

Studi Konversi Sintaks *Object Modeling Technique (OMT)* Ke Sintaks *Structure Query Language (SQL)* (Studi Kasus Sistem Pendaftaran Siswa Lembaga Bimbingan Belajar)

Yuni Hanafi^a

^aStaf Pengajar Fakultas Sains dan Teknologi
Universitas Islam Negeri Syarif Hidayatullah Jakarta
Tel : (021) 7493547 Fax : (021) 7493315
e-mail :

ABSTRAK

Penelitian ini adalah melakukan perancangan konversi model objek OMT ke model relasional dilanjutkan pembangkitan kode SQL dari skema relasi tersebut. Mekanisme konversi ini dirancang menggunakan aturan *Schemer*. Aturan *Schemer* adalah aturan konversi bentuk teks model logik menjadi perintah-perintah SQL yang diperlukan untuk membuat table-tabel relasional. Implementasi konversi OMT ke SQL diterapkan pada sistem pendaftaran siswa sebuah lembaga bimbingan belajar. Mulai dari analisis usecase diagram, class diagram, teks objek, normalisasi, menghasilkan tabel SQL. Metodologi berorientasi objek merupakan suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek-objek yang berisi data dan aksi yang dilakukan terhadapnya. Salah satu metodologi ini adalah metodologi objek Rumbaugh (*OMT / Object Modeling Technique*). Metodologi tersebut terdiri dari tiga buah model yaitu model objek, model dinamik, dan model fungsional. Dalam pengembangan perangkat lunak tertentu, sebagian objek yang digambarkan dalam skema model objek harus disimpan dalam memori sekunder dengan memanfaatkan sistem manajemen basisdata yang telah ada. Implementasi pengelolaan objek-objek tersebut dengan menggunakan sistem basisdata relasional memerlukan suatu bentuk konversi dari model objek OMT ke model relasional sehingga sampai deskripsi dalam SQL.

KATA KUNCI : *Object Modelling Technique, model relasional, Aturan Schemer, model objek, konversi.*

1. PENDAHULUAN

Suatu teknik atau cara pendekatan baru dalam melihat permasalahan dan sistem (sistem perangkat lunak. Sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata. Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dan abstraksi objek. kelas. hubungan antar kelas sampai abstraksi sistem. Saat mengabstraksikan dan memodelkan objek ini, data dan proses-proses yang dimiliki oleh objek akan dienkapsulasi (dibungkus) menjadi satu kesatuan. Dalam rekayasa perangkat lunak. konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis. perancangan. pemrograman, dan pengujian perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu.

Paradigma berorientasi objek berguna dalam perancangan sistem, sebab model objek dapat

ekspresif, ringkas mudah dikembangkan dan mudah untuk melakukan perubahan terhadap kekurangan dan kesalahan atribut pada teknik perancangan basisdata. Pemodelan relasional dapat dijadikan penunjang, sebab didukung oleh teknologi yang maju dan banyak tersedia komersil.

Relasional mungkin akan tetap berada dalam lingkungan aplikasi bisnis dan merupakan aplikasi yang handal. Selama OMT akan digunakan lebih banyak dalam rekayasa berbantuan komputer, perancangan dan aplikasi, secara perlahan perbedaan ini akan hilang. Bagaimanapun sebagai keistimewaan baru, seperti pengenalan pada konsep pewarisan (inheritance) dan penambahan tipe data dari relasional, serta penambahan fasilitas query yang handal dalam OMT. Aplikasi-aplikasi yang menggabungkan metode pengembangan berorientasi objek dengan implementasi relasional dapat lebih bermanfaat daripada menggunakan suatu pendekatan perancangan secara terpisah. Cara tersebut membuat para ahli akan bebas untuk memusatkan perhatiannya pada tahap perancangan untuk tugas-tugas tingkat rendah seperti perancangan basisdata. menawarkan keuntungan-keuntungan unik, teknologi

distandarisasikan, dan mempunyai kemampuan untuk bekerja dengan suatu bahasa deklaratif. Sistem manajemen basisdata juga mempunyai keuntungan-keuntungan sendiri. Seperti pengendalian struktur data yang cepat, mendukung tipe data berukuran besar (seperti audio dan video), dan berkemampuan untuk berintegrasi dengan paling sedikit satu bahasa. Bagaimanapun relasional adalah lebih sederhana dibanding network dan hirarki, jadi banyak pengembang harus tetap menemukan kelebihan atau manfaat objek untuk menciptakan perangkat-perangkat dan aplikasi relasional. Sebagai suatu kekuatan yang saling melengkapi antara relasional dan objek harus dibuat keduanya supaya berjalan secara berdampingan dalam beberapa bentuk aplikasi. Meskipun keduanya mungkin berfungsi pada aplikasi berbeda. Relasional

Kebanyakan sistem basisdata menggunakan model relasional, sistem basisdata berorientasi objek masih sedikit masih dalam proses penelitian dan pengembangan. Dengan demikian relasional merupakan salah satu pilihan untuk mengimplementasikan pengelolaan data. Maka diperlukan suatu konversi model objek OMT ke model relasional.

1.1 Rumusan Masalah

1. Bagaimana merubah metodologi perancangan sistem dari model objek OMT (Object Modeling Technique) ke model SQL (structure Query Language)?
2. Bagaimana penerapan konversi OMT ke SQL ke dalam sistem informasi pendaftaran siswa sebuah lembaga bimbingan belajar?

1.2 Tujuan Penelitian

Sesuai dengan permasalahan yang diangkat, tujuan penelitian ini adalah sebagai berikut :

1. Mengkaji cara perubahan metodologi perancangan sistem dari model objek OMT (Object Modeling Technique) ke model SQL (structure Query Language) secara rinci dan jelas.
2. Merancang sistem informasi pendaftaran siswa sebuah lembaga bimbingan belajar

2. LANDASAN TEORI

2.1 Model Objek

Metodologi berorientasi objek merupakan suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek-objek yang berisi data dan aksi yang dilakukan terhadapnya. Salah satu metodologi ini adalah metodologi objek OMT (Object Modelling Technique). Tujuannya adalah untuk mengidentifikasi secara jelas aspek-aspek metodologi dan penerapannya terhadap metodologi yang akan

dibangun. Komponen objek adalah : Objek, Kelas, Diagram-objek, atribut, Operasi dan metoda, Link dan Asosiasi, Kualifikasi, Agregasi, Generalisasi dan Pewarisan dan Metadata.

2.2 Model Relasional

Bertujuan untuk menghasilkan sekumpulan skema relasi dalam bentuk yang tepat (normal), menghindari redundansi dan mengakses informasi dengan mudah. Elemen dasar dari model relasional adalah relasi yaitu instans dari skema relasi yang berbentuk table dan terdiri atas baris dan kolom. Baris dari relasi disebut tuple, adalah kumpulan dari nilai yang dimiliki tiap atribut. Degree dari suatu relasi adalah jumlah kolom dari relasi, sedangkan kardinalitas adalah jumlah baris dari relasi. Skema model relasional adalah kumpulan skema relasi, skema relasi adalah gabungan atribut-atribut. Himpunan dari semua nilai yang mungkin dari atribut tertentu disebut domain atribut tertentu. Komponen model relasional adalah ketergantungan fungsional, relationship, key, normalisasi.

2.3 Metode Pembangun Perangkat Lunak

Metode pembangun perangkat lunak yang digunakan adalah metode Waterfall (Life cycle) lebih jelasnya :

1. **Pendekatan sistem** meliputi spesifikasi kebutuhan perangkat lunak yang dapat mewakili dunia nyata.
2. **Analisis** adalah suatu proses yang menghasilkan spesifikasi perangkat lunak yang sesuai dengan kebutuhan sistem.
3. **Perancangan** adalah mendefinisikan struktur data, arsitektur perangkat lunak yang mendefinisikan relasi diantara komponen-komponen struktur program dan suatu prosedur yang bertujuan untuk permmumberaan data.
4. **Pengkodean** adalah hasil perancangan harus ditranslasikan ke bentuk yang bisa dibaca oleh mesin (komputer).
5. **Ujicoba** meliputi pengujian kebutuhan program, validasi program dan modularitas perangkat lunak.
6. **Perawatan** dilakukan sejalan dengan penggunaan perangkat lunak yang dibuat.

3. METODE PENELITIAN

Penelitian ini menggunakan metode studi komparatif yaitu metode perbandingan mengenai pemograman model objek modeling technique (OMT) dengan pemograman relasional / SQL.

3.1 Perbandingan Struktur Model Relasional dan Model objek OMT

Model data terdiri dari atas tiga komponen utama yaitu aturan pembentuk struktur, aturan pembentuk

batasan dan operasi. Kebanyakan model data memiliki spesifikasi pembentuk aturan yang jelas dan telah tertentu (tetap) terutama pada model data yang diadopsi sistem manajemen basisdata.

Berbeda dengan struktur, aturan pembentuk batasan dan operasi bersifat lebih fleksibel. Bahkan dalam model data yang terdefinisi dengan baik secara formal seperti model relasional masih dapat dikembangkan operasi-operasi dan batasan-batasan baru sesuai dengan kebutuhan aplikasi. Batasan dan operasi dibuat dengan mengacu kepada struktur, dengan demikian yang paling mendasar dalam model data adalah pembentuk strukturnya (komponennya).

Sebagai dasar dari perancangan konversi maka dalam bagian ini akan dibahas bagian terpenting dari kedua model yaitu struktur model relasional dibandingkan dengan struktur model objek OMT mulai dari uraian secara umum dan dilanjutkan dengan lebih detail. Perbandingan ini dilakukan dengan tujuan untuk mengetahui apa yang dimiliki oleh model objek OMT dan yang tidak dimiliki oleh model relasional sehingga dalam perancangan konversi dapat dicari cara untuk mengatasi perbedaan tersebut.

Perbandingan komponen-komponen model objek OMT dan model relasional diperlihatkan dalam table berikut :

Table 1. komponen-komponen model objek OMT dan model Relasional

Komponen model objek OMT	Komponen model relasional
Kelas {atribut, operasi, descriptor}	Skema relasi {atribut}
Agregasi	Tidak ada
Asosiasi	Tidak ada
Generalisasi	Tidak ada

Dari table diatas dapat dilihat bahwa struktur dari skema model relasional sangat sederhana, dibentuk dari kumpulan skema relasi. Skema relasi tersebut dibentuk dari atribut-atribut dengan domain atomik dan skalar. Model objek OMT memiliki pembentuk struktur lebih kompleks, dibangun oleh kelas-kelas yang memiliki atribut, operasi dan deskriptor kelas. Kelas-kelas dalam objek OMT dapat memiliki hubungan antar satu kelas dengan yang lainnya dengan adanya asosiasi, agregasi, dan generalisasi. Skema relasi dalam model relasional sama dengan kelas dalam model objek OMT yang tidak memiliki operasi dan descriptor kelas sehingga bisa dikatakan bahwa pembentuk struktur dari model relasional adalah bagian dari pembentuk struktur model objek OMT.

Pada uraian berikut ini komponen-komponen tersebut akan dibandingkan secara lebih detail. Pembentuk struktur yang dimiliki model relasional, skema relasi merupakan alat pengelompokan objek-objek. Dalam perbandingan berikut ini skema relasi akan dibandingkan dengan komponen pengelompokan dalam model objekOMT yaitu kelas dan asosiasi.

3.2 Kelas (Objek) dan Skema Relasi (relasional)

Skema relasi terdiri atas atribut-atribut sedangkan kelas selain dibentuk dari atribut-atribut juga memiliki operasi dan descriptor kelas. Atribut-atribut dalam skema relasi bersifat atomik dan skalar, hal ini berbeda dengan atribut-atribut dalam kelas yang tidak harus skalar atau atomik.

Instans dari model objek OMT, objek memiliki identitas yang membedakan satu objek dengan objek lainnya sedangkan tupel dari model relasional tidak memiliki identitas. Identifikasi tupel bukan berdasarkan eksistensinya, seperti pada objek, melainkan berdasarkan sifat/karakteristiknya yang diwakili oleh nilai atribut-atribut kuncinya.

Perbedaan lain adalah antara satu kelas dengan kelas yang lain dapat memiliki hubungan pewarisan. Dalam model relasional pewarisan antara satu skema relasi dengan yang lainnya tidak dikenal.

Ringkasan perbandingan keduanya dapat dilihat dalam table berikut :

Tabel 2 Perbandingan kelas (Objek) dengan skema relasi (Relasional)

Sifat/Karakteristik	Kelas	Skema Relasi
Atribut	ya	Ya
Domain atomik dan skalar	tidak harus	Harus
Operasi	ya	Tidak
Deskriptor kelas	ya	Tidak
Mekanisme Pewarisan	ya	Tidak

3.3 Asosiasi dengan Skema Relasi

Asosiasi dapat dianggap sebagai kelas khusus yang mengelompokkan objek-objek dengan fungsi tambahan yang tidak dimiliki oleh kelas biasa yaitu menghubungkan objek-objek dari kelas lain. Asosiasi adalah kelas dengan kelebihan tertentu yaitu instannya, link, memiliki pointer yang menunjuk ke objek-objek yang dihubungkannya. Perbedaannya dengan skema relasi adalah seperti perbedaan skema relasi dengan kelas dengan tambahan bahwa tupel skema relasi tidak memiliki pointer seperti pada link dan kelas asosiasi tidak mengenal pewarisan.

4. HASIL DAN PEMBAHASAN

4.1 Perbandingan Kedua Model

Dari perbandingan-perbandingan yang dimasukkan dalam metode penelitian bahwa skema relasi adalah suatu konsep yang lebih sederhana dan merupakan bagian dari konsep kelas dan asosiasi. Model objek OMT merupakan model yang lebih kaya dan kompleks dibanding model relasional. Hal ini berarti bahwa model objek OMT lebih ekspresif dibanding model relasional, artinya model objek OMT dapat menyatakan suatu keadaan data dengan lebih tepat dan jelas dibanding model relasional.

Dalam setiap pernyataan struktur dunia nyata yang dinyatakan dengan skema model objek OMT (tanpa batasan) terkandung batasan-batasan implisit yang tidak dimiliki jika hal dinyatakan dengan skema model relasional yang tanpa tambahan batasan. Misalkan pada suatu skema model objek OMT ada asosiasi belajar yang menghubungkan antara kelas siswa dan sekolah. Dalam asosiasi tersebut terdapat batasan implisit yang mengharuskan keberadaan objek siswa dan objek sekolah sebelum adanya link belajar. Hal ini tidak bisa diwakilkan oleh skema relasi belajar yang tidak dilengkapi batasan lain.

Hal yang sangat penting dari perbandingan antara model objek OMT dan model relasional adalah perbedaan karakteristik atribut pada kelas dan skema relasi. Karakteristik atribut dalam skema relasi yang lebih terbatas domainnya membuat tidak mungkin untuk dapat merepresentasikan semua atribut-atribut pada kelas dengan atribut-atribut skema relasi. Nilai

dari atribut yang tidak skalar atau atomik tidak mungkin direpresentasikan oleh nilai atribut dari suatu relasi.

Oleh karena itu dalam perancangan konversi ini model relasional yang akan digunakan adalah model relasional yang diperluas dengan membolehkan atribut yang domainnya tidak atomik atau skalar. Model relasional ini disebut model relasional+. Pada bagian-bagian berikutnya dalam pembahasan yang berhubungan dengan konversi, model relasional yang dimaksud adalah model relasional+. Penggunaan nama model relasional+ hanya dilakukan dalam pembahasan yang menyangkut hal-hal yang dimiliki oleh model relasional+.

Model relasional+ merupakan perluasan dari Model relasional dengan memperbolehkan domain atribut yang tidak bersifat skalar atau atomik. Pada Model relasional yang murni setiap domain harus atomik dan skalar. Atomik berarti sistem tidak mengetahui struktur numberernal setiap elemen domain. Skalar berarti antara satu elemen domain dengan elemen domain yang lain dapat ditentukan suatu urutan tertentu seperti urutan lebih kecil-lebih besar atau lebih awal-lebih akhir. Perluasan ini dimaksudkan agar model relasional dapat menyesuaikan dengan karakteristik atribut dalam model objek OMT.

Sebenarnya model relasional+ masih dapat dikatakan sebagai model relasional karena kenyataannya banyak sistem basis data yang melanggar definisi domain tersebut tetapi masih dikatakan menggunakan model relasional.

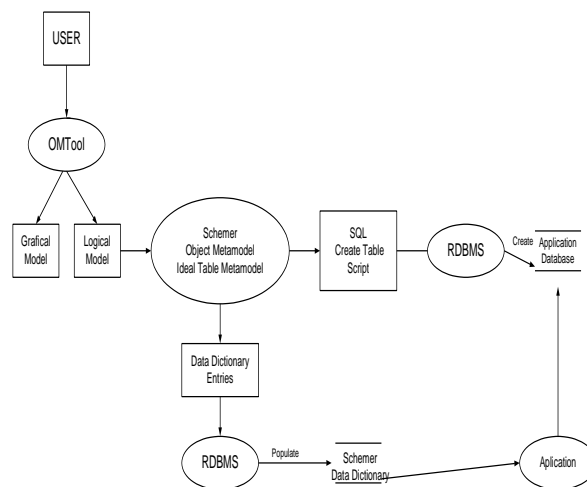
Pelanggaran terhadap sifat atomik yang umumnya dilakukan adalah dengan adanya tipe domain tanggal (DATE). Sistem-sistem basisdata yang memiliki tipe tanggal mengetahui bahwa dalam elemen domain tanggal terdiri atas hari, bulan dan tahun, hal ini bertentangan dengan sifat atomik. Pelanggaran terhadap sifat skalar yang umum adalah adanya tipe domain Boolean. Elemen-elemen domain Boolean tidak bersifat skalar, kita tidak dapat menentukan apakah nilai "benar" lebih besar atau lebih kecil dari nilai "salah".

Salah satu konsekuensi dari perluasan ini adalah dimungkinkannya domain bertipe prosedural.

Domain ini dapat merepresentasikan metoda pada model objek OMT. Domain prosedur memiliki karakteristik khusus yaitu operator yang berlaku bagi elemen domain ini. Hal ini disebabkan nilai-nilai prosedur bukanlah nilai-nilai yang bersifat skalar. Akibatnya hanya operator perbandingan sama dengan (“=”) dan tidak sama dengan (“#”). Lain dari nilai-nilai yang tidak skalar ini adalah atribut prosedur tidak dapat didefinisikan sebagai bagian dari kunci dalam relasi.

4.2 Aturan Schemer

1. Konversi OMT ke Model Relasional



Gambar 1. Konversi model OMT ke model relasional

2

. Komponen Konversi

a. OMTool

OMTool menyimpan dua pandangan terhadap model objek dalam notasi pemodelan OMT. Model grafik, menggambarkan bentuk kotak, garis, dan teks ke layar. Model Grafik digunakan untuk menambah antarmuka dengan pemakai dan menyajikan model objek secara visual untuk dokumentasi. Model logik merupakan ringkasan tentang pengertian yang mendasar pada gambar dan menjelaskan kelas-kelas dan keterhubungannya antara yang satu dengan lainnya.

OMTool memiliki beberapa keistimewaan untuk membuat pembangkit model yang termudah. OMTTool memungkinkan pemakai untuk membuat, memuat, mengubah, menyimpan dan mencetak diagram secara otomatis. OMTTool mengelola keterhubungan logik seperti pemakai memindahkan entitas yang terhubung.

Antarmuka disajikan dalam bentuk jendela-jendela melayang. Untuk melakukan pertanyaan secara detail

Suatu percobaan arsitektur yang dilakukan James Rumbaugh pada GE Corporate R & D, dan Schemer, yaitu batch compiler yang dikembangkan untuk membangkitkan kode SQL dengan mengotomatisasikan aturan-aturan pemetaan melalui suatu model kompilator.

Gambar 1 menunjukkan arsitektur untuk mengoptimasikan konversi model objek ke model relasional. Arsitektur memiliki tiga bagian utama : OMTool, Schemer dan RDBMS itu sendiri. Schemer adalah kumpulan aturan-aturan pemetaan objek dan metamodel table-tabel secara formal dan notasi pemetaan pemodelan ke skema basisdata.

tidak ditunjukkan pada diagram seperti lingkup atribut, pengijinan nilai kosong untuk setiap atribut & kunci-kunci primer. Juga jendela untuk menentukan aturan pemetaan yang akan diaplikasikan pada semua kelas dan relationship dalam model, yang mana secara mendasar menyediakan strategi global untuk implementasi basisdata. Pemakai dapat mengesampingkan aturan-aturan pemakai global untuk kelas-kelas khusus atau relationship pada hasil efisiensi, jangkauan perluasan dan numberegritis dalam kode relasional.

Ketika pemakai membuat keputusan implementasi, OMTTool menuliskannya dan model logik pada suatu file ASCII sebagai masukan untuk kompilator skema basisdata (dalam kasus ini shemer).

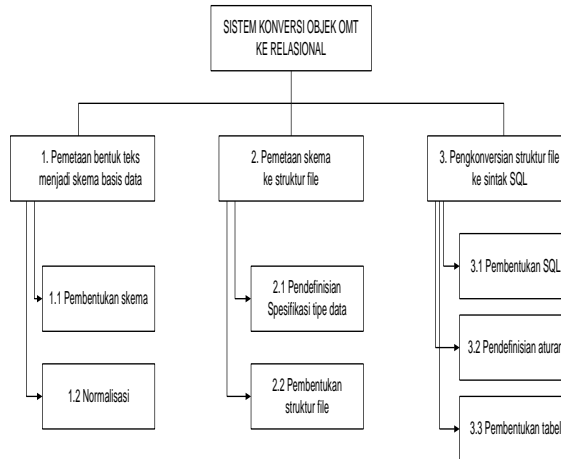
b. Schemer

Schemer mengkonversikan bentuk teks menjadi perintah-perintah SQL yang diperlukan untuk menciptakan table-tabel relasi, index, domain, hak akses dan bagian-bagian lain pada skema. Selama

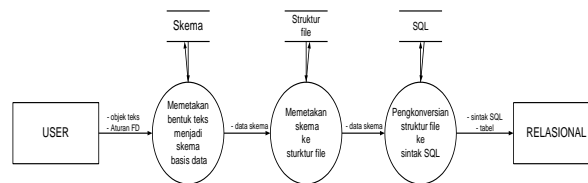
proses ini, Schemer mengeluarkan suatu pesan kesalahan seperti nama terlalu panjang atau informasi yang hilang. Schemer juga memeriksa batasan-batasan yang digunakan untuk pewarisan ganda, identitas yang diturunkan dan tipe data yang dihasilkan serta memperingatkan jika dipilih aturan pemetaan yang tidak konsisten.

3. Proses Konversi

Schemer mengkonversikan model objek menjadi table-table relasi dalam tiga tahap, untuk lebih jelasnya dapat dilihat pada struktur proses dan diagram aliran data.

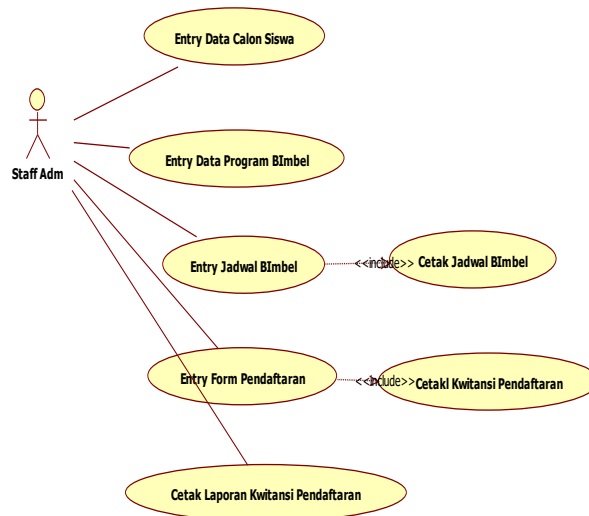


Gambar 3.1 Struktur Proses



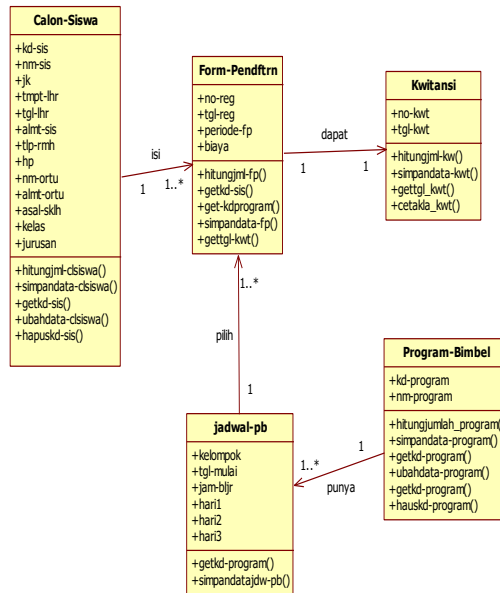
Gambar 3.2. Diagram Aliran Data

a. Contoh Schemer Konversi Model Objek menjadi Tabel Relasional



Gambar 3.2.1 Diagram Usecase

Contoh kasus konversi Sistem Pendaftaran Siswa Baru Lembaga Bimbingan Belajar



Gambar 3.2.2 Contoh kasus konversi Sistem Pendaftaran Siswa Baru Lembaga Bimbingan Belajar

Gambar diatas adalah salah satu contoh dari fungsi Schemer yaitu proses konversi dari model teks objek menjadi sintaks relasional sql. Contoh yang diterapkan adalah kasus konversi Sistem Pendaftaran Siswa Baru Lembaga Bimbingan Belajar. Dimana Sistem Pendaftaran Siswa Baru Bimbel tersebut menjelaskan beberapa objek yang terkait diantaranya objek calon siswa mengisi banyak objek form pendaft dan mendapat objek kwitansi, objek program Bimbel mempunyai banyak objek Jadwal Program Bimbel. Schemer mengkonversikan gambar model objek Sistem Pendaftaran Siswa Baru Lembaga Bimbingan Belajar diatas menjadi sintaks sql dengan beberapa tahapan sebagai berikut :

1. Batasan, Deskripsi dengan Algoritma dalam sintaks teks objek

A. Class dan object description

Class Calon Siswa

- Attribute Kd-sis:number
- Attribute nmsis:string
- Attribute jk:string
- Attribute tmpt-lhr:string
- Attribute tgl-lhr:date
- Attribute almt-sis:string
- Attribute Telp:number
- Attribute HP:number
- Attribute nmortu:string
- Attribute almt-ortu:string
- Attribute asal-sklh:string
- Attribute : kelas:string
- Attribute : jurusan:string
- Attribute AssFormPendaftaran:FormPendaftaran

```

OperationSet_Kdsis,nmsis,jk,tmptlhr,tglhr,almtsis,tlprmh,h
p,nmortu,almtortu,asalsklh,kelas,jurusan(PKdsis,Pnmsis,Pj
k,Ptmptlhr,Ptglhr,Palmtsis,Ptlprmh,Php,Pnmortu,Palmt-
ortu,Pasal-sklh,Pkelas,jurusan)
OperationSet_AssFormPendaftaran(Pno-reg,Ptgl-
reg,Pperiode-fp,Pbiaya)
End;
    
```

Class Form Pendaftaran

- Attribute no-reg:number
- Attribute tgl-reg:date
- Attribute periode-fp:string
- Attribute biaya:number
- Attribute AssKwitansi:Kwitansi
- Attribute AssCalonSiswa :CalonSiswa
- Attribute AssJadwalPB:JadwalPB
- Operation Set_no-reg,tgl-reg,periode-fp,biaya,(Pno-
reg,Ptgl-reg,Pperiode-fp,Pbiaya)
- OperationSet_AssKwitansi(Pno-kwt,Ptgl-kwt)
- OperationSet_AssCalonSiswa(PKdsis,Pnmsis,Pjk,Ptmptlhr,
Ptglhr,Palmtsis,Ptlprmh,Php,Pnmortu,Palmt-ortu,Pasal-
sklh,Pkelas,jurusan)
- OperationSet_AssJadwalPB(Pidkel,Ptglmulai,Pjambelajar
,Phari1,Phari2,Phari3)
- End;

Class Kwitansi

- Attribute no-kwt:number
- Attribute tgl-kwt:date
- Attribute AssFormPendaftaran
- Operation Set_no-kwt,tgl-kwt (Pno-kwt,Ptgl-kwt)

```

OperationSet_AssFormPendaftaran(Pno-reg,Ptgl-
reg,Pperiode-fp,Pbiaya)
End.
Class JadwalPB
Attribute idkel:number
Attribute Tglmulai:date
Attribute jambelajar:integer
Attribute hari1t:string
Attribute hari2t:string
Attribute hari3t:string
AttributeAssFormPendaftaran:FormPendaftaran
AttributeAssProgramBimbel:ProgramBimbel
OperationSet_idkel,tglmulai,jambelajar,hari1,hari2,hari3(
Pidkel,Ptgmulai,Pjambelajar,Phari1,Phari2,Phari3)
OperationSet_AssFormPendaftaran(Pno-reg,Ptgl-
reg,Pperiode-fp,Pbiaya)
OperationSet_AssProgramBimbel(Pkdprogrambimbel,Pn
amaprogrambimbel)
End;

```

```

Class ProgramBimbel
Attribute kdprogrambimbel:number
Attribute namaprogrambimbel:string
Attribute AssJadwalBimbel:JadwalBimbel
OperationSet_kdprogrambimbel,namaprogrambimbel(Pk
dprogrambimbel,Pnamaprogrambimbel)

```

Kd- sis	Nm- sis	Jk	Tmpt- lhir	tgllhr	almtsi s	telp
------------	------------	----	---------------	--------	-------------	------

Form Pendaftaran (bentuk normal)

No-reg	tglr	Per- fp	biaya	Kd- sis	idkel	kdPB
--------	------	------------	-------	------------	-------	------

Kwitansi (bentuk normal)

No-kwt	Tgl-kwt	No-reg
--------	---------	--------

Jadwal Program Bimbel (bentuk normal)

Id- kel	tgmlu ai	jamb l jr	Har i1	Har i2	Har i3	kdP B
--------------------	---------------------	----------------------	-------------------	-------------------	-------------------	------------------

Program Bimbel (bentuk normal)

kdprogrambimbel	nmprogrambimbel
-----------------	-----------------

3.

Pembentukan SQL

Tahap ini adalah hasil akhir dari fungsi schemer (mengkonversi teks model objek ke sintaks sql). Dimana struktur file yang terbentuk dari tahapan ke dua (skema relasi) dirubah ke sintaks sql dengan memperhatikan aturan kunci tabel yang terkait.

1) Create Table Calon Siswa

```

((kd-sis Varchar[15] Notnull Default,
Nm-sis Varchar[30] Notnull Default
'none'),
Jk Varchar[10] Notnull Default,

```

```

OperationSet_AssJadwalPB(Pidkel,Ptgmulai,Pjambelajar
,Phari1,Phari2,Phari3)
End;

```

2. Pembentukan skema relasi ke struktur file

Tahap ini adalah proses ke dua fungsi schemer (mengkonversi teks model objek ke sintaks sql). Dimana teks model objek yang terbentuk dirubah ke struktur file / skema relasi.

a. Diperoleh deskripsi skema relasi sebagai berikut :

1. CalonSiswa(kdsis,nmsis,jk,tmptlhr,tgllhr,almtsis,telprmh)
2. FormPendaftaran(no-reg,tgl-reg,periodefp,biaya)
3. Kwitansi(no-kwt,tgl-kwt)
4. JadwalProgramBimbel(idkel,tglmulai,jambelajar,hari1,hari2,hari3)
5. ProgramBimbel(kdprogrambimbel,nmprogrambimbel)

b. Normalisasi

Calon Siswa (bentuk normal)

```

tmptlhr Varchar[35] Notnull Default
tgllhr Varchar[11] Notnull,
almtsis Varchar[35] Notnull Default
telprmh Varchar[11] Notnull,
Primary key (kd-sis)
)

```

2) Create Table Form Pendaftaran

```

((no-reg Varchar[11] Notnull Default,
Tgl-reg Varchar[11] Notnull Default
'none'),

```


Periode-fp Varchar[20] Notnull Default,
 biaya Varchar[25] Notnull Default
 kd-sis Vardate Notnull Default
 idkelompok Vardate Notnull Default
 kdprogram Vardate Notnull Default
 Primary key (no-reg)
 Foreign key (kd-sis)
 Foreign key(idkelompok)
 Foreign key(kdprogram)
 Reference kd-sis to Calon Siswa(kd-sis)
 Reference idkelompok to JadwalPB
 (idkelompok)
 Reference kdprogram to
 ProgramBimbel(kdprogram)
)

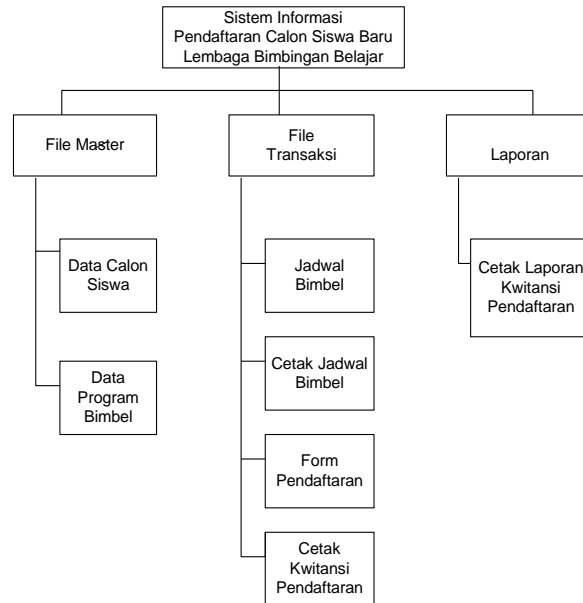
3) Create Table Kwitansi
 ((no-kwt Varchar[11] Notnull Default,
 Tgl-kwt Varchar[30] Notnull Default
 'noname',
 no-reg Vardate Notnull Default
 Primary key (no-kwt)
 Foreign key (no-reg)
 Reference no-reg to FormPendaftaran (no-
 reg)
)

4) Create Table Jadwal Program Bimbel

((id-kel Varchar[11] Notnull Default,
 Tgl-mulai Varchar[30] Notnull
 Default 'noname',
 Jambelajar Vardate[11]Notnull Default
 Hari1 Vardate[11]Notnull Default
 Hari2 Varchar[15] Notnull,
 Hari3 Varchar[11] Notnull Default
 Kd-program Varchar[15] Notnull,
 Primary key (id-kel)
 Foreign key (kd-program)
 Reference kdprogram to
 ProgramBimbel(kdprogram)
)

5) Create Table Program Bimbel
 ((kdprogrambimbel Varchar[10]
 Notnull Default,
 nmprogrambimbel Varchar[50] Notnull
 Default,
 Primary key (kdprogrambimbel),
)

4. Contoh Rancangan antar Muka kasus sistem perpustakaan



5. KESIMPULAN DAN SARAN

1. Kesimpulan

a. Tool pendukung konversi model objek ke relasional adalah Schemer yaitu komponen penting dalam konversi sistem ini. Konversi yang tidak

memenuhi aturan schemer ini dapat menimbulkan basisdata yang tidak konsisten. Konversi dengan menggunakan schemer dilakukan secara otomatis oleh suatu perangkat lunak yang menggunakan algoritma yang telah dirancang. Hal ini yang dimungkinkan karena skema yang dihasilkan sudah memenuhi kebutuhan data. Tetapi jika dikehendaki kinerja yang lebih baik maka perbaikan secara teratur harus dilakukan.

b. Untuk lingkungan implementasi tertentu dimungkinkan perbaikan model relasional hasil konversi untuk mencapai peningkatan dalam efisiensi ruang penyimpanan dan kecepatan akses data. Modifikasi harus tetap memelihara terpenuhinya aturan konversi.

2. Saran-saran

a. Dosen perlu mensosialisasikan atau mengajak kepada mahasiswa tentang kemudahan dalam pemograman objek dibandingkan pemograman lainnya. Sehingga mahasiswa bisa lebih termotivasi untuk menggunakan pemograman objek.

b. Dosen perlu sinergi untuk menyediakan modul perkuliahan berbasis pemograman objek untuk bahan perkuliahan mahasiswa.

c. Diharapkan kepada para pecinta jurnal, baik itu dosen, praktisi, peneliti maupun mahasiswa diharapkan untuk mau melakukan penelitian-penelitian yang bersifat studi perubahan atau konversi ilmu. Mudah-mudahan tulisan ini bermanfaat dan harapan penulis bisa menjadi bahan untuk pengembangan penelitian yang lebih optimal.

DAFTAR PUSTAKA

- [1]. Adi Nugroho (2004). Pemograman Berorientasi Objek. Informatika Bandung.
- [2]. Bambang Hariyanto (2004). Rekayasa Sistem Berorientasi Objek. Informatika Bandung.
- [3]. _____Pemodelan (2004), Perancangan dan Terapan Basisdata. Informatika Bandung.
- [4]. Derek Coleman (1994). Object Oriented Development The Fusion Method. Prentice Hall Numberernational Editions.
- [5]. Gio Wiederhold (1993). Database Design.Mc Graw Hill.
- [6]. Henry F Kort & Abraham S (1996). Database System Consepts. Mc Graw Hill.
- [7]. Roger S Pressman, Ph D (1992). Software Engineering. A Practitioner's Approach, Mc Graw-Hill Singapore.
- [8]. Rumbaugh (1991). Object Oriented Modelling and Design. Prentice Hall NewJersey, J. et al.

[9]. Yuni (2005). Memahami Algoritma Pemograman. Jurnal Teknodik Kemendepdiknas. no 16/1X/TEKNODIK/JUNI/2005.

[10]. Yuni (2009). Studi Komparatif Pemogram Objek dan SQL. Jurnal Teknodik Kemendepdiknas. no 23/XII/TEKNODIK/Okt/2009.