Analisis Pemodelan Data Lanjutan: Enhanced Entity Relationship Diagram (EERD)

Muhammad Riduwan Khafidi
Program Studi Teknik Rekayasa Perangkat Lunak
Politeknik Negeri Batam
4342611034@students.polibatam.ac.id

November 2, 2026

Abstract

Entity Relationship Diagram (ERD) standar terkadang tidak cukup untuk merepresentasikan kompleksitas aturan bisnis di dunia nyata. Untuk mengatasi keterbatasan ini, muncullah Enhanced Entity Relationship Diagram (EERD). EERD memperluas konsep ERD dengan menambahkan fitur-fitur berorientasi objek seperti Spesialisasi (Specialization), Generalisasi (Generalization), Pewarisan (Inheritance), dan Agregasi (Aggregation). Makalah ini membedah secara komprehensif mekanisme EERD dalam memodelkan hirarki entitas, di mana sebuah entitas induk dapat memiliki sub-entitas yang lebih spesifik. Melalui visualisasi diagram, makalah ini menjelaskan bagaimana atribut dan relasi diwariskan dari entitas tingkat tinggi ke tingkat rendah, serta bagaimana sebuah relasi dapat diabstraksikan menjadi entitas baru (agregasi) untuk memenuhi aturan temporal (waktu) dalam suatu transaksi.

1 Pendahuluan: Mengapa ERD Perlu Diperluas?

Dalam praktikum sebelumnya, kita telah memahami bahwa ERD adalah fondasi untuk memetakan entitas dan relasi. Namun, bayangkan sebuah universitas yang memiliki dua jenis dosen: Dosen Tetap (Permanent Lecturer) dan Dosen Tidak Tetap (Part-time Lecturer). Kedua jenis dosen ini memiliki atribut yang sama seperti Nama, ID Pegawai, dan Alamat. Namun, Dosen Tetap memiliki atribut spesifik seperti Golongan (Grade) dan Tanggal Mulai, sedangkan Dosen Tidak Tetap memiliki atribut seperti Nama Perusahaan dan Jumlah Jam Mengajar.

Jika kita memaksakan keduanya ke dalam satu entitas "Dosen", akan banyak atribut yang kosong (bernilai *null*). Jika kita memisahkannya menjadi dua entitas yang benar-benar terpisah, kita akan kehilangan esensi bahwa keduanya sama-sama "Dosen". Di sinilah Enhanced Entity Relationship Diagram (EERD) hadir sebagai solusi. EERD memungkinkan kita merancang sub-kelompok dari sebuah entitas set, sebuah konsep yang disebut Spesialisasi.

2 Spesialisasi (Specialization)

Spesialisasi adalah proses desain untuk membuat sub-kelompok (sub-entitas) dari sebuah entitas set berdasarkan karakteristik atau atribut yang berbeda. Ini adalah pendekatan *top-down*,

di mana kita memulai dari entitas yang paling umum (Superclass), lalu memecahnya menjadi entitas yang lebih spesifik (Subclass).

Dalam notasi EERD, Spesialisasi digambarkan dengan sebuah segitiga terbalik yang di dalamnya terdapat tulisan **IS A** (adalah sebuah). Segitiga ini menghubungkan Superclass di atas dengan Subclass di bawahnya. Perhatikan Gambar 1 di bawah ini.

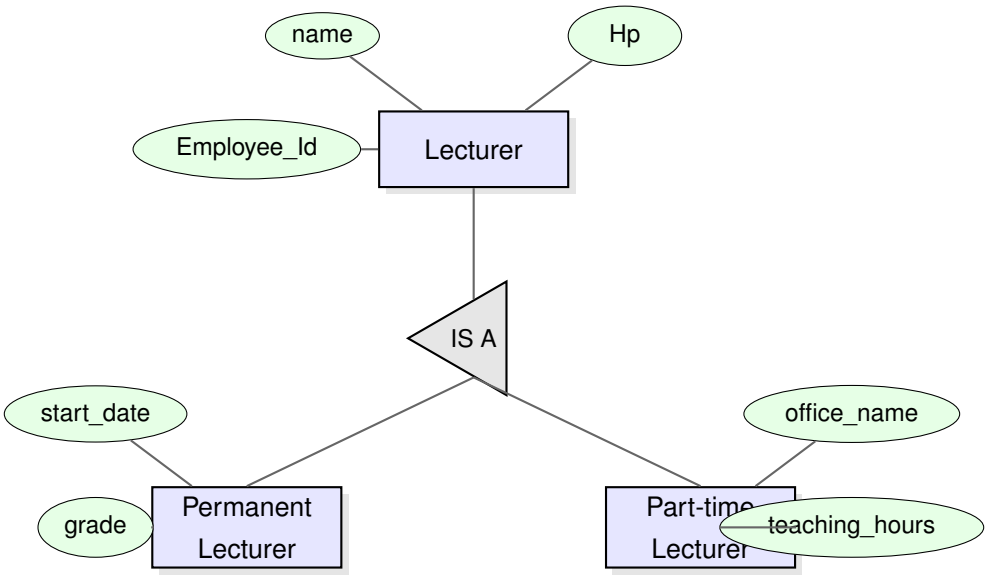


Figure 1. Contoh Spesialisasi pada Entitas Lecturer.

Melalui Spesialisasi, entitas "Permanent Lecturer" hanya akan memiliki atribut spesifiknya sendiri tanpa harus memaksakan atribut "Part-time Lecturer" masuk ke dalamnya.

3 Generalisasi (Generalization)

Jika Spesialisasi adalah pendekatan *top-down*, maka Generalisasi adalah kebalikannya, yaitu pendekatan *bottom-up*. Generalisasi adalah proses menggabungkan beberapa entitas set yang memiliki atribut serupa menjadi satu entitas set yang lebih umum (Superclass).

Sebagai contoh, bayangkan kita memiliki entitas "Mahasiswa D3" dan "Mahasiswa D4". Setelah dianalisis, ternyata keduanya memiliki atribut yang sama persis (NIM, Nama, Jurusan, IPK), kecuali pada jenjang pendidikannya. Daripada membuat dua tabel terpisah, kita dapat menggeneralisasi keduanya menjadi satu entitas "Mahasiswa", lalu menambahkan satu atribut baru untuk menyimpan status jenjang (D3 atau D4), seperti yang diilustrasikan pada Gambar 2.

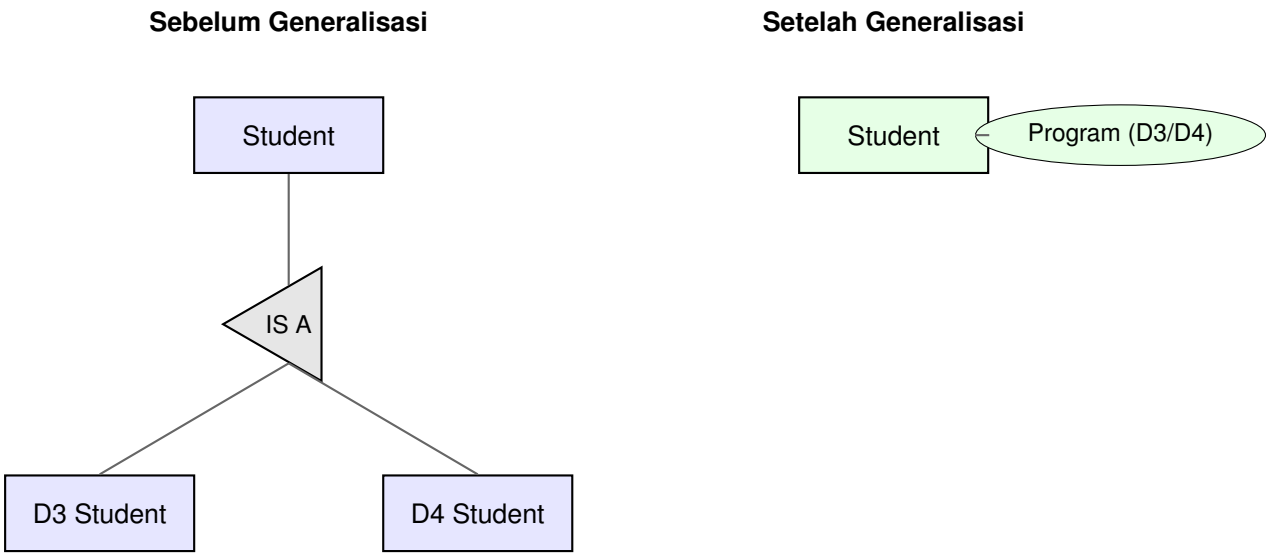


Figure 2. Proses Generalisasi Entitas Mahasiswa.

4 Pewarisan (Inheritance)

Konsep kunci dalam EERD adalah **Pewarisan** (*Inheritance*). Sebuah sub-entitas secara otomatis mewarisi seluruh atribut dan relasi yang dimiliki oleh entitas induknya (Superclass).

Sebagai contoh, karena "Permanent Lecturer" adalah bagian dari "Lecturer", maka ia otomatis memiliki atribut *Employee_Id*, *name*, dan *Hp* tanpa perlu mendeklarasikannya ulang. Selain atribut, relasi juga diwariskan. Jika entitas "Lecturer" memiliki relasi "mengajar" (teach) ke entitas "Student", maka baik "Permanent Lecturer" maupun "Part-time Lecturer" juga memiliki relasi tersebut. Namun, sub-entitas juga diizinkan untuk memiliki relasi eksklusif yang tidak dimiliki oleh sub-entitas lainnya, seperti yang ditunjukkan pada Gambar 3.

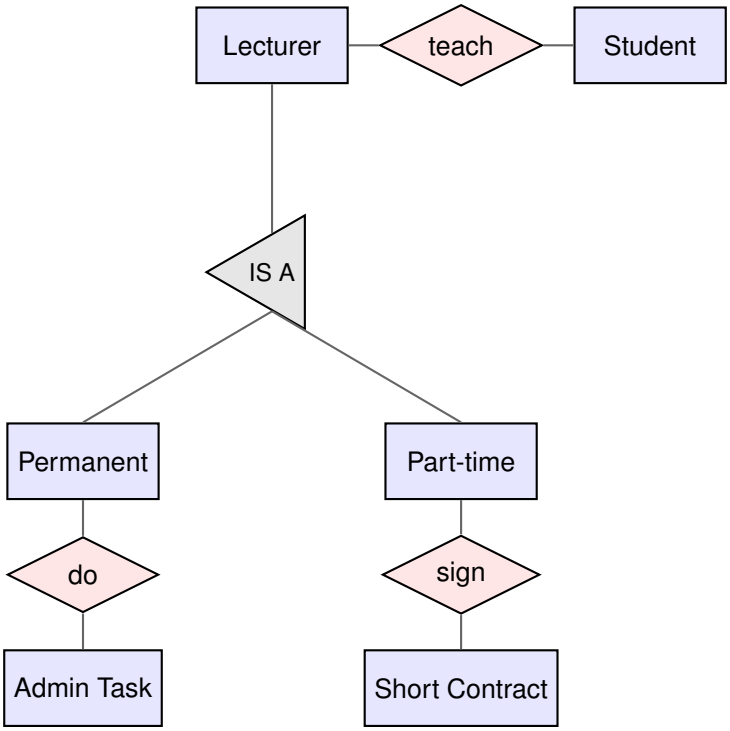


Figure 3. Pewarisan Relasi dan Relasi Eksklusif pada Sub-entitas.

5 Agregasi (Aggregation)

Dalam pemodelan bisnis nyata, terkadang kita menjumpai sebuah relasi yang baru bisa terjadi *setelah* relasi lainnya terjadi. Sebagai contoh, pada sistem perbankan, seorang "Customer" meminjam uang (relasi *borrow*) ke entitas "Loan". Setelah pinjaman itu terjadi, barulah muncul transaksi "Pembayaran" (relasi *pay*) ke entitas "Payment".

Kita tidak bisa langsung menghubungkan entitas "Payment" ke entitas "Customer" atau "Loan" secara langsung, karena pembayaran hanya valid jika ada pinjaman. Untuk merepresentasikan hal ini, kita menggunakan konsep **Agregasi**. Agregasi adalah abstraksi di mana sebuah relasi (beserta entitas-entitas yang terlibat di dalamnya) diperlakukan sebagai satu entitas set baru (entitas tingkat tinggi).

Perhatikan perbedaan desain pada Gambar 4.

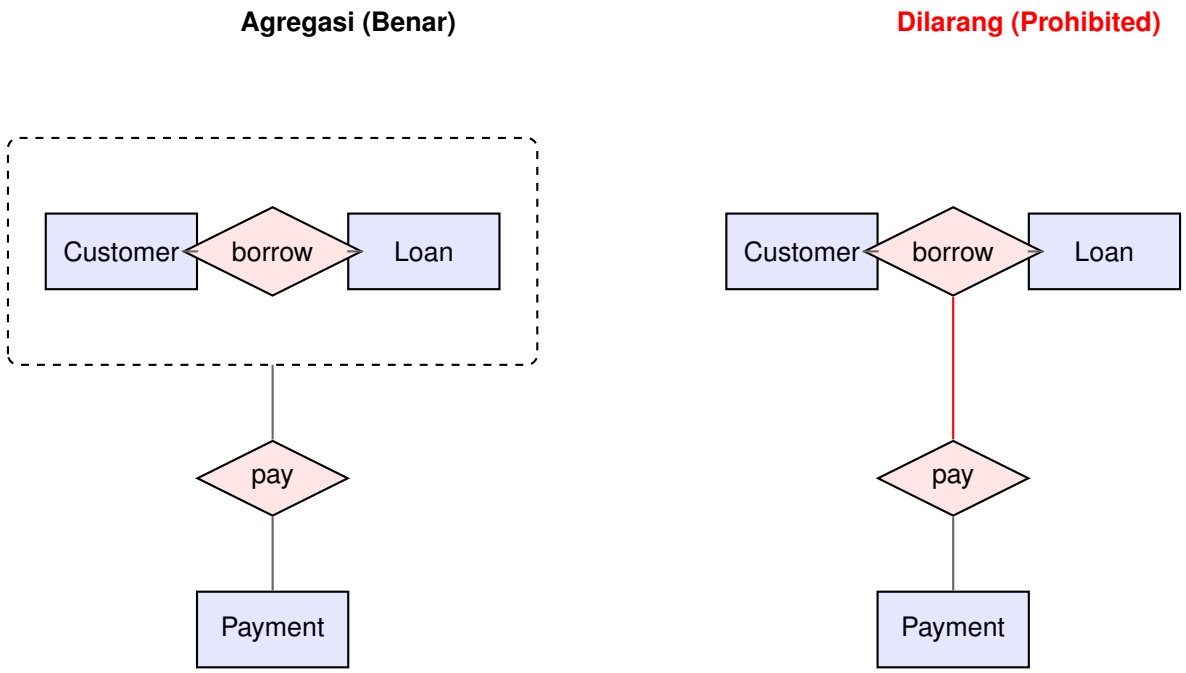


Figure 4. Representasi Agregasi pada Relasi Pinjaman dan Pembayaran.

Pada gambar di atas, kotak bergaris putus-putus menandakan bahwa "Customer", "Loan", dan relasi "borrow" telah diagregasi menjadi satu kesatuan entitas. Dengan demikian, relasi "pay" dapat dihubungkan ke agregat tersebut. Menghubungkan relasi secara langsung ke relasi lain (seperti pada gambar kanan) adalah hal yang dilarang dalam aturan baku EERD.

6 Kesimpulan

Enhanced Entity Relationship Diagram (EERD) memberikan kekuatan lebih bagi perancang basis data untuk memodelkan aturan bisnis yang kompleks. Melalui Spesialisasi dan Generalisasi, kita dapat mengelola hirarki entitas secara efisien, memastikan bahwa setiap entitas hanya menyimpan atribut yang relevan tanpa terjadi duplikasi. Konsep Pewarisan (Inheritance) memudahkan pemeliharaan relasi antar entitas. Sementara itu, Agregasi memungkinkan kita memodelkan ketergantungan temporal antar relasi, di mana sebuah transaksi baru dapat terjadi setelah transaksi sebelumnya tervalidasi. Penguasaan EERD adalah jembatan krusial sebelum skema konseptual ini ditransformasikan menjadi tabel fisik yang dinormalisasi di dalam DBMS.

Referensi

References

[1] Lubis, A. I. (2026). *Enhanced Entity Relationship Diagram* [Slide Presentasi Mata Kuliah Pengantar Basis Data]. Politeknik Negeri Batam.

[2] Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database System Concepts* (7th ed.). McGraw-Hill Education.

[3] Fathan, S. (2012). *Basis Data Edisi Revisi*. Penerbit Informatika.

[4] Kadir, A. (2009). *Dasar Perancangan dan Implementasi Database Relasional*. Penerbit Andi.