

**Milestone 3**  
*of*  
**Data Warehousing and Integration**  
**BY**  
**GROUP NUMBER 14:**

ARCHIT SINGH (002813253)

SANCIA SEROPHENE SALDANHA (002851577)



**DEPARTMENT OF COLLEGE OF ENGINEERING**  
**NORTHEASTERN UNIVERSITY**  
**BOSTON, MASSACHUSETTS – 022115**

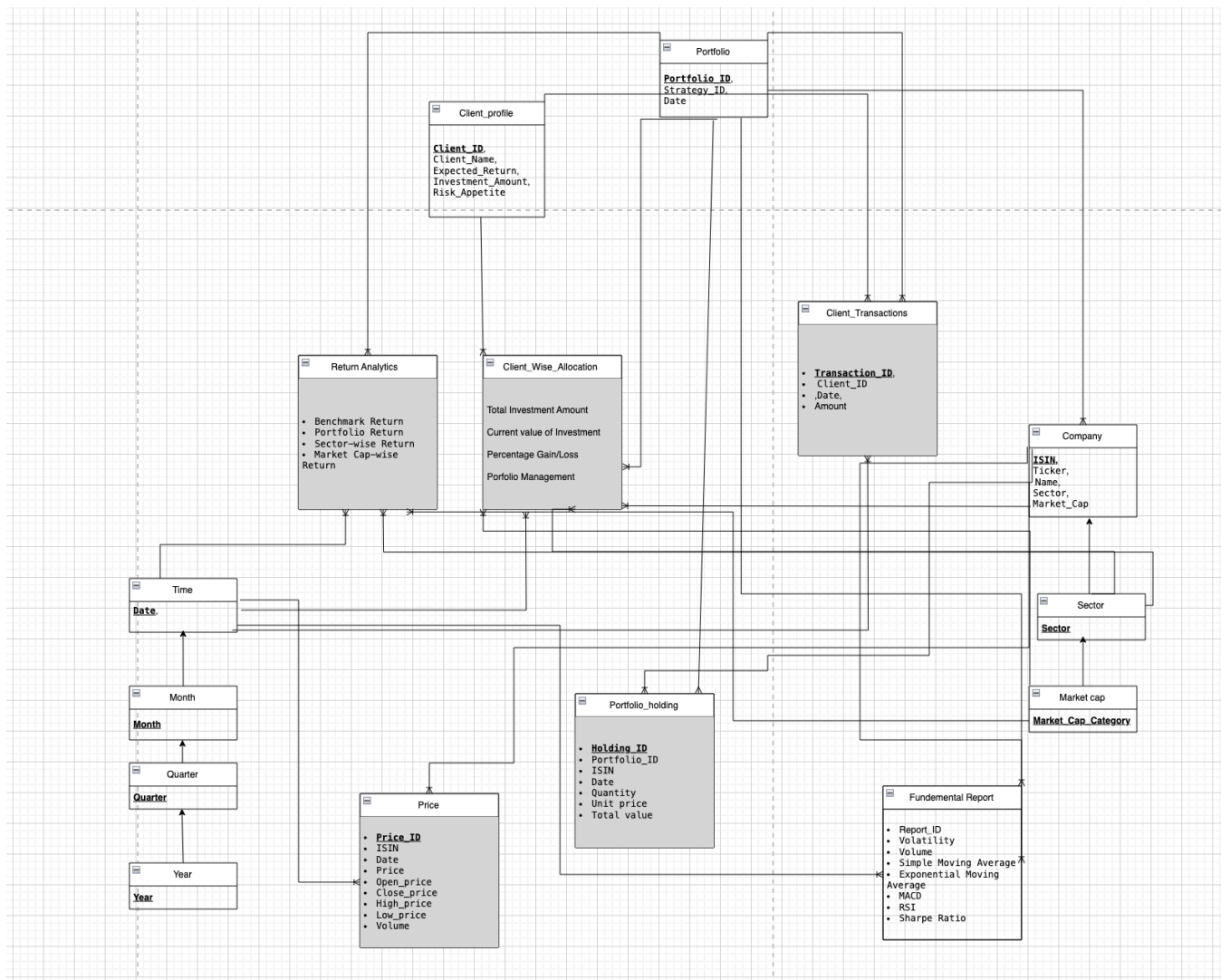
## **A. Conceptual Model**

The primary entities in the conceptual model include:

- Client\_Profile
- Company
- Portfolio\_Creation
- Technical\_Strategy
- Fundamental\_Report
- Time

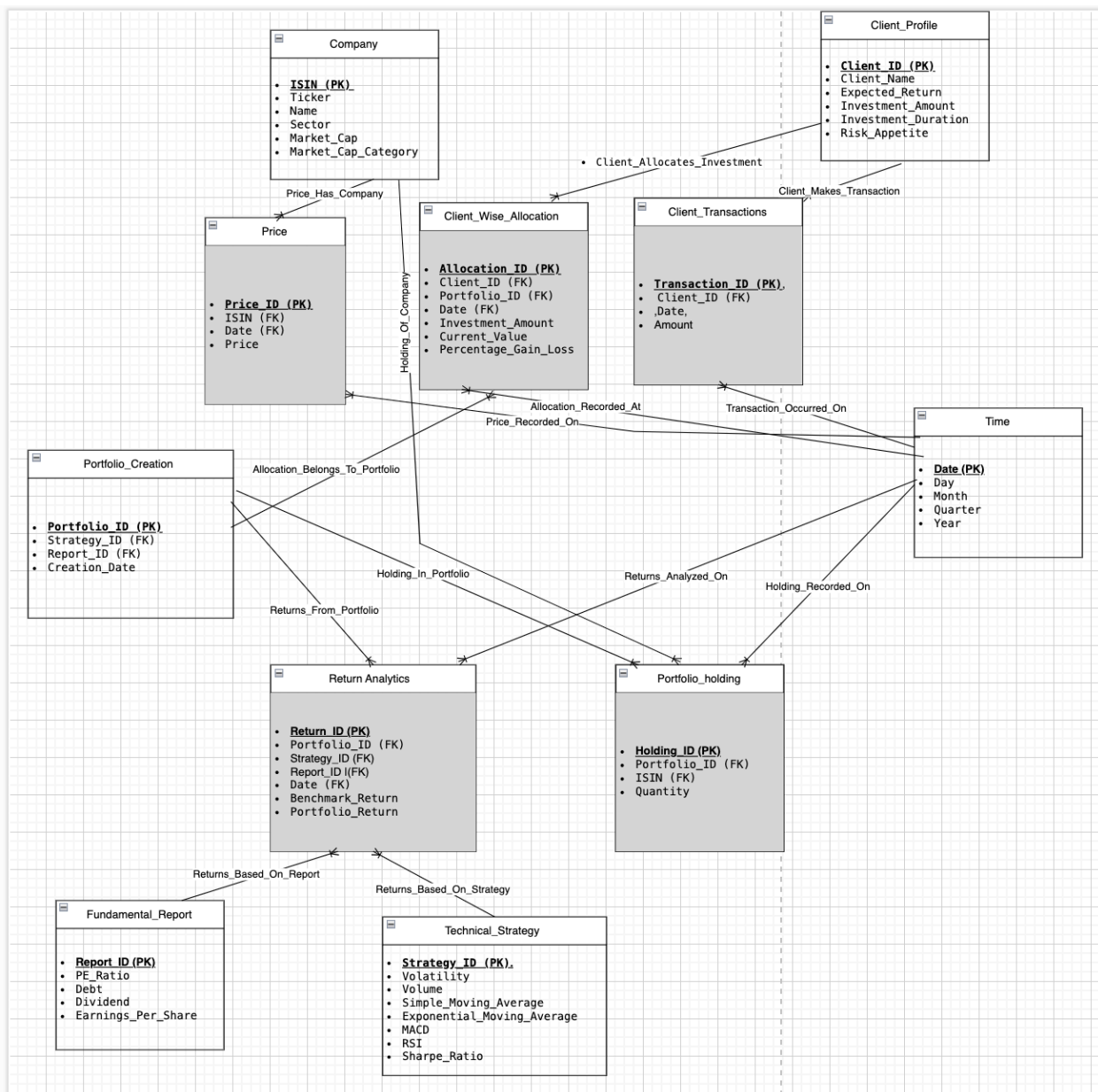
These entities represent clients, companies available for investment, strategies and reports used to form portfolios, and time, which is vital for temporal analysis.

Hierarchies within the conceptual model include the Time dimension, which includes attributes such as Year, Quarter, Month and Day, allowing for aggregation at different levels of granularity



## B. Logical Model

The logical model of this data warehouse provides a structured blueprint that translates the high-level conceptual design into a detailed representation suitable for implementation. It defines the key entities, such as Client\_Profile, Company, Portfolio\_Creation, along with their attributes, primary keys (PK), and foreign keys (FK), specifying how they relate to each other. This model captures fact tables that store essential business events—such as Client\_Transactions and Portfolio\_Holding—and dimension tables that provide descriptive context, like client details and investment strategies.



### C. OLAP Operations:

The following examples show how each **OLAP operation** can be utilized in our financial data warehouse to derive valuable insights:

1. **Roll-UP:** Summarize Client\_Transaction data monthly to see the total transactions per month

```
R1 ← ROLLUP* (Client_Transaction, Date → Month, SUM(Amount) AS Total_Transaction_Amount)
```

2. **Drill-Down:** Drill down from yearly returns to monthly returns in Return\_Analytics to identify specific trends

```
R1 ← ROLLUP* (Return_Analytics, Date → Year, SUM(Portfolio_Return) AS Yearly_Return)
```

```
R2 ← ROLLUP* (Return_Analytics, Date → Month, SUM(Portfolio_Return) AS Monthly_Return)
```

3. **Slice:** Slice Client\_Wise\_Allocation for high-risk clients to analyze their investment patterns

```
R1 ← JOIN (Client_Wise_Allocation, Client_Profile USING Client_ID)
```

```
R2 ← DICE (R1, Client_Profile.Risk_Appetite = 'High')
```

4. **Dice:** Dice Portfolio\_Holding to view holdings only for 2023 portfolios using a specific strategy

```
R1 ← JOIN (Portfolio_Holding, Portfolio_Creation USING Portfolio_ID)
```

```
R2 ← DICE (R1, Portfolio_Creation.Date BETWEEN '2023-01-01' AND '2023-12-31' AND Portfolio_Creation.Strategy_ID = [Specific_Strategy_ID])
```

5. **Drill-Across:** Compare Client\_Transaction with Client\_Wise\_Allocation to analyze client spending vs investment allocation

```
R1 ← JOIN (Client_Transaction, Client_Wise_Allocation USING Client_ID)
```

```
R2 ← ROLLUP* (R1, Client_ID → Client_ID, SUM(Client_Transaction.Amount) AS Total_Spending, SUM(Client_Wise_Allocation.Investment_Amount) AS Total_Investment)
```

6. **Pivot:** Pivot Client\_Transaction to view transaction totals by client instead of by year

```
R1 ← ROLLUP* (Client_Transaction, Date → Year, SUM(Amount) AS Total_Amount)
```

R2 ← PIVOT (R1, Rows: Client\_ID, Columns: Year, Values: Total\_Amount)

7. **Drill** : Through: Drill-through from portfolio returns to see the daily transactions contributing to the overall performance

R1 ← SELECT \* FROM Return\_Analytics

R2 ← DRILLTHROUGH (R1, Client\_Transaction WHERE Client\_Transaction.Date = Return\_Analytics.Date AND Client\_Transaction.Client\_ID IN (SELECT Client\_ID FROM Generates WHERE Portfolio\_ID = Return\_Analytics.Portfolio\_ID))

8. **Ranking Rank**: Rank Client\_Wise\_Allocation by Percentage\_Gain\_Loss to find the top 10 clients by profit percentage

R1 ← RANK (Client\_Wise\_Allocation, ORDER BY Percentage\_Gain\_Loss DESC)

R2 ← SELECT TOP 10 \* FROM R1

9. **Drill – Across with Conformed Dimensions**: Use Company to drill across Price and Portfolio\_Holding to analyze the impact of price changes on portfolio holdings

R1 ← JOIN (Price, Portfolio\_Holding USING ISIN)

R2 ← ROLLUP\* (R1, ISIN → ISIN, Date → Date, SUM(Price.Price) AS Total\_Price, SUM(Portfolio\_Holding.Quantity) AS Total\_Quantity)

#### D. Primary Events :

These **primary events** will correspond to the **fact tables** in your model, representing significant activities that drive data generation.

Primary Event	Fact Table	Key Measures	Dimensions Involved
Portfolio Creation Event	Portfolio_Creation	Portfolio Details	Client_Profile, Technical_Strategy, Fundamental_Report, Time
Client Transaction Event	Client_Transaction	Transaction Amount	Client_Profile, Time
Portfolio Holding Update Event	Portfolio_Holding	Asset Quantity	Portfolio_Creation, Company, Time
Company Price Update Event	Price	Stock Price	Company, Time
Portfolio Return Analysis Event	Return_Analytics	Benchmark Return, Portfolio Return	Portfolio_Creation, Time
Client Allocation Update Event	Client_Wise_Allocation	Investment Amount, Current Value, Gain/Loss	Client_Profile, Portfolio_Creation, Time

## E. Schema Implementation in Postgres:

