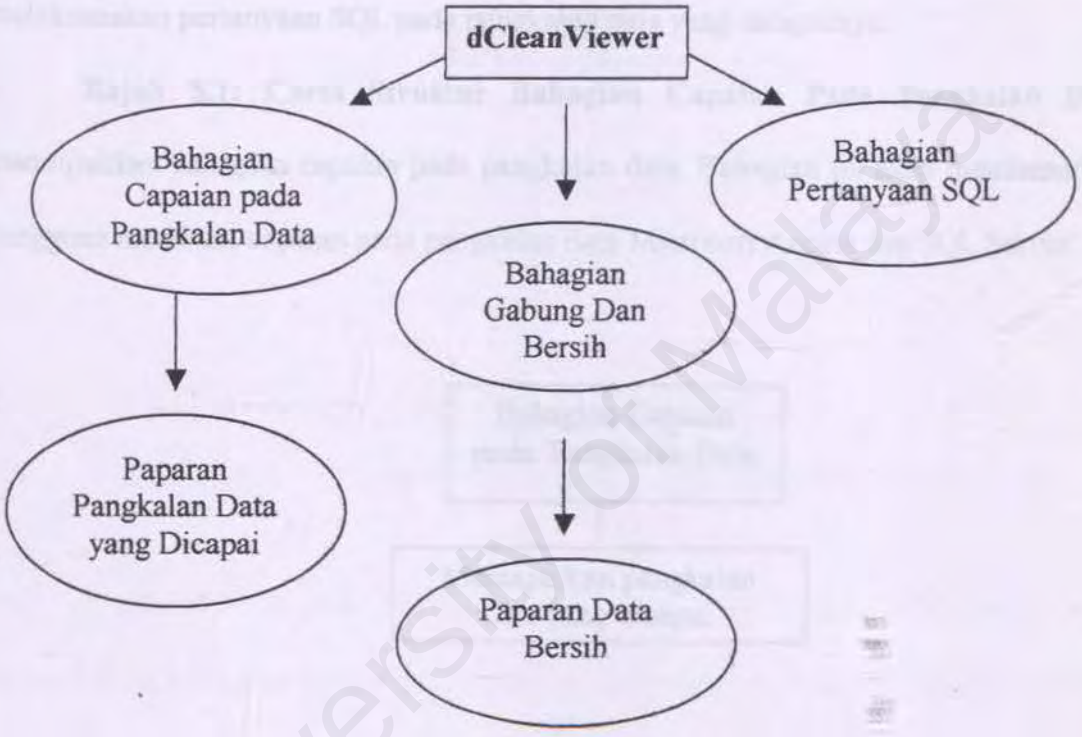
Bab ini menerangkan secara terperinci mengenai bagaimana sistem ini memenuhi keperluan yang dikenalpasti semasa fasa analisis sistem. Rekabentuk sistem merupakan faktor penting dalam pembangunan sistem yang mana ianya akan menentukan kejayaan sesebuah sistem. Spesifikasi sistem menerangkan ciri-ciri dan komponen-komponen sistem dan bagaimana ia dipersembahkan kepada pengguna sistem[15]. Antara rekabentuk utama yang akan dibincangkan dalam bab ini adalah Rekabentuk Proses, Rekabentuk Logikal Sistem dan Rekabentuk Antaramuka Pengguna.

5.1 REKABENTUK PROSES

Pemodelan proses merujuk kepada fungsian atau aspek bagi sistem. Ini termasuk membaca data ke dalam proses, menulis data untuk simpanan data dan cetakan laporan. Rekabentuk ini telah digambarkan dalam Carta Struktur.

5.1.2 CARTA STRUKTUR

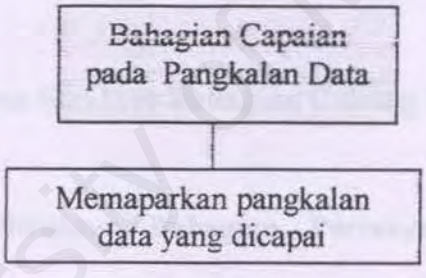
Carta struktur digunakan untuk menunjukkan bagaimana hubungan di antara komponen-konponen dalam **dCleanViewer**. **Rajah 5.0 : Carta Struktur Untuk dCleanViewer** mewakili hirarki carta bagi keseluruhan yang terdapat dalam sistem ini.



Rajah 5.0: Carta Struktur untuk dCleanViewer

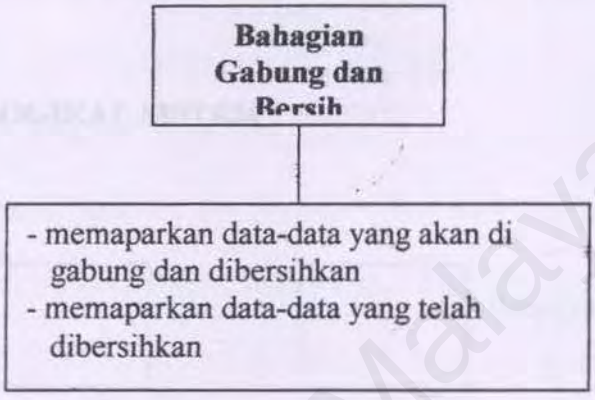
dCleanViewer mengandungi 3 bahagian utama iaitu Bahagian capaian pada pangkalan data , bahagian gabung dan bersih dan bahagian pertanyaan SQL. Secara asasnya bahagian capaian pada pangkalan data membolehkan pengguna iaitu Pentadbir Gudang data melihat pangkalan data sumber dalam format asal sebelum digabungkan. Manakala bahagian gabung dan bersih akan menunjukkan pangkalan data yang akan digabung dan dibersihkan. Bahagian pertanyaan SQL membenarkan pengguna untuk melaksanakan pertanyaan SQL pada pangkalan data yang dicapainya.

Rajah 5.1: Carta Struktur Bahagian Capaian Pada Pangkalan Data menunjukkan bahagian capaian pada pangkalan data. Bahagian ini akan membenarkan pengguna membuat capaian pada pangkalan data Microsoft Access dan SQL Server.



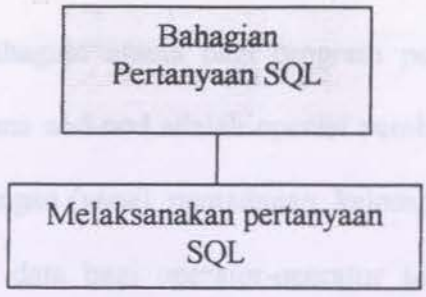
Rajah 5.1: Carta Struktur Bahagian Capaian Pada Pangkalan Data

Rajah 5.2: Carta Struktur Bahagian Gabung Dan Bersih menunjukkan bahagian gabung dan bersih. Bahagian ini akan membolehkan pengguna membuat capaian pada pangkalan data yang akan dibersihkan. Bahagian ini juga akan membolekan pengguna melihat data yang telah dibersihkan.



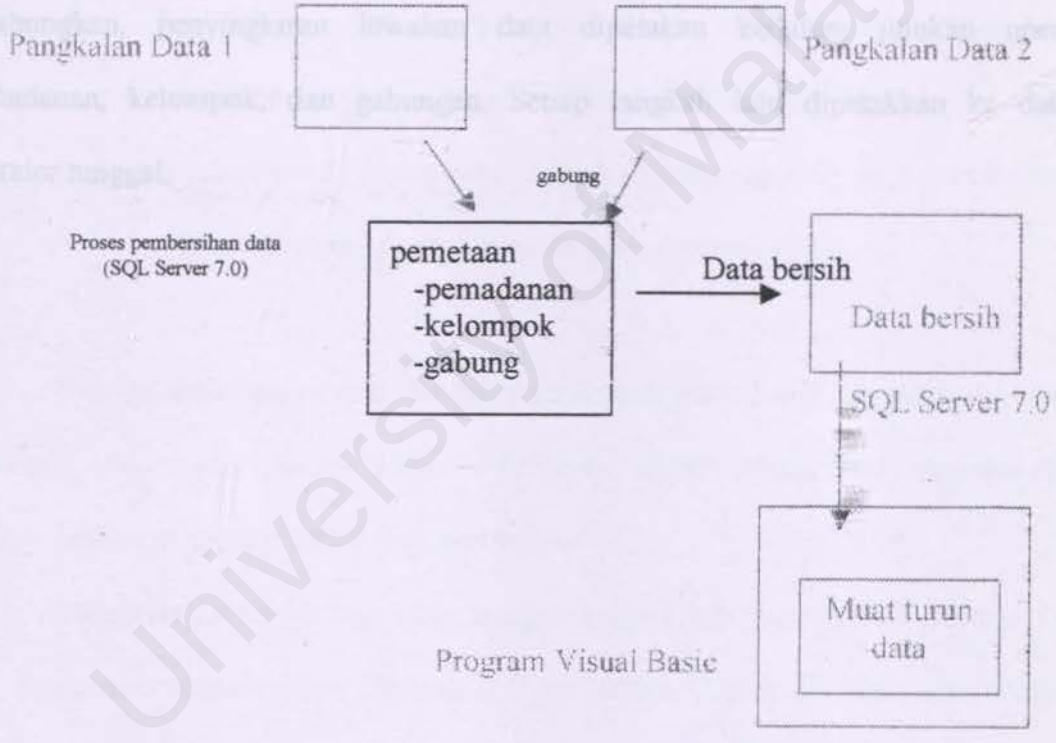
Rajah 5.2: Carta Struktur Bahagian Gabung Dan Bersih

Rajah 5.3: Carta Struktur Bahagian Pertanyaan SQL pula akan menunjukkan bahagian yang akan membenarkan pengguna membuat pertanyaan SQL pada pangkalan data yang dicapai termasuk pada pangkalan data yang telah dibersihkan. Bahagian ini akan melaksanakan pertanyaan SQL yang dibuat oleh pengguna.



Rajah 5.3: Carta Struktur Bahagian Pertanyaan SQL

5.2 REKABENTUK LOGIKAL SISTEM



Rajah 5.4: Rangka Kerja Sistem Pembersihan Data

Pada paras logikal, bahagian utama bagi program pembersihan data adalah spesifikasi aliran data yang mana nod-nod adalah operasi pembersihan data bagi jenis-jenis berikut: pemetaan, pandangan (*view*), pemadanan, kelompok (*cluster*), gabungan, dan input serta output aliran data bagi operator-operator secara logik dimodelkan sabagai pangkalan data hubungan. Rekabentuk operator logikal adalah berdasarkan semantik SQL yang menyokong perubahan data semasa pembersihan data. Setiap operator boleh digunakan untuk takrifan luaran fungsian atau algoritma seperti penormalan rentetan (*string*), pengekstrakan subrentetan daripada rentetan dan lain-lain.

Bagi setiap aliran data output semasa pemetaan bagi dua pangkalan data yang digabungkan, penyingkiran lewahan data dipetakan kedalam jujukan operasi pemadanan, kelompok, dan gabungan. Setiap langkah lain dipetakkan ke dalam operator tunggal.

5.3 REKABENTUK ANTARAMUKA

Rekabentuk grafik antara muka pengguna perlu memenuhi empat objektif utama[16] iaitu :

i. Keberkesanan dalam membenarkan pengguna membuat capaian kepada sistem dengan cara yang sesuai dengan keperluan individu mereka.

ii. Kecekapan dalam kelajuan masukan data dan mengurangkan bilangan kemunculan ralat.

iii. Pertimbangan terhadap pengguna dengan menyediakan maklumat yang sesuai dan berguna sebagai maklum balas kepada pengguna.

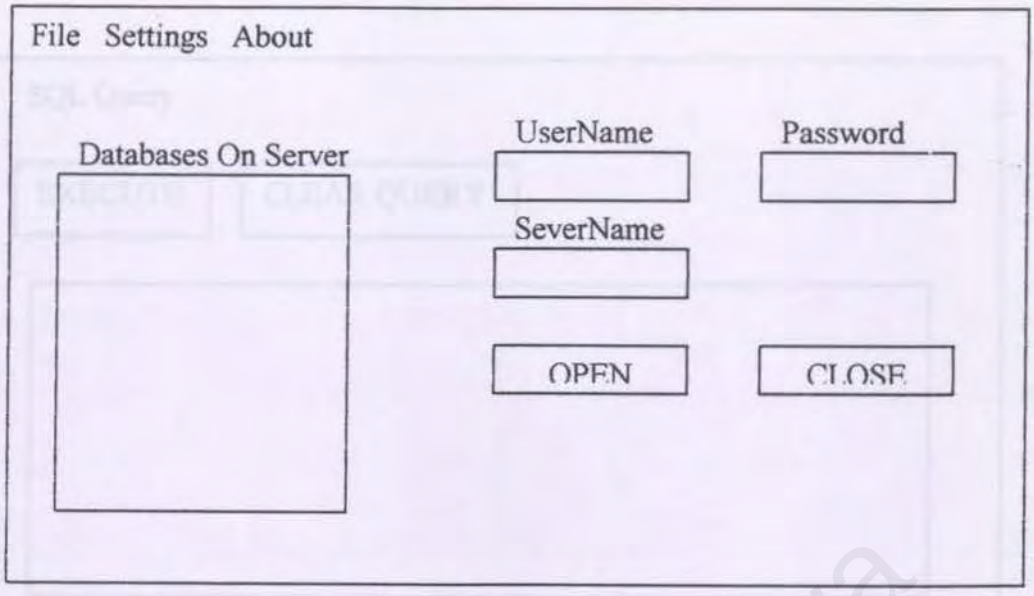
iv. Produktiviti diukur melalui prinsip-pronsip ergonomi bagi merekabentuk antaramuka pengguna dan ruang kerja (*workspaces*).

Bagi **dCleanViewer**, antaramuka utamanya akan menyediakan menu *File*, menu *Settings*, dan menu *About*. Menu *File* akan menyediakan satu fungsian yang membolehkan pengguna keluar daripada sistem.

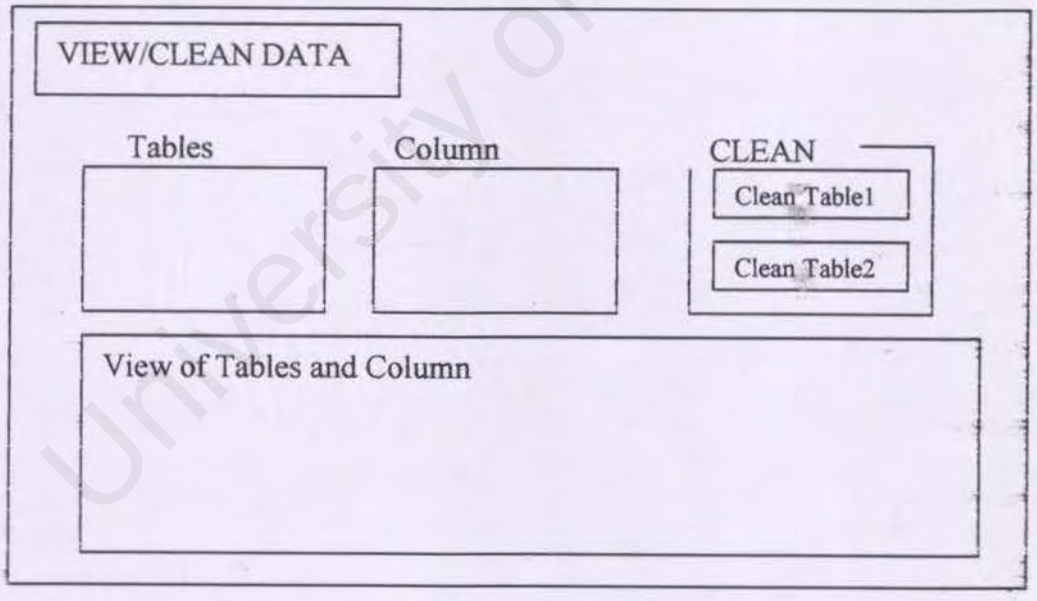
Manakala menu *Settings* akan mengandungi 3 sub menu iaitu *Database Type*, *NT Integrated Security* dan *Recordset Type*. Menu *Database Type* membolehkan pengguna membuat capaian pada dua jenis pangkalan data iaitu Microsoft Access dan SQL Server. Manakala menu *NT Integrated Security* membolehkan pengguna yang telah memilih untuk membuat capaian pada pangkalan data SQL Server mencapainya

dengan akaun domain NT atau sebaliknya. Menu *Recordset Type* membolehkan pengguna memilih untuk membuat pengeditan pada pangkalan data yang dicapai.

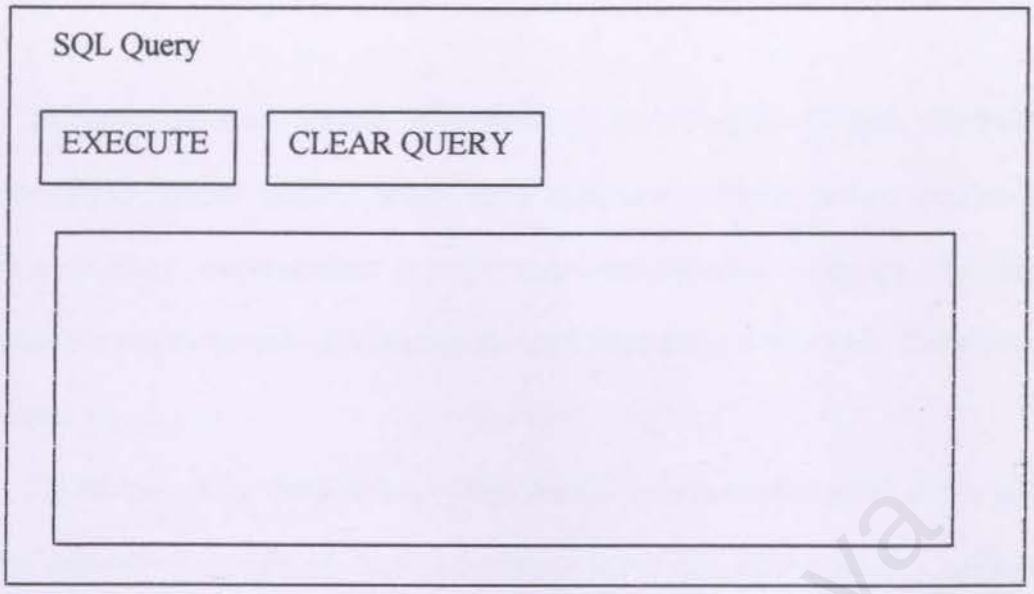
Menu *About* pula mengandungi sub menu *Help* dan *E-Mail*. Menu *Help* adalah penerangan tentang **dCleanViewer** dan digunakan untuk pengguna yang pertama kali menggunakan **dCleanViewer**. Manakala menu *E-Mail* membolehkan pengguna menghantar *E-Mail* kepada pembangun sistem bagi melaporkan sebarang aduan atau masalah berkaitan dengan **dCleanViewer**.



Rajah 5.5: Rekabentuk Antaramuka Utama dCleanViewer



Rajah 5.6: Rekabentuk Antaramuka *View/Clean Data* Pada dCleanViewer



Rajah 5.7: Rekabentuk Antaramuka SQL *Query* dCleanViewer

BAB 6 PEMBANGUNAN SISTEM

6.0 PENGENALAN

Merujuk kepada **Rajah 3.0 Model Air Terjun dengan Prototaip**, pemprototaipan adalah langkah yang harus dilakukan semasa proses menganalisa keperluan sistem, merekabentuk sistem dan merekabentuk program. Ia adalah merupakan proses berterusan dan berulangan bagi rekabentuk antaramuka dan penilaian pengguna.

Tujuannya adalah untuk memodelkan produk yang akan dibangunkan disamping menguji ciri-ciri pada sistem bagi memenuhi keperluan dan kehendak pengguna seterusnya memperbaiki kelemahan yang ada pada sistem. Terdapat banyak jenis pendekatan prototaip dalam pengujian pembangunan sistem. Antaranya adalah seperti Prototaip Melintang (horizontal), Prototaip Menegak (vertical), Modular Prototaip, *Evolutionary Prototype* dan *Throwaway Prototype.*

Prototaip yang digunakan ialah gabungan antara Prototaip Melintang dan *Evolutionary Prototype*. Dimana Prototaip Melintang merangkumi proses ciri-ciri dan fungsi secara keseluruhan tetapi tidak berfungsi sepenuhnya atau berfungsi dengan lebih mendalam. Ia sangat sesuai untuk pengujian tetapi tidak untuk kegunaan sebenar. Manakala *Evolutionary Prototype* adalah di mana prototaip awal diubah suai dari semasa ke semasa dan digunakan untuk prototaip yang seterusnya sehinggalah kepada produk akhir.

6.1 PENETAPAN OBJEKTIF PEMPROTOTAIPAN

Objektif utama sistem ini adalah melakukan proses pembersihan data bagi dua buah pangkalan data yang berkaitan. Sebagai projek permulaan, sistem telah menetapkan dua buah pangkalan data yang akan dibersihkan. Proses pembersihan menekankan pada masalah data yang tidak konsisten diantara dua buah pangkalan data. Selain itu, sistem membolehkan pengguna sasaran melihat dua buah pangkalan data dari format yang berbeza iaitu pangkalan data Microsoft Access dan pangkalan data SQL Server. Sistem ini juga membenarkan pengguna membuat pertanyan SQL (*Structured Query Language*) pada pangkalan data yang dicapai. Oleh itu, adalah penting bagi sistem untuk menghasilkan data yang benar-benar bersih dan konsisten serta membenarkan pengguna membuat capaian pada data bersih tersebut. Pembangunan prototaip antaramuka merupakan langkah yang penting bagi menghasilkan suatu antaramuka yang mudah difahami oleh pengguna sasaran. Antara objektif yang ingin dicapai ialah:

i. Sistem berupaya membersihkan pangkalan data-pangkalan data yang telah ditetapkan

ii. Untuk memastikan kebolehterimaan pengguna terhadap gaya antaramuka.

iii. Untuk memastikan pengguna memahami bagaimana setiap antaramuka berfungsi.

iv. Untuk mengesahkan keperluan setiap antaramuka adalah dipenuhi.

v. Untuk memastikan penambahan yang dibuat adalah benar-benar mencapai kebolehgunaan yang sebenar dan pengubahsuaian itu telah menyelesaikan masalah yang ditemui dalam prototaip yang sebelumnya.

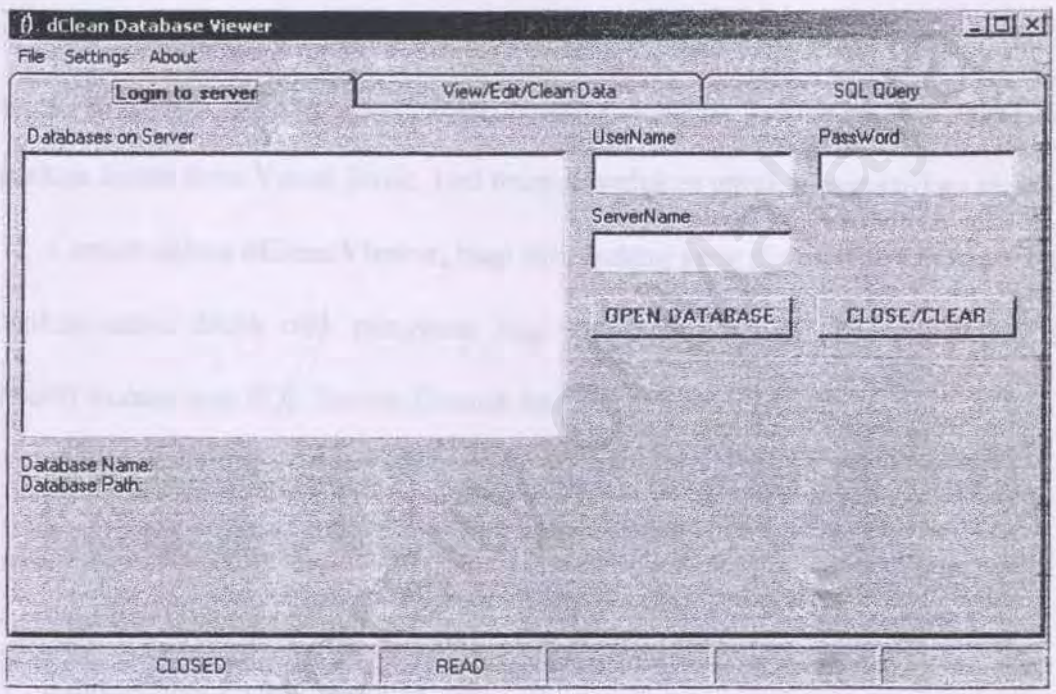
6.2 PEMBANGUNAN ANTARAMUKA PENGGUNA

Antaramuka pengguna dibangunkan menggunakan perisisan Visual Basic 6.0 menyediakan banyak ciri-ciri menarik dalam merekabentuk rupa borang (*form*), menentukan *event* dan komponen-komponen untuk memudahkan pembangunan suatu sistem kecil.

Perisian ini menyediakan bebrapa pilihan untuk memulakan pembangunan sistem. **dCleanViewer** dibangunkan dengan format projek *Standard Exe*, yang mana ianya memberikan persekitaran untuk pembangunan berasaskan Windows yang biasa. Rupa persekitaran ini dikenali sebagai VBIDE atau Integrated Development Environment yang mewujudkan satu persekitaran pengaturcaraan yang menarik dan berstruktur. Visual Basic memudahkan rekabentuk antaramuka sekaligus menghidupkan skrin tersebut melalui pengkodan.

Komponen Microsoft Tabbed Dialog Control 6.0 telah digunakan untuk membina **dCleanViewer** dan membahagikan **dCleanViewer** kepada 3 bahagian utama. Bahagian pertama akan membenarkan pengguna membuat capaian pada 2 jenis pangkalan data berlainan format iaitu pangkalan data Microsoft Access dan SQL Server. Bahagian kedua akan memaparkan pangkalan data yang telah dicapai dalam

bentuk jadual dengan menggunakan Komponen Microsoft DataGrid Control 6.0. Bahagian ketiga akan membentuk antaramuka yang membolehkan pengguna melaksanakam pertanyaan SQL pada pangkalan data yang dicapai. Microsoft Common Dialog Control 6.0 digunakan untuk menghubungkan skrin dengan pangkalan data. **Rajah 6.0: Antaramuka utama dCleanViewer** menunjukkan antaramuka utama bagi projek **dCleanViewer**.



Rajah 6.0: Antaramuka Utama dCleanViewer

6.3 PENGATURCARAAN

Meskipun Visual Basic memberikan satu rekabentuk antaramuka yang menarik, skrin tidak akan menjadi aktif tanpa pengaturcaraan. Pengaturcaraan merupakan suatu proses penterjemahan logik setiap spesifikasi aturcara yang telah disediakan semasa fasa rekabentuk sistem ke bentuk kod-kod arahan dalam bahasa pengaturcaraan. Matlamat pengaturcaraan menggunakan Visual Basic ialah untuk menyediakan satu kod bebas yang akan diaktifkan apabila sesuatu *event* berlaku (disesuaikan dengan sifat kod yang setiap objek miliki). Bagi setiap *control* (seperti butang, kotak teks) yang digunakan diatas skrin Visual Basic, kod tetap diperlukan untuk mengaktifkan objek.

Contoh dalam **dCleanViewer,** bagi satu butang yang dinamakan cmdCon_Click digunakan untuk diklik oleh pengguna bagi membuat capaian pada pangkalan data Microsoft Access atau SQL Server. Contoh kod aturcarannya adalah :

```
Private Sub cmdCon_Click()
On Error GoTo ErrHandler
mnuDBType.Enabled = False
Set objCon = New ADODB.Connection
Set objRS = New ADODB.Recordset


If AccessDbPath = "" Then
  GetCon
```

```
    sqlDatBase = "Master"
    GetRecordSet ("exec sp_databases")
    If objRS.State = adStateOpen Then
      While Not objRS.EOF = True
        lstDatBas.AddItem objRS.Fields(0)
        objRS.MoveNext
      Wend
    End If
  StatusBar1.Panels(4).Text = "Found: " & objRS.RecordCount & " Databases"
  Else
    sqlDatBase = AccessDbPath
    GetCon
    Set objRS = objCon.OpenSchema(adSchemaTables)
  While Not objRS.EOF
  If objRS!TABLE_TYPE = "TABLE" Then lstTables.AddItem
objRS!TABLE_NAME
    objRS.MoveNext
  Wend
  End If
  cmdCon.Enabled = False
  cmdTSql.Enabled = True
  cmdClose.Enabled = True
  mnuDBType.Enabled = False
```

```
If bolAccess = True Then SSTab1.Tab = 1
CheckConState
ErrHandler:
If Err.Number <> 0 Then
  Call CentralErrhandler("cmdCon_Click")
  mnuDBType.Enabled = True
  cmdClose.Enabled = True
End If
End Sub
```

Pembangunan **dCleanViewer** dimulakan dengan pengenalpastian 2 buah pangkalan data yang berkaitan tetapi berlainan format untuk digabung dan dibersihkan. Ianya diikuti dengan rekabentuk skrin menggunakanVisual Basic. Untuk menyempurnakan **dCleanViewer** sebagai suatu aplikasi, pengkodan diperlukan untuk mengintegrasikan kedua-dua komponen ini. Proses pengaturcaraan ini mengambil masa hampir 3 bulan.

Fasa pencapaian pada dua buah pangkalan data berlainan format dan fasa pembersihan gabungan dua buah pangkalan data merupakan fasa yang paling mencabar dan memerlukan teknik pengaturcaraan yang rumit. Pembangun sistem harus memastikan **dCleanViewer** yang dibangunkan dapat membuat capaian dan memaparkan dua buah pangkalan data sebelum digabungkan dalam format asal disamping dapat memaparkan data-data yang telah dibersihkan dalam pangkalan data

SQL Server. Proses pembersihan adalah berdasarkan ralat atau data kotor yang wujud dalam pangkalan data-pangkalan data yang telah dikenalpasti.

BAB 7 PENGUJIAN SISTEM

7.0 PENGENALAN

Pengujian sistem dilakukan adalah bertujuan untuk memastikan sistem yang dibangunkan dapat beroperasi dengan baik dan berkesan. Selain itu pengujian juga adalah bertujuan untuk memastikan keseluruhan bahagian yang dibangunkan memenuhi skop dan keperluan yang telah ditetapkan. Disamping itu, pengujian sistem membolehkan sebarang ralat dikenalpasti dan seterusnya dapat diatasi. Faktor yang lebih penting ialah kebolehpercayaan terhadap keupayaan sistem dapat ditingkatkan. Glen Myres [19] menyatakan tiga peraturan yang boleh dijadikan sebagai objektif pengujian, iaitu :

i. Pengujian ialah proses melaksanakan aturcara dengan tujuan mengesan kesilapan.

ii. Kes pengujian yang baik ialah kes yang berdaya tinggi untuk mengesan kesilapan yang masih wujud.

iii. Pengujian yang berjaya ialah pengujian yang dapat mengesan kesilapan yang masih wujud.

Pengujian perlu direkabentuk untuk mengesan kesilapan yang berlainan jenis secara teratur, dengan menggunakan masa dan usaha paling minima. Selain mengesan kesilapan, pengujian juga memberi gambaran bahawa sistem yang dibina berfungsi mengikut spesifikasi yang diberikan. Bagimanapun, jika pengujian tidak mengesan sebarang kesilapan, itu tidak membuktikan bahawa sistem adalah bebas dari kesilapan atau sudah memenuhi spesifikasi yang telah ditetapkan.

7.1 PERSEKITARAN PEMBANGUNAN

Persekitaran pembangunan adalah memberi kesan kepada pembangunan sesebuah sistem. Dengan menggunakan perkakasan dan perisian yang bersesuaian membantu mempercepatkan proses pembangunan sistem. Di samping itu, penggunaan perisian terbaru yang lebih baik akan memudahkan proses pelaksanaan pembangunan sistem terutamanya dalam merekabentuk antaramuka dan seterusnya dapat mengintegrasikan antaramuka tersebut dengan pangkalan data-pangkalan data dari format yang berbeza. Peralatan perkakasan dan perisian yang digunakan untuk membangunkan **dCleanViewer** ialah :

Keperluan perkakasan

- Komputer peribadi
- Ruang ingatan 64 MB RAM
- Pemproses Intel Pentium III - kelajuan sekurangnya 500 mmx
- Hard disk 10 GB
- Monitor SVGA/VGA
- Papan kekunci dan tetikus

Keperluan perisian

- Microsoft Visual Basic 6.0
- Microsoft Access 2000
- Microsoft SQL Server 7.0
- Sistem Pengendalian-Windows 2000

7.2 STRATEGI PENGUJIAN

Strategi pengujian menghuraikan pendekatan bagaimana pengujian akan dilaksanakan. Ia merangkumi perancangan pengujian, rekabentuk pengujian dan penilaian hasil pengujian. Pengujian **dCleanViewer** bermula secara kecilan, dimana setiap komponen akan diuji menggunakan teknik kotak hitam.

Pengujian kotak hitam dikenali juga sebagai pengujian kelakuan perisian. Ia menguji ketepatan fungsi perisian berdasarkan spesifikasi keperluan. Ia adalah pelengkap kepada pengujian kotak putih, dan dapat mengesan kesilapan dari jenis lain pula. Pengujian kotak hitam cuba mengesan kesilapan dari jenis :

i. Fungsi yang tertinggal dan tidak betul.

ii. Kesalahan antaramuka.

iii. Kesalahan struktur data atau capaian.

iv. Kesalahan kelakuan dan kemampuan.

v. Kesalahan pada rutin awalan dan akhiran.

Dengan menggunakan kaedah pengujian kotak hitam, kes pengujian yang dijana akan dapat memenuhi kriteria berikut.

i. Dapat mengurangkan bilangan pengujian tambahan.

ii. Dapat mengesan kehadiran atau ketiadaan sesuatu kesilapan.

Seterusnya apabila komponen diintegrasikan menjadi satu sistem, pengujian integrasi akan dilaksanakan. Proses pengujian yang dijalankan perlu menggunakan suatu pendekatan yang teratur dan berstruktur.

Oleh itu, strategi pengujian yang digunakan adalah meliputi dua peringkat utama, iaitu :

i. Pengujian unit

ii. Pengujian integrasi

Pengujian dilakukan peringkat demi peringkat bagi memastikan sistem yang dibangunkan mudah untuk digunakan oleh pengguna. Pengujian amat penting dalam menentukan kesalahan-kesalahan ralat yang boleh memberikan masalah kepada perlaksanaan sistem yang telah dibangunkan. Pengujian yang dilakukan adalah meliputi fungsian-fungsian yang ada pada **dCleanViewer**.

7.2.1 PENGUJIAN UNIT

Pengujian unit lebih menumpukan kepada pengujian terhadap rekabentuk unit terkecil aturcara. **dCleanViewer** terbahagi kepada 3 bahagian utama, yang mana setiap bahagian adalah satu koleksi komponen-komponen saling berkait antara satu komponen dengan komponen yang lain.

Pengujian dijalankan terlebih dahulu bagi memastikan komponen yang dibina adalah memenuhi keperluan pengguna serta memastikan tidak berlaku sebarang ralat semasa sistem dijalankan. Langkah pertama dalam pengujian unit adalah memeriksa kod program melalui bacaan bagi mengesan kesalahan algoritma dan sintaks. Ianya diikuti dengan membandingkan kod dengan spesifikasi dan rekabentuk untuk memastikan semua kes yang berkaitan dipertimbangkan. Seterusnya, butang *run* pada Visual Basic dilaksanakan bagi melihat perlaksanaan fungsi pada sistem dan mengesan ralat-ralat yang wujud pada sistem.

Pengujian unit tertumpu kepada 3 bahagian utama pada **dCleanViewer**. Pengujian unit terbahagi kepada tiga iaitu :

i. Pengujian 1 – Capaian pada pangkalan data

ii. Pengujian 2 – Pemaparan pangkalan data yang dicapai dan hasil akhir pangkalan data yang telah dibersihkan.

iii. Pengujian 3 – Pelaksanaan Bahasa Pertanyaan Berstruktur (SQL)

Contoh Kes Pengujian Unit

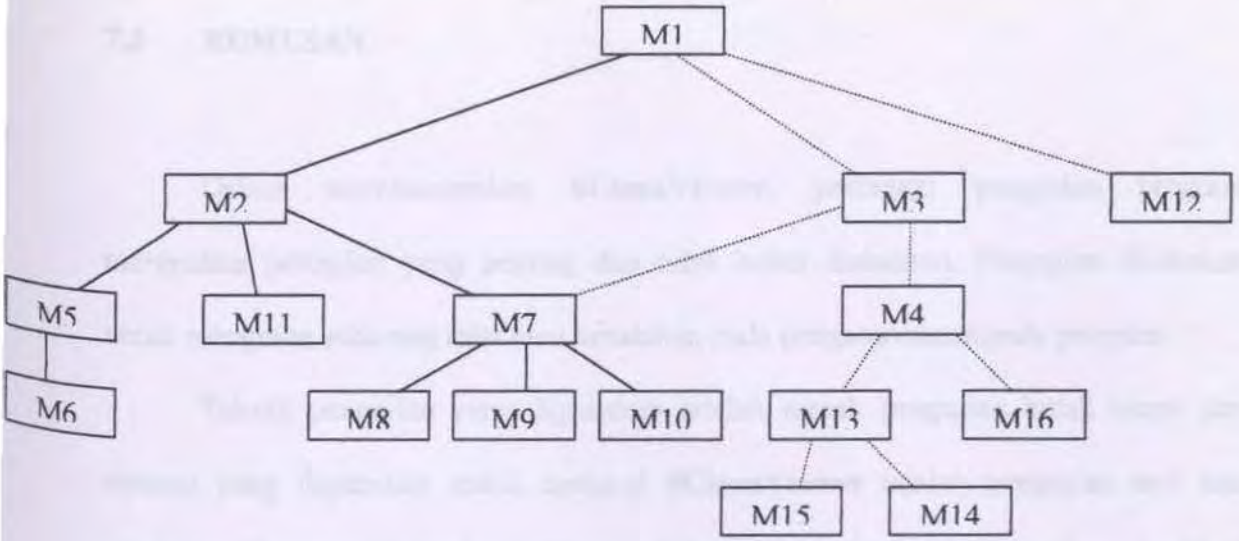
Pengujian 2 melibatkan bahagian pembersihan data dalam **dCleanViewer** yang akan melakukan proses pembersihan dan gabungan pangkalan data berdasarkan 2 buah pangkalan data yang telah dikenalpasti atau ditetapkan. Pengujian unit dilakukan bagi memastikan **dCleanViewer** berupaya untuk menggabung dan membersihkan data-data dari 2 buah pangkalan data yang telah dikenalpasti.

Langkah	Prosedur pengujian	Output yang dijangkakan	Analisis hasil pengujian
1	Membuat capaian pada pangkalan data yang akan dibersihkan	Pemaparan data yang telah digabung dan dibersihkan dalam bentuk jadual	Satu jadual baru yang mengandungi data-data yang telah digabung dan dibersihkan berjaya dibentuk dan dipaparkan

7.2.2 PENGUJIAN INTEGRASI

Pengujian Integrasi yang dijalankan pada **dCleanViewer** merupakan teknik sistematik dalam pembinaan struktur program dan dalam pada masa yang sama ujian dilakukan bagi mengenalpasti ralat yang berkaitan dengan antaramuka. Pengujian Integrasi Atas-Bawah di jalankan pada **dCleanViewer**, yang mana ianya memberikan pendekatan untuk membina struktur program. Modul-modul di integrasikan dengan bergerak kebawah melalui hirarki kawalan, bermula dengan modul kawalan utama (program utama). Sub modul-sub modul bagi modul utama akan digabungkan ke dalam struktur mengunakan *dept-firt manner*[19].

Dengan merujuk **Rajah 7.0: Integrasi Depth-First** akan mengintegrasikan semua komponen pada laluan kawalan major pada aplikasi–karakteristik spesifik. Sebagai contoh, pemilihan laluan sebelah kanan, komponen M1, M2, M5, M6, M7, M8, M9, M10, M11 akan di integrasikan terlebih dahulu bagi menguji fungsian keseluruhan bagi M2. Kemudian diikuti dengan pengujian melalui laluan tengah dan kiri.



Rajah 7.0: Integrasi Depth-First

Petunjuk:
M1 - Pilih pangkalan data
M2 - Capaian pada pangkalan data Microsoft Access
M3 - Capaian pada pangkalan data SQL Server
M4 - Capaian pangkalan SQL Server menggunakan NT domain akaun atau sebaliknya
M5 - Buka dan papar pangkalan data Microsoft Access
M6 - Baca / edit pangkalan data Microsoft Access
M7 - Pertanyaan SQL
M8 - Simpan pertanyaan SQL
M9 - Buka pertanyaan SQL
M10- Padamkan pertanyaan SQL
M11- Tutup pangkalan data Microsoft Access yang dicapai
M12- Keluar dari sistem
M13- Buka dan papar pangkalan data SQL Server
M14- Bersihkan data
M15- Baca / edit pangkalan data SQL Server
M16- Tutup pangkalan data SQL Server yang dicapai

7.3 RUMUSAN

Dalam membangunkan **dCleanViewer**, peringkat pengujian program merupakan peringkat yang penting dan tidak boleh diabaikan. Pengujian dilakukan untuk mengesan sebarang ralat atau kesalahan pada pengaturcaraan pada program.

Teknik pengujian yang digunakan adalah teknik pengujian kotak hitam dan strategi yang digunakan untuk menguji **dCleanViewer** adalah pengujian unit dan pengujian integrasi. Objektif pengujian ini adalah untuk memastikan kod-kod pengaturcaraan yang diimplementasikan pada rekabentuk adalah secukupnya dan bebas dari ralat.

8.0 KEPUTUSAN YANG DIPEROLEHI

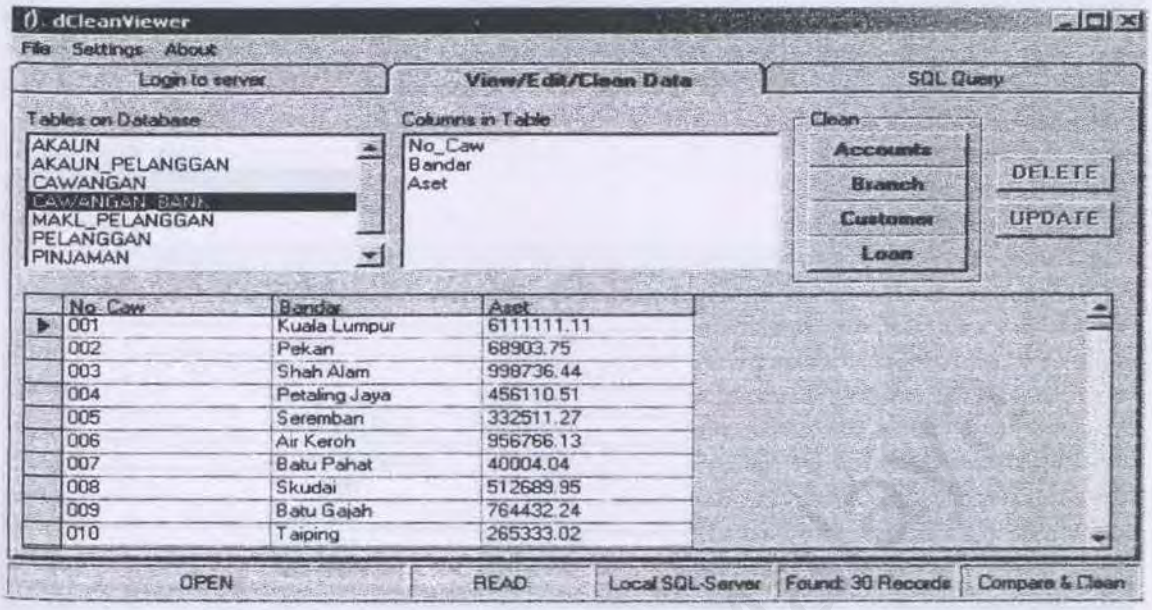
dCleanViewer yang dihasilkan adalah berasaskan dan berpandukan kepada skop dan objektif yang telah ditetapkan. Antara ciri-ciri yang dimiliki oleh **dCleanViewer** ini ialah :

i. **Melakukan Pembersihan 2 Buah Pangkalan Data Berkaitan**

Proses pembersihan yang dijalankan oleh **dCleanViewer** melibatkan penyelarasan data yang tidak konsisten. **dCleanViewer** memaparkan hasil data bersih yang diperolehi dari penyatuan dua buah pangkalan data yang telah ditetapkan. Proses pembersihan dijalankan pada pangkalan data SQL Server.

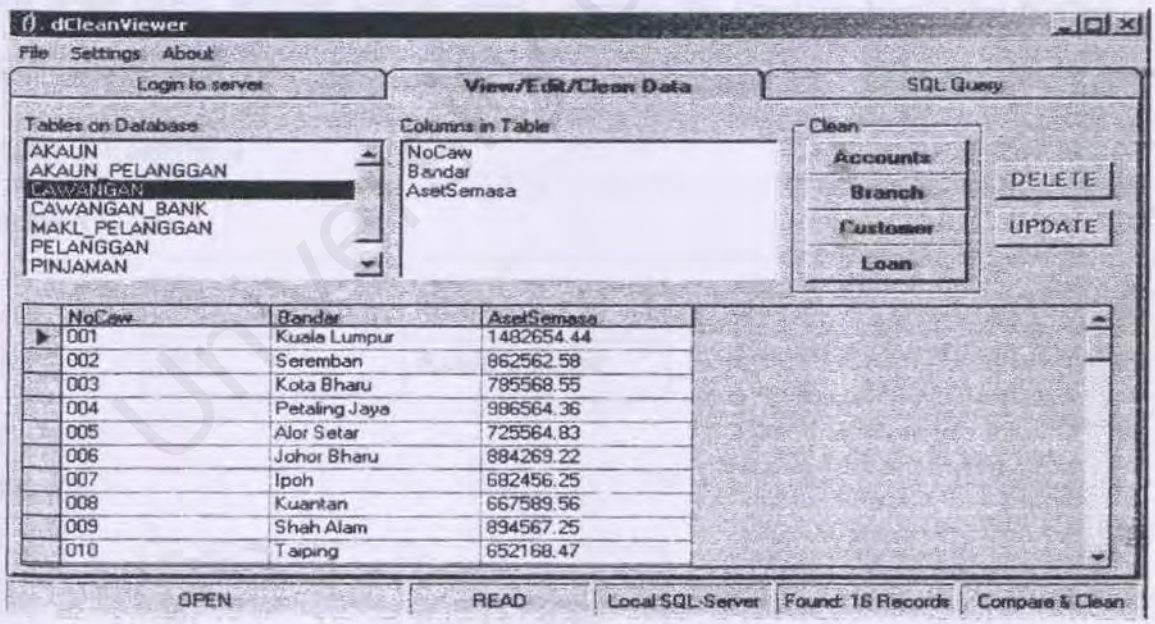
Contoh Pembersihan Data

Sumber 1 (Jadual CAWANGAN_BANK)



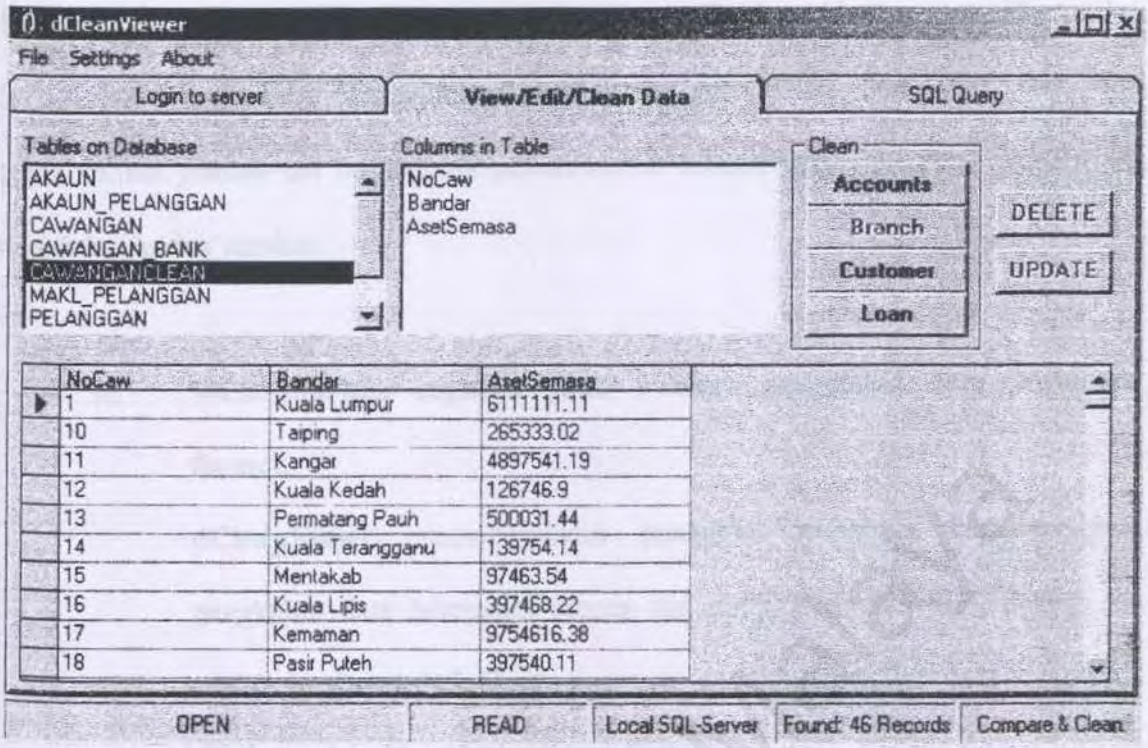
Rajah 8.0: Sumber 1

Sumber 2(Jadual CAWANGAN)



Rajah 8.1: Sumber 2

Penyepaduan sasaran dengan data bersih (CAWANGANCLEAN)



Rajah 8.3: Jadual Yang Mengandungi Data Bersih

Pada tahap skima, terdapat konflik nama [sinonim No_Caw(Sumber 1) dan NoCaw (Sumber 2)]. Terdapat juga pertindanan perwakilan pada data bagi sumber 1 dan sumber 2. Contohnya

Sumber 1

No_Caw	Bandar
001	Kuala Lumpur

Sumber 2

NoCaw	Bandar
001	Kuala Lumpur

Setelah data digabung dan dibersihkan konflik penamaan entiti telah diatasi dengan menyelaraskannya kepada NoCaw. Selain itu, perwakilan bagi NoCaw juga diberi nilai perwakilan semula bagi mengelakkan pertindanan perwakilan. Pada status bar juga boleh dilihat jumlah set rekod bagi jadual bersih adalah hasil dari penyepaduan set rekod dari kedua sumber.

ii. **Membenarkan capaian pada 2 buah pangkalan data berlainan format.**

dCleanViewer membolehkan pengguna membuat capaian pada pangkalan data Microsoft Acces dan pangkalan data SQL Server. Ini adalah bagi membolehkan pengguna melihat pangkalan data yang akan digabung dan dibersihkan dalam format asal.

iii. **Pertanyaan SQL**

Apabila pengguna telah membuat capaian pada pangkalan data, pengguna boleh melaksanakan pertanyaan SQL pada pangkalan data yang dicapai. Selain itu, pengguna juga boleh membuat pertanyaan SQL pada pangkalan data yang telah dibersihkan. Pertanyaan SQL ini turut membenarkan pengguna mengubah suai pangkalan data yang dicapai dengan menghapuskan atau menambah jadual pada pangkalan data yang telah dicapai.

iv. **Pengeditan pada pangkalan data**

dCleanViewer juga membenarkan pengguna untuk membuat pengeditan pada pangkalan data yang dicapai.

iv. **Pengeditan pada pangkalan data**

dCleanViewer juga membenarkan pengguna untuk membuat pengeditan pada pangkalan data yang dicapai.

8.1 KELEBIHAN SISTEM

dCleanViewer ini dapat memberi kelebihan kepada pengguna, antaranya :

i. **Antaramuka yang konsisten dan mudah.**

Rekabentuk antaramuka ynag dibangunkan adalah sistematik dan teratur bagi memudahkan pengguna mengunakan sistem ini. Ciri-ciri antaramuka yang baik yang dipelajari dan di analisis dari kuliah, buku dan sumber maklumat yang lain diimplimentasikan bagi mewujudkan suatu antara muka yang mudah untuk difahami oleh pengguna.

ii. **Membenarkan capaian pada dua jenis pangkalan data yang berlainan format dalam satu antaramuka.**

dCleanViewer membenarkan pengguna membuat capaian pada pangkalan data Microsoft Access dan SQL Server. Ini adalah bagi membenarkan pangguna melihat dan membuat capaian pada pangkalan data yang diintegrasikan dalam format yang asal.

iii. **Capaian yang mudah**

dCleanViewer membenarkan pengguna melakukan proses pembersihan data pada pangkalan data yang telah ditetapkan tanpa perlu membuka pangkalan data tersebut dalam format asal. Selain itu, pengguna juga tidak perlu membuka banyak aplikasi bagi membuat capaian pada pangkalan data Microsoft Acces dan SQL Server kerana **dCleanViewer** membenarkan capaian pada pangkalan data-pangkalan data tersebut. **dCleanViewer** juga membenarkan pengguna untuk mengubahsuai data pada pangkalan data yang dicapainya.

iv. **Membenarkan pertanyaan SQL pada pangkalan data yang dicapai.**

Ciri atau fungsi tambahan yang di wujudkan pada **dCleanViewer** ini membenarkan pengguna membuat pertanyaan menggunakan sintaks SQL. Oleh kerana pengguna sasaran merupakan mereka yang berpengetahuan dalam pengurusan pangkalan data, maka ianya memudahkan pengguna untuk mendapatkan maklumat daripada pangkalan data yang dicapai selain dapat membuat capaian pada data yang telah dibersihkan.

v. **Masa capaian pada pangkalan data, pembersihan pangkalan data dan pelaksanaan pertanyaan SQL yang pantas**

Pengguna boleh membuat capaian pada pangkalan data,capaian pada pembersihan data dan perlaksanaan pertanyaan SQL dalam tempoh masa yang pantas. Masa maklumbalas yang diberi adalah diantara 2-3 saat.

8.2 Kelemahan Sistem

Terdapat beberapa kelemahan yang dikenalpasti, antara kelemahan tersebut ialah :

i. **Pembersihan data pada pangkalan data yang terhad**

Pembersihan pangkalan data dari sumber yang pelbagai hanya dapat dilakukan pada pangkalan data yang telah ditetapkan. **dCleanViewer** tidak dapat melakukan proses pembersihan data bagi penyatuan pangkalan data yang berlainan atau menghadapi masalah data kotor yang berbeza.

ii. **Langkah pembersihan pada pangkalan data yang digabungkan tidak ditunjukkan.**

Pengguna hanya akan melihat hasil akhir data yang telah dibersihkan. Langkah-langkah pembersihan pangkalan data tidak ditunjukkan kepada pengguna.

iii. **Capaian pada SQL Server berjenis local**

Pengguna hanya boleh membuat capaian pada pangkalan data SQL Server berjenis *local* sahaja. **dCleanViewer** tidak menyediakan satu fungsian yang membolehkan pengguna membuat capaian pada pangkalan data SQL Server berjenis *remote*.

8.3 MASALAH DAN PENYELESAIAN

Sepanjang membangunkan **dCleanViewer** ini terdapat beberapa masalah yang dihadapi. Antara masalah yang dihadapi ialah :

i. **Keselarasan dan kestabilan perisian yang dipilih**

Semasa proses awal pembangunan dClean, perisian Microsoft Server 7.0 yang digunakan kurang sepadan atau sesuai dengan sistem pengendalian yang digunakan iaitu Windows 98. Oleh itu, sistem pengendalian yang digunakan kemudiannya ditukarkan kepada Windows 2000.

ii. **Kepakaran pada bahasa pengaturcaraan yang digunakan**

Visual Basic telah digunakan sebagai perisian yang digunakan bagi membentuk antaramuka pada sistem. Kurangnya pendedahan dan penguasaan pembangun sistem menggunakan perisian ini menyukarkan sedikit proses pembangunan **dCleanViewer**. Rujukan dilakukan pada buku-buku, halaman web-halaman web, forum-forum perbincangan pada intenet yang melibatkan bahasa pengaturcaraan tersebut, dan perbincangan dengan rakan-rakan yang berpengalaman dilakukan bagi menambah pengetahuan dan kemahiran pada bahasa pengaturcaraan yang dipilih.

iii. **Kekurangan sumber rujukan**

Memandangkan pembangunan sistem pembersihan data pada gudang data merupakan sesuatu yang masih baru, maka sumber rujukan agak sukar diperolehi. Malah masih banyak pihak yang tidak berkeyakinan terhadap pembangunan sistem sebegini, kebanyakan sistem pembersihan data yang wujud menghadkan jenis pangkalan data yang boleh dibersihkan. Selain itu, pembangunan sistem seperti ini memerlukan kepakaran dan pengalaman yang lebih meluas dalam penyelengaraan pangkalan data-pangkalan data.

iv. **Kurang pendedahan tentang penggunaan SQL Server 7.0**

Pembangunan **dCleanViewer** merupakan pengalaman pertama dan pendedahan pertama pembangun sistem kepada perisian SQL Server 7.0. Oleh itu, inisiatif bagi mempelajari tentang penggunaan perisian tersebut dilakukan berdasarkan maklumat yang diperolehi daripada buku rujukan dan melalui perbincangan dengan rakan-rakan yang berpengalaman.

8.4 CADANGAN SISTEM

Melalui pemerhatian, kajian literasi dan pengalaman terhadap pembangunan sistem yang di implimentasikan , beberapa cadangan dapat diberikan bagi meningkatkan lagi tahap kebolehgunaan sistem yang telah dibangunkan. Antaranya ialah :

i. **Menunjukkan langkah pembersihan pangkalan data**

dCleanViewer melibat pembersihan 2 buah pangkalan data yang telah ditetapkan. Ianya akan memaparkan data-data yang bersih dalam bentuk jadual kepada pengguna. Adalah dicadangkan pada masa hadapan, **dCleanViewer** dapat menunjukkan langkah-langkah pembersihan kepada pengguna.

ii. **Melakukan proses penukaran format pangkalan data dalam sistem yang dibangunkan.**

Pada ketika ini, pangkalan data yang menggunakan perisian Microsoft Acces perlu di '*import*" secara manual terlebih dahulu kedalam Micrososft SQL Server sebelum proses pembersihan menggunakan sistem yang dibangunkan dijalankan. Adalah dicadangkan pada masa akan datang **dCleanViewer** dapat diperbaiki dengan menambahkan fungsi penukaran format pangkalan data yang akan digabung dan dibersihkan.

iii. **Pembersihan pangkalan data secara lebih meluas**

Sebagai permulaan, **dCleanViewer** melakukan pembersihan pangkalan data pada pangkalan data-pangkalan data yang telah ditetapkan. Dicadangkan **dCleanViewer** dapat diperbaiki dengan membolehkan proses penyatuan dan pembersihan pangkalan data dapat dilakukan pada-pada mana-mana pangkalan data yang saling berkait untuk digabungkan dan diselaraskan.

iv. **Kepelbagaian teknik pembersihan**

Masalah data kotor yang wujud dari sumber yang pelbagai meliputi bukan sahaja pada data yang tidak konsisten, malah ianya turut melibatkan kehilangan data, data hingar dan sebagainya. Oeh itu, peluasan pada teknik pembersihan perlu dilakukan bagi membolehkan proses pembersihan data yang lebih efisyen dapat dilaksanakan.

8.5 KESIMPULAN

dCleanViewer merupakan suatu usaha permulaan dalam menghasilkan suatu perisian pembersihan pangkalan data yang diintegrasikan. Sebagai permulaan, **dCleanViewer** boleh membersihkan 2 buah pangkalan data yang saling berkait dalam format yang berbeza. Selain itu, ianya membenarkan pengguna membuat capaian pada pangkalan data Microsoft Access dan pangkalan data SQL Server. **dCleanViewer** juga membenarkan pengguna melaksanakan pertanyaan SQL pada pangkalan data yang telah dicapai. Ianya merupakan satu usaha dalam menghasilkan suatu perisian pembersihan data dan ianya boleh menjadi titik tolak permulaan kepada pembangun sistem ini untuk menghasilkan suatu sistem pembersihan data yang lebih baik, yang mana ianya tidak lagi menggunakan *hard coding* untuk melakukan proses pembersihan data. Walaupun **dCleanViewer** dibangunkan mengikut garis panduan skop yang telah ditetapkan, adalah penting bagi penganalisa data yang akan datang agar dapat setkan parameter yang lebih pelbagai dan seterusnya dapat membenarkan suatu aturcara yang lebih kompleks dihasilkan bagi mewujudkan suatu sistem atau perisian pembersihan data yang lebih efektif dan efisyen.

Didapati, kajian yang mendalam atau pembangunan tentang peralatan atau perisian yang membolehkan pembersihan data pada gudang data dengan teknik yang pelbagai di negara kita adalah kurang berbanding negara luar. Oleh itu, dengan adanya inisiatif latihan ilmiah yang telah dijalankan, diharapkan sedikit sebanyak dapat memberikan suatu anjakan paradigma terutamanya

kepada golongan pelajar dalam membangunkan suatu sistem yang kurang meluas implimentasinya (pembangunannya) tetapi amat penting penghasilan dan penggunaannya. Diharapkan pada masa akan datang lebih banyak sistem pembersihan data pada pangkalan data atau gudang data dapat dibangunkan.

BIBLIOGRAFI

[1] Sagent Technology, Inc,Centrus Merge/Purge Library (Qualitative Marketing Software).(2000). Diperolehi dari http://wwww.qmsoft.com/Merge.htm

[2] School of Computing, National University of Singapore, Clasification Based on Association (CBA), 1999. diperolehi dari http://www.comp.nus.edu.sg/~dm2/index.html

[3] helpIT Systems Limited, matchIT,2002. Diperolehi dari http://www.helpit.co.uk

[4] Peoplesmith, Inc, DoubleTake, StyleList and Personator, 1997-2002. Diperolehi dari http://www.peoplesmith.com.

[5] DataTools Pty Ltd, Data Tools Twins, 1999-2000. Diperolehi dari http://www.datatools.com.au

[6] Han,J.,Kamber M.,(2001). *Data Mining Concepts and Techniques*. 1st ed. Morgan Kaufmann Publishers.

[7] Pfleeger, S.L,(2001). *Software Engineering Theory and Practice*. 2nd ed. Prentice Hall International, Inc.

[8] Masrek, M. N., Safawi, A.R., and Kamarulariffin, A.J.,(2001). *Analisis & Rekabentuk Sistem Maklumat.* Mc Graw Hill.

[9] Galhardas H., Florescu D., Shasha D., Simon E., C. Saita A., (2001) Declarative Data Cleaning : Language, Model, and Algorithms. *Extended version of the VLDB '01 paper.*

[10] Laksmanan L.V.S., Sadri F., and Subranian I.N. (1999). SchemaSQL – A Language for Interoperability in Relational Multi-Database Systems. *In Proc. 26th VLDB, Mumbai.*

[11] Redmon N.A (1999) *Microsoft SQL Server & Database Implementation Training Kit.* Microsoft Press.

[12] Dupont Y. (1999). *Resolving Fragmentation Cinflicts in Schema Integration.* P. Loucopoulos Ed. LNC 881, Springer-Verlad German.

[13] Cristine P., Stefano S.,(1999). Issue and Approach of Database Integration. *IEEE Trans. Knowledge Data Eng. 4,1.*

[14] Rahm E. and Do H.H. (2000) data Cleaning : Problems and current Approaches.*IEEE Data Engineering Bulletin, 23 (3).*

[15] Sellapan P, (2000). *Software Engineering Management & Method*, Sejana Publishing.

[16] Kenneth and J. Kendall, (1998). System Analisys & Design, 1st ed, Prentice Hall International Inc.

[17] Sarawagi S, Vijay T. R.,(1999). *Cleaning Methods in Data Warehousing.* KR School of Information technology (ITT), Bombay.

[18] Larry Greenfield. (2002). *An (Informal) Taxanomy of Data Warehouse Data Errors.* Diperolehi dari http://www.dwifocenter.org/errors.html

[19] Pressman, R. S. (2001). *Software Engineering : A Practitioner's Approach.* 5th ed. Mc Graw Hill