

PERMODELAN DATA

1. Pengertian Permodelan Data

Satu cara untuk membuat struktur masalah yang tidak berstruktur ialah dengan cara membuat model. Model merupakan gambaran realitas sebuah aktivitas. Model dibuat untuk memahami sistem dengan mudah atau mendokumentasikan spesifikasi bisnis/desain teknis bagi sistem yang sedang dikembangkan. Terdapat dua konsep model, yaitu model logikal dan model fisik.

Model Logikal menunjukkan apa yang dilakukan oleh sistem. Model logikal tidak tergantung kepada pelaksanaan/aspek teknis. Sedangkan model fisik menunjukkan apa yang dibuat oleh sistem dan bagaimana sistem berjalan secara fisik dan teknis. Model fisik tergantung kepada pelaksanaan karena model ini menunjukkan teknologi yang dipilih serta batasannya.

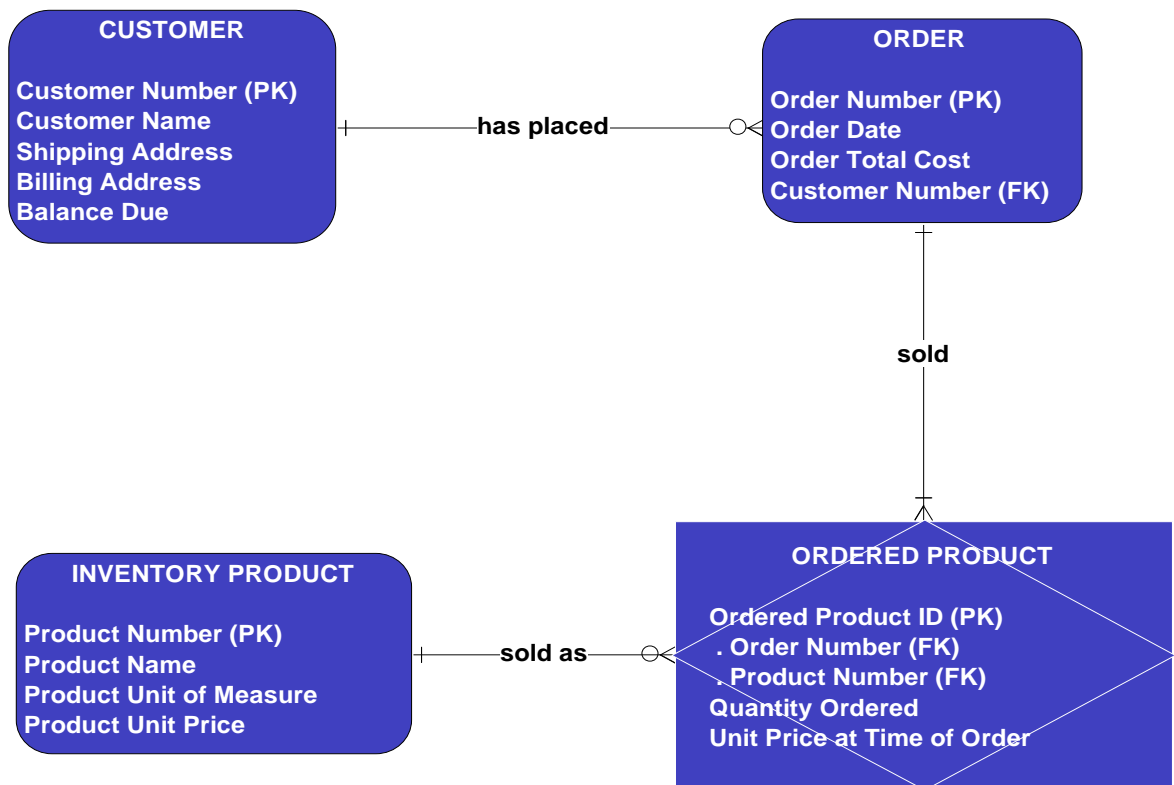
Sistem analisis menggunakan model logikal untuk menunjukkan kebutuhan bisnis dan model fisik untuk menunjukkan desain teknis. Aktivitas analisis sistem terfokus kepada model sistem logikal karena :

- Model logikal tidak bias kepada cara sistem pada saat yang dilaksanakan atau cara seseorang individu memikirkan bagaimana sistem harus dilaksanakan.
- Model logikal mengurangi risiko kebutuhan bisnis terlupakan yang disebabkan oleh terlalu memberi fokus kepada ciri-ciri teknis.
- Model logikal bisa berkomunikasi dengan pengguna akhir dalam bahasa bukan/kurang teknis.

Permodelan data adalah suatu teknik untuk menyusun dan mendokumentasikan data sistem. Permodelan data bisa dipanggil juga dengan istilah permodelan pangkalan data karena model data biasanya dilaksanakan sebagai pangkalan data. Atau biasa juga disebut dengan permodelan informasi.. Permodelan data dianggap penting karena data dilihat sebagai sumber yang perlu dibagi-bagikan dengan

menggunakan proses sebanyak-banyaknya. Oleh itu, data mesti dikelola dengan cara yang fleksibel dan sesuai dengan berbagai kebutuhan bisnes.

Struktur data dan ciri-cirinya biasanya tetap dibandingkan dengan pemrosesan datanya sendiri. Biasanya model data sistem hampir sama dengan sistem yang akan dikembangkan. Model data lebih kecil dari model proses dan objek dan bisa dikembangkan dengan lebih cepat. Proses pengembangan model data membantu analis sistem dan pengguna memahami istilah dan peraturan bisnes dengan cepat.



Gambar 1 : An Entity Relationship Data Model

2. Konsep Sistem dalam Permodelan Data

Kebanyakan teknik analisis sistem dimulai dari pemikiran sistem yang merupakan aplikasi teori dan konsep sistem formal kepada penyelesaian masalah. Permodelan data yang benar-benar memodelkan data dalam suatu sistem disebut Diagram Hubungan Entiti (Entity Relationship Diagram) yaitu menggambarkan data dari sudut entiti dan hubungannya yang diterangkan oleh data.

Entiti

Semua sistem memiliki data yang menerangkan 'benda'. Konsep yang mewakili semua ciri-ciri 'benda' dalam kelompok yang sama disebut entiti, seperti: individu, tempat, objek, peristiwa, pelajar, pelanggan, sekolah, produk, mata kuliah, pendaftaran. Ada juga yang disebut dengan entiti tunggal (entity instance) seperti: Afiffah, Irfan dan Ariffin.

Atribut

Semua ciri yang menerangkan entiti disebut atribut (*element, property, field*) seperti: Pelajar (nama, NIM, jurusan, fakultas). Ada beberapa atribut bisa diletakkan dalam kumpulan *super-atribut* yang dipanggil dengan atribut campuran (*concatenated attribute, composite attribute, data structure*).

Domain

Nilai setiap atribut didefinisikan dalam tiga kategori, yaitu: jenis data, domain dan default.

- ❖ Jenis Data: atribut mendefinisikan kelas data yang bisa disimpan dalam atribut tersebut, seperti: aksara, numerik. Yang perlu diperhatikan bahwa jenis data atribut dapat menentukan domainnya.
- ❖ domain : atribut mendefinisikan nilai/ukuran atribut, seperti: 10, 20
- ❖ Nilai default atribut adalah nilai yang perlu disimpan oleh atribut jika atribut tersebut tidak ditentukan oleh pengguna.

Setiap entiti mempunyai banyak instances, oleh karena itu perlu diketahui setiap instance berdasarkan nilai data dari satu atau lebih atribut. Setiap entiti mesti mempunyai satu kunci/ identifier yaitu satu/sekumpulan atribut yang memegang nilai

unik untuk setiap instance entiti. Kadangkala lebih dari satu atribut diperlukan untuk mengetahui instance entiti yang unik.

Sekelompok atribut yang mengetahui instance entiti disebut kunci komposit atau **concatenated key**. Setiap entiti bisa mempunyai lebih dari satu kunci. Setiap atribut yang bisa menjadi kunci dipanggil sebagai calon kunci. yaitu ‘calon yang akan menjadi kunci utama’. Kunci utama yaitu calon kunci yang akan sering digunakan untuk mengetahui instance entiti. Sedangkan calon kunci yang tidak dipilih sebagai kunci utama disebut kunci pilihan.

Hubungan

Setiap entiti berhubungan/memberi kesan di antara satu dengan yang lainnya melalui hubungan untuk menunjang misi bisnis. Oleh itu biasanya hubungan diterjemahkan sebagai keterkaitan bisnis di antara entiti. Garis yang menyambung diantara dua entiti dalam Diagram Hubungan Entiti (Entity Relationship Diagram) merupakan hubungan, seperti gambar di bawah ini.

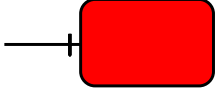
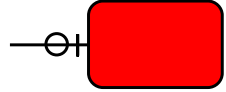
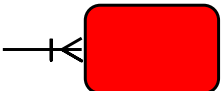
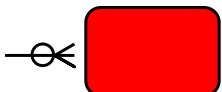
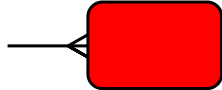


Gambar 2: Hubungan di antara dua entiti

Kardinalitas hubungan adalah kompleksitas sesuatu hubungan. Jumlah hubungan antara 1 entiti dengan entiti yang lain didefinisikan pada hubungan dua arah, seperti: 1:1, 1:m, m:n.

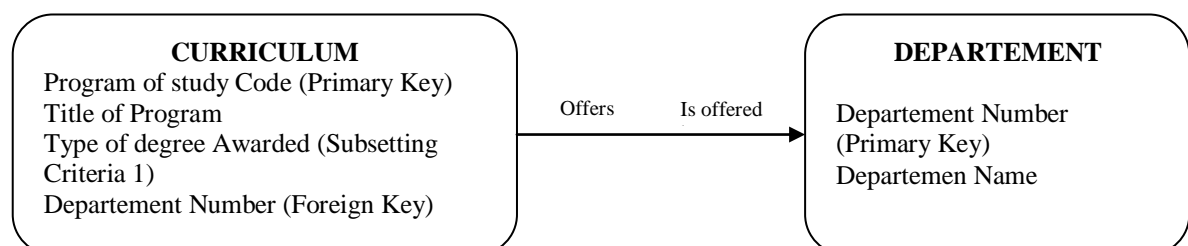
Tingkat hubungan merupakan jumlah entiti yang terlibat dalam hubungan tersebut yang terdiri dari empat tingkat, yaitu:

- ❖ Hubungan rekursif : tingkat = 1 karena melibatkan satu entiti saja, seperti: operator melaporkan kepada supervisor.
- ❖ Hubungan binary : tingkat = 2, karena 2 entiti terlibat dalam hubungan ini, seperti: pelajar mendaftar mata kuliah.
- ❖ Hubungan ternari : tingkat = 3. seperti: sponsor menawarkan beasiswa kepada pelajar.
- ❖ Entiti komposit yaitu entiti yang kunci utamanya diperoleh dari lebih satu entiti

Cardinality Interpretation	Minimum Instances	Maximum Instances	Graphic Notation
Exactly one	1	1	
Zero or one	0	1	
One or more	1	many (> 1)	
Zero, one, or more	0	many (> 1)	
More than one	> 1	> 1	

Tabel 1: Cardinality Notations

Di dalam hubungan juga terdapat istilah kunci asing yaitu kunci utama sebuah entiti yang ada dalam entiti lain. Kunci asing (dalam entiti child) senantiasa sebanding dengan kunci utama(dalam entiti parent).



Gambar 3: Hubungan di antara dua entiti

Entiti super-kelas & sub-kelas

Pada umumnya untuk mengetahui persamaan di antara setiap entiti maka biasanya super-kelas mempunyai sub-kelas. Atribut yang sama untuk beberapa jenis entiti dikelaskan dalam satu entiti yang dipanggil *super-kelas*. Seperti: sub-kelas dosen dan sub kelas karyawan mempunyai atribut yang sama yaitu nomor pekerja.

Entiti superkelas mempunyai beberapa atribut yang sama dengan satu atau lebih entiti subkelas. Dan entiti subkelas mempunyai atribut yang sama dengan entiti superkelas dan atribut lain yang membedakannya dengan entiti-entiti subkelas lain. Oleh karena itu entiti bisa menjadi superkelas dan subkelas.

3. Proses Permodelan Data Logikal

(a) Permodelan Data Strategik

Kebanyakan organisasi memilih proyek pembangunan aplikasi berdasarkan perancangan sistem informasi strategik. Projek ini menghasilkan perancangan strategik sistem informasi yang mendefinisikan visi dan arsitektur keseluruhan sistem informasi. Biasanya, arsitektur ini menggunakan model data enterprise.

Model data enterprise biasanya mengetahui entiti-entiti yang paling penting saja. Entiti-entiti ini didefinisikan (seperti dalam kamus) tetapi tidak diterangkan dari segi kunci atau atribut. Model ini bisa memasukkan atau tidak konsep hubungan (tergantung kepada kemampuan metodologi perancangan dan tahap perincian yang dikehendaki oleh pengelola eksekutif). Model data ini biasanya disimpan dalam *corporate repository*.

(b) Permodelan Data Pada Waktu Analisis Proses

Model data untuk sistem/aplikasi tunggal disebut model data aplikasi. Permodelan data pada waktu analisis proses beranggapan bahwa:

- ❖ Model data logikal mempunyai fokus data dan perspektif pengguna sistem.
- ❖ Model data logikal dibuat sebagai output fase kajian dan definisi projek.
- ❖ Model data logikal tidak berkaitan dengan teknologi dan bisa dibuat (melalui *reverse engineering*) dari pangkalan data langsung.

- ❖ Model data jarang dikaitkan dengan fase kajian analisis sistem. Kebanyakan analis sistem lebih suka melukis model proses untuk mendokumentasikan sistem langsung.
- ❖ Kebanyakan analis sistem mengatakan model data jauh lebih baik dikarenakan:
 - Model data membantu analis sistem mengetahui istilah bisnis dengan lebih lengkap dibandingkan dengan model proses.
 - Model data memerlukan waktu yang singkat dibandingkan model proses.
 - Model data yang lengkap bisa dilukis pada selembar kertas tetapi model proses memakan tempat lebih banyak.
 - Analisis model proses selalu memasukkan perincian yang tidak sepatutnya.
 - Model data sistem langsung dan yang disarankan lebih kurang sama dibandingkan model proses bisa menghemat waktu/tenaga.
- ❖ Model fase kajian perlu mengurutkan hubungan entiti - tanpa atribut - model data konteks. Ini bertujuan untuk lebih memahami ruang lingkup tanpa memerinci entiti.
- ❖ Model data fase definisi dibuat dalam dua tahap, yaitu:
 - Model data berdasarkan kunci : Model ini menghapuskan hubungan yang tidak spesifik, menambah entiti komposit, kunci utama, pilihan dan asing serta hirarki generalisasi.
 - Model data beratribut penuh : Memiliki semua atribut dan kriteria subset.

Setiap atribut didefinisikan dengan jenis, domain dan default dalam repository. Model data yang lengkap mewakili semua kebutuhan bisnis bagi pangkalan data sistem.

(c) Pencarian Fakta dan Pengumpulan Informasi untuk Permodelan Data

Model data dibuat tanpa fakta & informasi yang mencukupi dari pengguna. Fakta bisa dikumpulkan dari teknik persampelan fail dan pengisian secara langsung, kajian dari sistem yang sama, angket dan wawancara. Cara yang paling cepat untuk mengumpulkan data serta membuat dan mengesahkan model data secara serentak menggunakan joint application development (JAD).

(d) CASE Bagi Permodelan Data

Model data disimpan dalam repository. Model data ialah metadata yaitu data tentang data bisnis. CASE menyediakan repository untuk menyimpan model data & penerangan terperinci. Model data bisa dibuat dengan mudah dan model bisa diperbaiki berdasarkan saran pengguna. CASE menyediakan peralatan analitis yang bisa menyemak model dari segi perbaikan mekanikal, kelengkapan dan konsistensi.

4. Cara Membuat Model Data

Langkah 1 : Mengetahui Entiti

Perhatikan:

- Mengetahui entiti utama sistem yang mungkin diterangkan oleh data.
- Entiti mempunyai banyak instances : berpu;uh, ratus, ribu.
- Entiti dinamakan dengan sifat nama yang menerangkan individu, peristiwa, tempat atau benda yang bisa menyimpan informasi (coba jangan menggunakan nama ringkasan).
- Definisikan setiap entiti dalam istilah bisnis.
 - Jangan definisikan entiti dalam istilah teknikal dan ‘data tentang....’.
 - Nama dan definisi entiti perlu menunjukkan indeks istilah bisnis untuk kegunaan sistem analisis (termasuk akan datang) selama beberapa tahun.

Teknik:

- Memberi perhatian kepada *key words* yang disebut oleh pengguna dan pemilik sistem sewaktu wawancara atau sesi joint application development (JAD).
- Meminta pemilik dan pengguna sistem mengetahui masalah yang mereka ingin hasilkan, simpan dan dapatkan informasi.
- Meneliti folmulir atau fail pada saat ini.
- Adakan peralatan CASE yang bisa “*reverse engineer*” untuk membuat fail dan pangkalan data pada saat ini menjadi model data fizikal.

Entity Name	Business Definition
AGREEMENT	A contract whereby a member agrees to purchase a certain number of products within a certain time. After fulfilling that agreement, the member becomes eligible for bonus credits that are redeemable for free or discounted products. Note: A major system improvement objective is to make agreements more flexible with respect to other clubs. Currently, only purchases within the club that issued an agreement count toward credits. Another system improvement objective would award bonus credits for each purchase leading up to fulfillment of the agreement, with accelerated bonuses after fulfillment of the agreement.
CLUB	A SoundStage membership group to which members can belong. Clubs tend to be organized according to product interests such as music versus movies versus games; or specialized media interests such as Digital Video Disks (DVD) or Nintendo. Note: Cross-club interaction is a desired objective for the new system.
MEMBER	An active member of one or more clubs. Note: A target system objective is to re-enroll inactive members as opposed to deleting them.
MEMBER ORDER	An order generated for a member as part of a monthly promotion, or an order initiated by a member. Note: The current system only supports orders generated from promotions; however, customer initiated orders have been given a high priority as an added option in the proposed system.
PRODUCT	An inventoried product available for promotion and sale to members. Note: System improvement objectives include (1) compatibility with new bar code system being developed for the warehouse, and (2) adaptability to a rapidly changing mix of products.
PROMOTION	A monthly or quarterly event whereby dated orders are generated for all members in a club. Members then have some period of time to cancel or accelerate fulfillment of that order, after which the order is automatically filled.

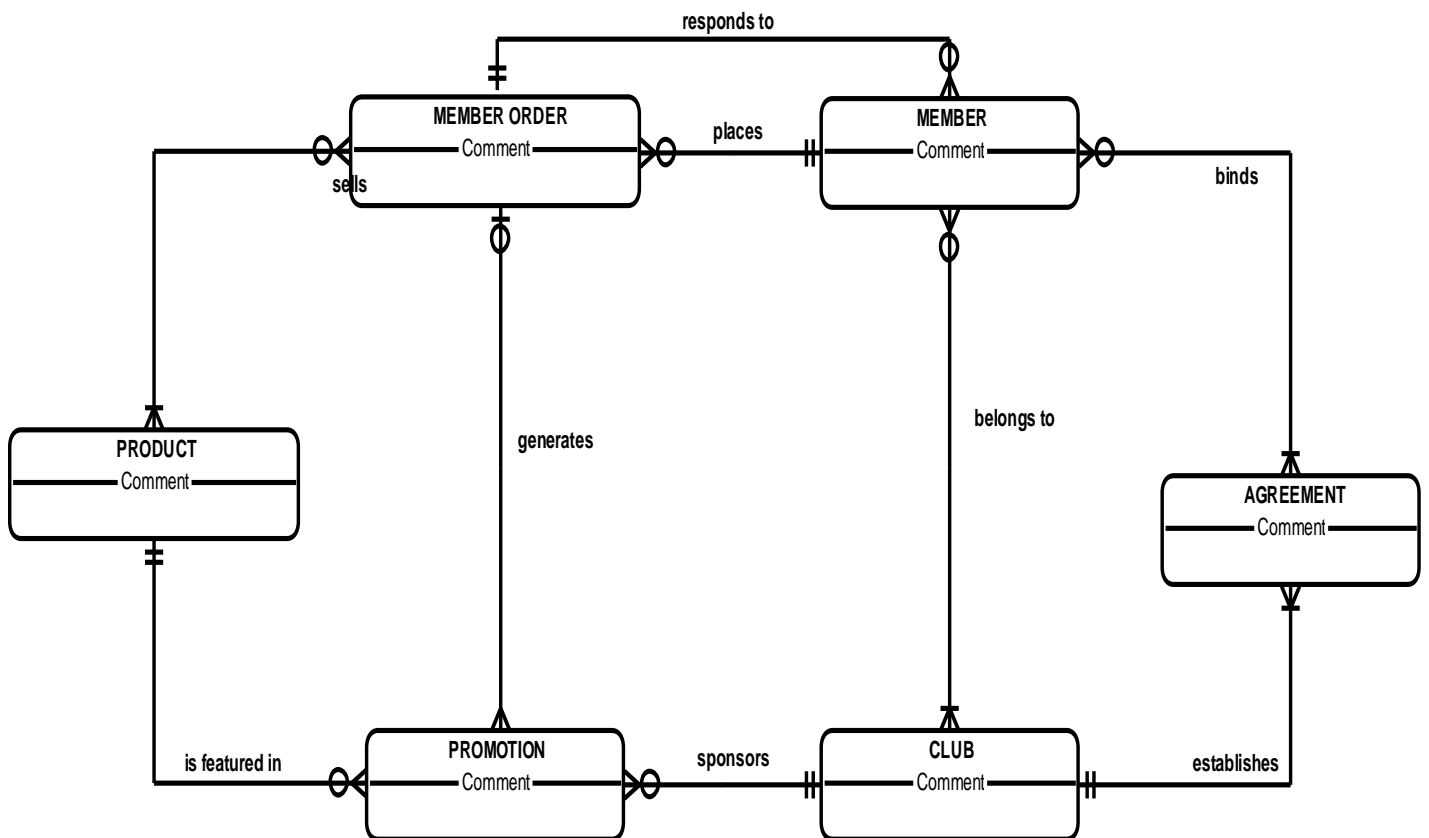
Tabel 2 : Fundamental Entities Identified for the soundstage Project

Langkah 2 : Membuat Model Data Konteks

Model data konteks meliputi entiti dasar/bebas yang telah diketahui. Entiti bebas kunci utamanya adalah tidak meliputi atribut yang tergantung kepada keadaan entiti lain. Hubungan di dalam model menggunakan kata kerja yang membentuk kalimat bisnis bila digabungkan dengan entiti.

Model bisa menggunakan:

- **Kode Abjad:** menggunakan kombinasi huruf (atau nombor) tertentu untuk menerangkan instance entiti. Kode abjad biasanya digabungkan dengan kode berseri/blok untuk mengetahui keadaan entiti.
- **Significant position codes:** setiap digit/sekelompok digit menerangkan ciri-ciri entiti instance yang bisa diukur/diketahui. Biasanya digunakan untuk membuat kode item inventori.



Gambar 4 : A Context Data Model

Langkah 3 : Model Data Berdasarkan Kunci

Perhatikan:

- Mengetahui kunci setiap entiti, melalui:
 - Nilai kunci tidak akan bertukar di selama waktu entiti.
 - Nilai kunci tidak bisa nol.
 - Kontrol mesti diletakkan untuk memastikan nilai kunci sah.

- Coba cipta satu kunci baru untuk menggantikan kunci komposit yg panjang. (Saran ini tidak sesuai utk entiti komposit karena setiap bagian kuncinya merupakan kunci asing yang mesti sebanding dengan kunci utama entiti parent).
- Jika kunci tidak bisa ditakrif, entiti tersebut mungkin tidak ada.

Kode-kode Bisnes :

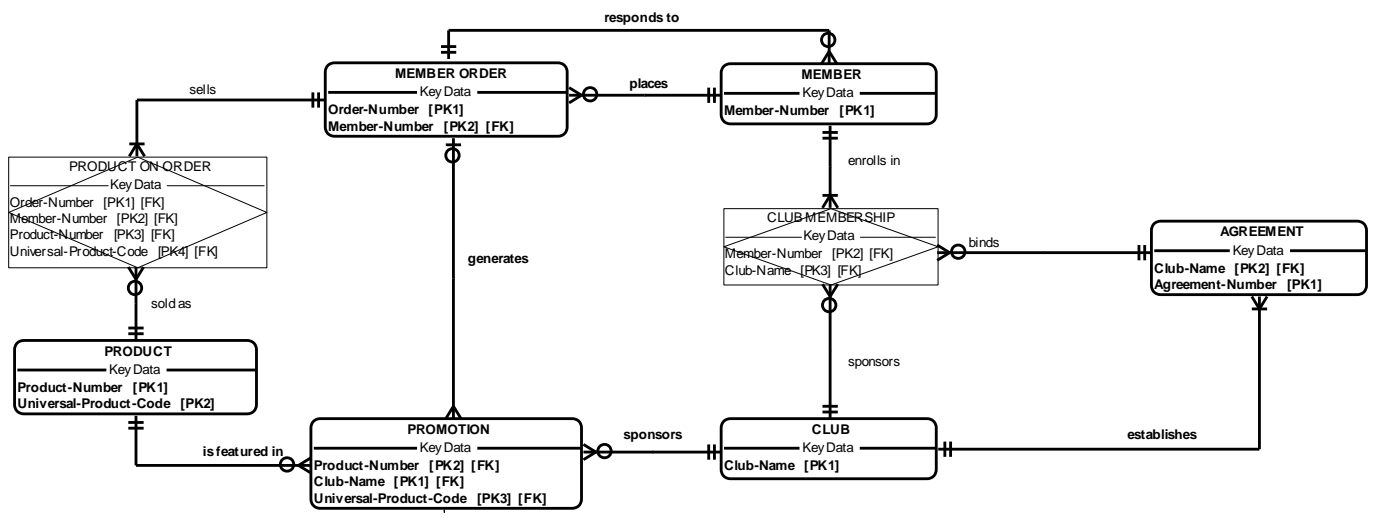
Terdapat beberapa jenis kode yang bisa digabungkan untuk mengetahui entiti,yaitu:

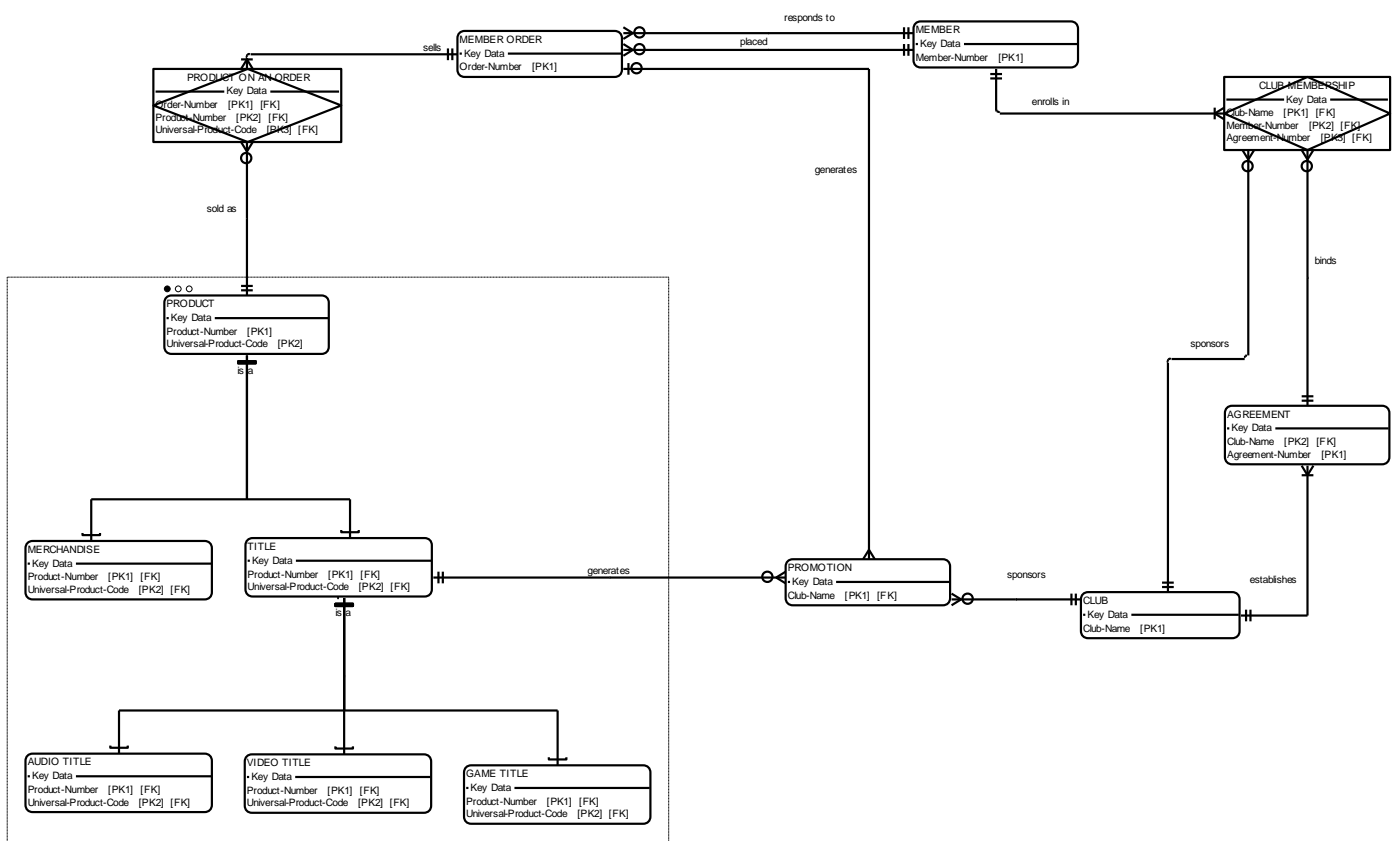
- Kode Berseri: mengumpulkan nombor-nombor yang disusun secara beraturan terhadap instance entiti. Kebanyakan DBMS bisa menyusun kode berseri terhadap kebutuhan bisnes.
- Kode Blok: sama dengan kode berseri kecuali nombor berseri tersebut dibagikan kepada kelompok yang mempunyai makna bisnes yang sama.
- Kode Hirarki: memberikan makna secara top-down terhadap instance entiti. Setiap item yg dikodekan diletakkan kedalam kelompok, subkelompok, dll.

Panduan membuat skema pengkodean bisnes:

- Kode bisa dipanjangkan untuk menampung perkembangan bisnes.
- Kode lengkap mesti menghasilkan nilai unik setiap entiti.
- Ukuran kode memerlukan cukup besar untuk menerangkan perbedaan ciri tetapi cukup kecil untuk didefinisikan oleh individu tanpa menggunakan komputer.
- Kode perlu “convenient”.

Langkah 4 - Menggeneralisasikan Hirarki : mengetahui hirarki-hirarki generalisasi dalam masalah bisnes.

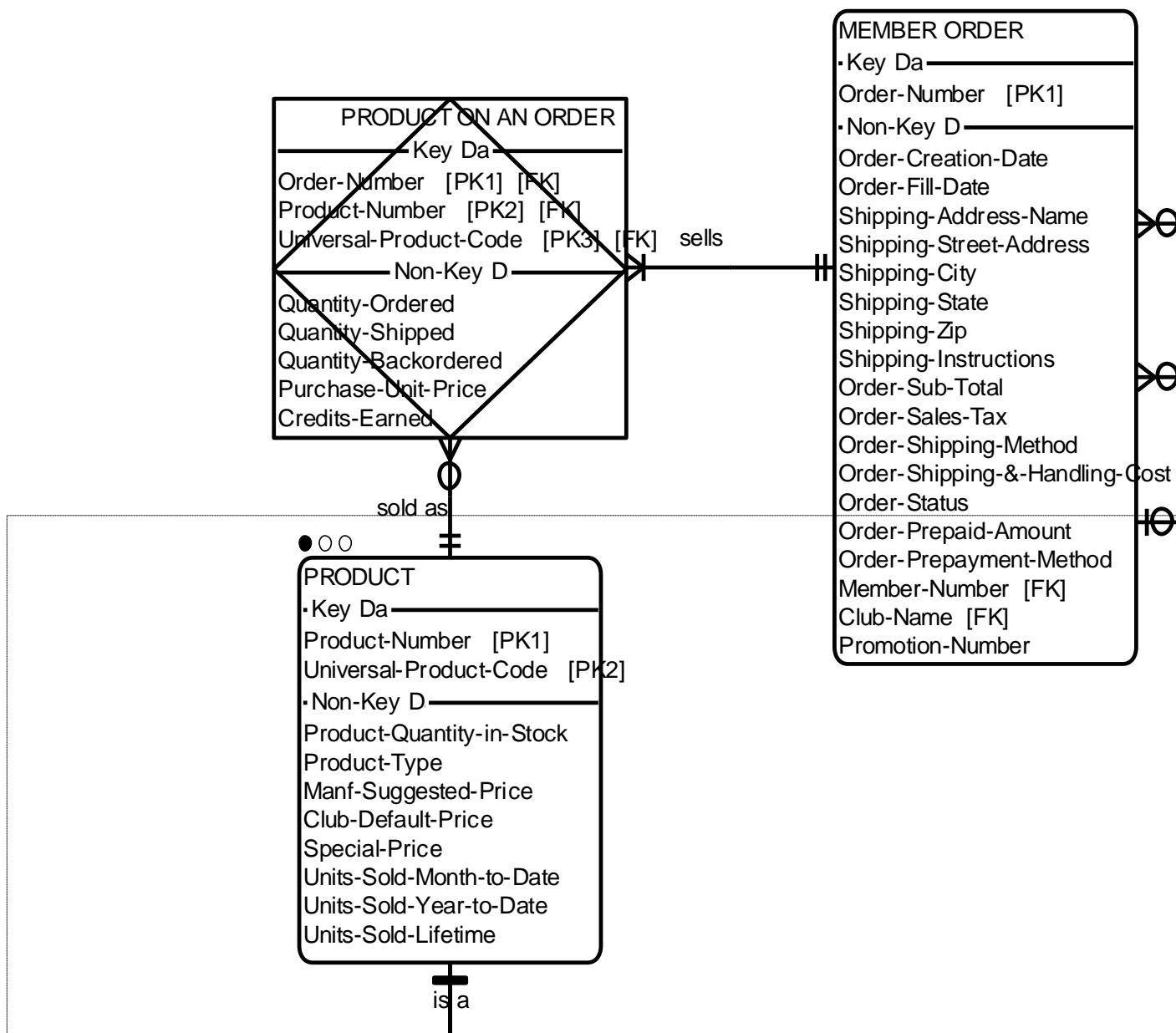




Langkah 5 - Model Data Beratribut Lengkap : Mengenal atribut yang lain.

Panduan :

- ❖ Kebanyakan organisasi mempunyai aturan dalam memberi nama dan singkatan yang disetujui. Ini bisa dilakukan oleh manajemen data/repository.
- ❖ Kebanyakan atribut berpadu dengan nama dasar yang sama seperti NAMA, ALAMAT, TANGGAL. Berikan nama yang unik untuk setiap atribut (kecuali jika atribut bisa digeneralisasikan ke dlm superkelas) :
CUSTOMER NAME vs SUPPLIER NAME
- ❖ Untuk setiap atribut yang hanya mempunyai nilai YA/ TIDAK, namakan sebagai pertanyaan. Misalnya: CANDIDATE FOR A DEGREE?
- ❖ Setiap atribut perlu dipetakan kepada hanya satu entiti, kecuali kunci asing - ia mengenal entiti lain yang berkaitan.
- ❖ Domain atribut tidak harus berpatokan kepada logika.



Langkah 6 - Model Yang Diterangkan Sepenuhnya

Model yang harus diterangkan :

- ❖ Menerangkan Data Model sepenuhnya .
- ❖ Mengambil waktu yang paling lama.
- ❖ Tugas ini bisa dimulai sekaligus dengan model berdasarkan kunci atau model beratribut lengkap. Biasanya ia model terakhir yang disiapkan.
- ❖ Pada tahap ini penerangan atribut masih tidak lengkap - tanpa domain. Kebanyakan peralatan CASE tools menyediakan fasilitas untuk menerangkan jenis data, domain dan nilai default utk setiap atribut.
- ❖ Ciri-ciri penerangan tambahan bisa disimpan untuk atribut seperti :
 - ✓ Siapa yang bisa mencipta, hapus, merubah dan mencapai atribut?
 - ✓ Berapa lama harus atribut/entiti disimpan sebelum dihapus/diarkib?