

TECHNICAL DOCUMENTATION

Misi 2: Physical Design & Development

Project: Data Mart Biro Akademik Umum (BAU) ITERA

Kelompok: 19

Tanggal: 24 November 2025

1. Pendahuluan

Dokumen ini berisi rincian teknis mengenai implementasi fisik (*Physical Design*) dari Data Mart Biro Akademik Umum ITERA. Lingkup dokumen mencakup struktur database, strategi ETL, optimasi performa, serta mekanisme keamanan yang telah diterapkan pada lingkungan *production* menggunakan Microsoft SQL Server 2019.

2. Arsitektur Database (Physical Design)

2.1 Struktur Schema

Database `datamart_bau_itera` diorganisasikan ke dalam beberapa skema logis untuk memisahkan fungsi dan tanggung jawab data:

Schema	Deskripsi	Tipe Objek
<code>stg</code>	<i>Staging Area.</i> Tempat pendaratan data mentah dari CSV sebelum diproses.	Tabel (Raw Data)
<code>dim</code>	<i>Dimension.</i> Menyimpan data referensi dan master data (SCD Type 1 & 2).	Tabel Dimensi
<code>fact</code>	<i>Fact.</i> Menyimpan data transaksional dan snapshot histori.	Tabel Fakta
<code>etl</code>	<i>ETL Logic.</i> Menyimpan prosedur pemrosesan data.	Stored Procedures
<code>etl_log</code>	<i>Logging.</i> Menyimpan riwayat eksekusi ETL dan audit error.	Tabel Log

reports	<i>Reporting.</i> Menyimpan View untuk kebutuhan dashboard/laporan.	Views
----------------	---	-------

3. Strategi ETL (Extract, Transform, Load)

Proses pemindahan data dilakukan menggunakan pendekatan **ELT (Extract, Load, Transform)** dengan memanfaatkan kekuatan *processing* dari SQL Server.

3.1 Alur Data

1. **Ingestion:** Data dari file CSV diimpor ke tabel `stg` menggunakan *Bulk Insert* atau *Import Wizard*.
2. **Transformation:** Data dibersihkan, divalidasi, dan ditransformasi menggunakan T-SQL Stored Procedures.
3. **Loading:**
 - o **Dimensi:** Menggunakan logika `MERGE` untuk menangani data baru dan perubahan data (*SCD Type I*).
 - o **Fakta:** Menggunakan `INSERT INTO ... SELECT` dengan *Lookup* ke tabel dimensi untuk mendapatkan *Surrogate Key*.

3.2 Orchestration

Seluruh proses dikendalikan oleh satu prosedur utama yaitu `etl.usp_MasterETL`, yang memastikan urutan eksekusi yang benar (Dimensi dahulu, baru Fakta) dan mencatat log eksekusi.

4. Optimasi & Performa

Untuk menjamin kecepatan *query* pada saat pelaporan (Misi 3), strategi berikut telah diterapkan:

4.1 Indexing

- **Clustered Index:** Diterapkan secara otomatis pada *Primary Key* (Surrogate Key) setiap tabel.
- **Non-Clustered Index:** Dibuat pada seluruh kolom *Foreign Key* di tabel Fakta untuk mempercepat operasi JOIN.
- **Filtered Index:** Dibuat khusus untuk query operasional, contohnya index pada `fact_surat` dengan filter `status_akhir <> 'Selesai'` untuk mempercepat pemantauan surat yang tertunda.

4.2 Partitioning

Tabel Fakta utama (`Fact_Surat` dan `Fact_Layanan`) telah dipartisi secara horizontal berdasarkan **Tahun** (`DateKey`).

- **Partition Function:** `pf_YearlyRange` (Range Right).
- **Manfaat:** Mempercepat query yang memfilter data berdasarkan periode tahun tertentu (Partition Elimination).

5. Keamanan & Operasional

5.1 Manajemen Akses (RBAC)

Akses database dikelola menggunakan prinsip *Least Privilege* dengan dua Role utama:

- **role_analyst**: Hak akses **Read-Only** (SELECT) pada schema **dim**, **fact**, dan **reports**. Digunakan oleh Tim BI (Power BI).
- **role_etl_admin**: Hak akses **Full Control** (CRUD + Execute) pada semua schema. Digunakan oleh Tim Data Engineer untuk proses ETL.

5.2 Backup & Recovery

Strategi pencadangan data telah disiapkan menggunakan Stored Procedures:

- **Full Backup**: Mingguan (usp_Backup_Full).
- **Differential Backup**: Harian (usp_Backup_Diff).
- **Transaction Log Backup**: Per jam (usp_Backup_Log).