Data Modeling Techniques

Introduction

The data warehouse was designed to support both analytics and operational use cases. By creating a well-designed model that supports both the dimensional tables and transactional tables. For a time tracking and resource allocation system, the data model has been developed to accommodate both analytical (OLAP) and operational (OLTP) use cases.

Dimensional Model (Analytical Purposes)

Design Choices

1. Star Schema Implementation

- Two fact tables: ***fact_time_tracking*** and ***fact_allocation***
- Five-dimension tables: ***dim_client, dim_project, dim_employee, dim_role, and dim_date***
- Optimized for complex analytical queries and reporting

Fact Tables
This contains the incremental tables that will continue to grow on a steady basis. Optimized for analytical queries.

- **fact_time_tracking:** Stores actual time entries from ClickUp
- **fact_allocation:** Stores resource allocation/planning data from Float

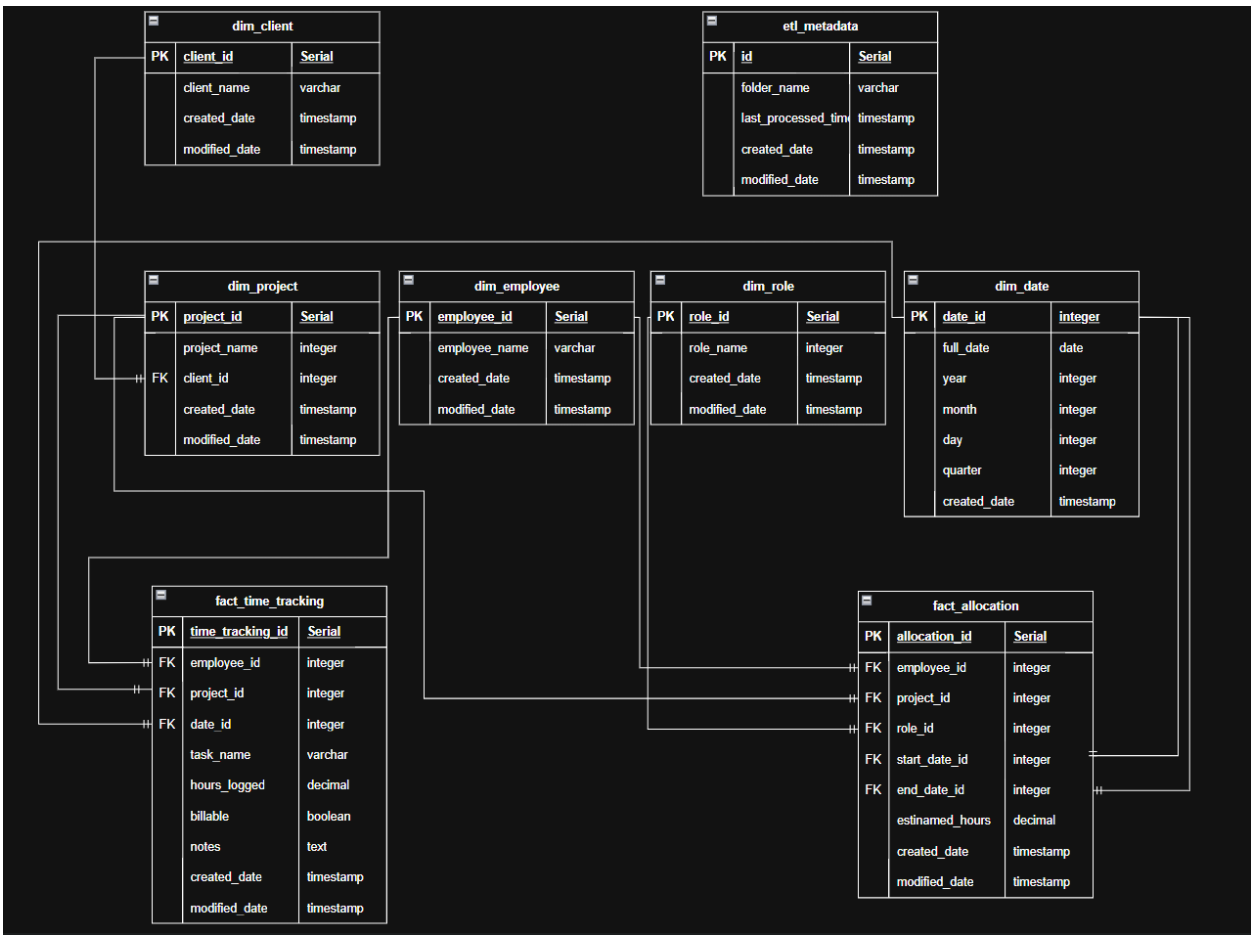
Dimension Tables
This is information about the fact table containing necessary information used in the modeling process.

- **dim_client:** Client information
- **dim_project:** Project details with client relationships
- **dim_employee:** Employee information
- **dim_role:** Role definitions
- **dim_date:** Date dimension for time-based analysis

ETL Metadata
These are standard data used in keeping records and changes.

Entity Relationship Diagram

Entity-relationship (ER) modeling is a visual approach to data modeling used to represent the structure of a database. It is used to identify the "things" (entities) in a system and how they relate to each other.



Full ER Diagram & Modeling(Analytics & Operational Purpose)

One-to-Many Relationships:

- A client can have multiple projects (1:M)
- A project can have multiple time tracking entries (1:M)
- A project can have multiple allocations (1:M)
- An employee can have multiple time tracking entries (1:M)
- An employee can have multiple allocations (1:M)
- A role can be used in multiple allocations (1:M)
- A date can be referenced by multiple time tracking entries (1:M)
- A date can be the start or end date for multiple allocations (1:M)

Fact Table Relationships:

fact_time_tracking connects to:

- dim_employee
- dim_project
- dim_date

fact_allocation connects to:

- dim_employee
- dim_project
- dim_role
- dim_date (twice, for start and end dates)

Independent Tables:

- etl_metadata is independent and used for ETL process tracking

Analytical Benefit

- **Performance Optimization:** Denormalized dimension tables reduce JOIN complexity.
- **Business Intelligence Capabilities:** Easy aggregation of hours by various dimensions.
- **Reporting Flexibility:** Multiple grain levels (daily, monthly, quarterly)

Entity-Relationship Model (Operational Purposes)

Design Choices

Normalized Structure

- Clear entity relationships with referential integrity
- Primary and foreign key constraints
- Audit fields (***created_date, modified_date***)

Operational Benefits

Data Integrity

- Foreign key constraints ensure referential integrity
- SERIAL primary keys prevent duplicate entries
- NOT NULL constraints where appropriate
- UNIQUE constraints on critical fields

Transaction Processing

- Efficient CRUD operations
- Minimal data redundancy

Operational Features

- Real-time time tracking entry
- Resource allocation management

Conclusion

This hybrid design effectively balances analytical and operational needs by ensuring data integrity, enabling efficient reporting, supporting real-time and historical analysis, and allowing for future scalability and modifications.