

# Data Warehousing

*by* Iwan Dyke

---

**Submission date:** 11-Mar-2024 12:05AM (UTC+0000)

**Submission ID:** 226241181

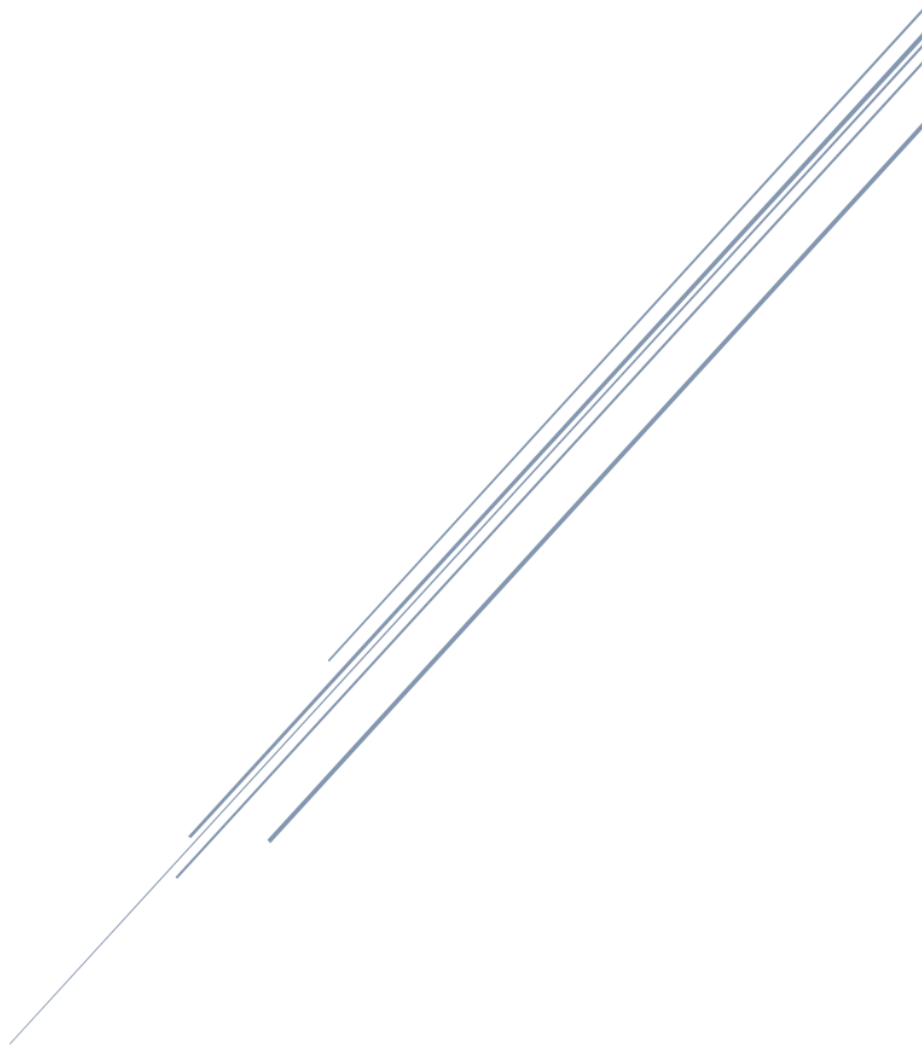
**File name:** 24865\_Iwan\_Dyke\_Data\_Warehousing\_565908\_350088734.docx (889.71K)

**Word count:** 3770

**Character count:** 20083

# DATA WAREHOUSING AND DATA MINING

ACCA6026



Iwan Dyke  
UWTSD

## Contents

|                                                         |           |
|---------------------------------------------------------|-----------|
| Introduction .....                                      | 2         |
| OLTP vs. Data Warehouse .....                           | 2         |
| Data Warehouse Architecture and ETL .....               | 3         |
| Extract .....                                           | 3         |
| Transform .....                                         | 3         |
| Load .....                                              | 3         |
| Data Marts for Management .....                         | 4         |
| Managing Director .....                                 | 4         |
| Finance Director .....                                  | 4         |
| Sales and Marketing Director .....                      | 4         |
| Normalized vs. Dimensional Structures .....             | 5         |
| Normalised Model .....                                  | 5         |
| First Normal Form .....                                 | 5         |
| Second Normal Form .....                                | 5         |
| Third Normal Form .....                                 | 5         |
| Dimensional Model .....                                 | 6         |
| <b>Source System Identification .....</b>               | <b>6</b>  |
| <b>Star Schema Design .....</b>                         | <b>7</b>  |
| Data Dictionary .....                                   | 7         |
| Business Dictionary .....                               | 7         |
| Event Log Table .....                                   | 8         |
| Incorporation of ETL_LoadDate .....                     | 8         |
| Data Warehouse Design (Aggregate Tables) .....          | 8         |
| <b>Data Marts Schema .....</b>                          | <b>10</b> |
| <b>Simplified ETL Plan .....</b>                        | <b>10</b> |
| Extract .....                                           | 10        |
| Transform .....                                         | 11        |
| Load .....                                              | 11        |
| <b>Data Quality and Governance .....</b>                | <b>11</b> |
| Conclusion .....                                        | 11        |
| Bibliography .....                                      | 11        |
| <br>Figure 1 (ETL Process) .....                        | <br>4     |
| Figure 2 (Normalised Properties Database Example) ..... | 5         |
| Figure 3 (Dimensionalised Properties Database) .....    | 6         |

|                                                         |    |
|---------------------------------------------------------|----|
| Figure 4 (Identified Source Systems) .....              | 6  |
| Figure 5 (Metadata Tables) .....                        | 7  |
| Figure 7 (Example Aggregated Data Warehouse Facts)..... | 8  |
| Figure 6 (Example Customer Data Mart) .....             | 10 |

## Introduction

In a world where data is essential in strategic decision making, the Cambrian Real Estate Company need to develop their data infrastructure to continue moving forward. The difference between Online Transactional Processing (OLTP) systems and the reporting-based functionality of a Data Warehouse is explored in the context of the Cambrian Real Estate Company. The Data Warehouse architecture aided by the ETL processes combines and utilises different data sources to make a comprehensive view of the data landscape. The construction of Data Marts will be exhibited around specific data specialisations in line with the organisation's strategic goals.

## OLTP vs. Data Warehouse

OLTP (Online Transactional Processing) systems are databases specifically designed and optimised for capturing and storing transactional data within an RDBMS (Relational Database Management System). Examples of these could be SQL Server, Oracle DB or MySQL. OLTP systems usually contain data which follows a normalised data model minimising redundancy, optimising for data storage and integrity. This type of database design can process concurrent data transactions while maintaining the integrity of the database. All databases should follow the ACID principles of database design:

**Atomicity:** All commands within a transaction should be treated as a single entity and will either succeed or fail. In the event of an unplanned interruption such as a system crash or power failure the integrity of the database can be maintained.

**Consistency:** All changes contained within a transaction follow the set constraints of the database. All data must follow the rules, constraints, datatypes and triggers of a database.

**Isolation:** Transactions should be able to run concurrently as they should be run in isolation. Allowing for different commands to affect different tables without interference. If a change is being made concurrently to the same table, one must wait for the other to finish.

**Durability:** Any insertions, updates or deletes are persisted within the database even when the system fails or has no power.

In contrast Data Warehouses are not concerned with data redundancy and are more focused on efficient querying and analytical processing. A dimensional model is used within a data warehouse sorting the data into facts and dimensions. Facts hold the measurable quantitative data such as dates or integers while dimensions hold context about the transactions such as strings or character fields [1]. There is a myriad of benefits to reporting using a Data Warehouse compared to an OLTP system:

**Increased Operational Efficiency:** Complex queries can be processed by the Data Warehouse not only improving speed and performance of the querying but reducing stress on the OLTP system. Data Warehouses also do not need to handle real-time data minimising impact on operational system.

**Improved Decision Making:** Consolidated, aggregated data enables the option of trend analysis, forecasting and strategic data views. Allowing for understanding of long-term market trends or customer preferences aiding in marketing campaigns.

**Enhanced Business Intelligence:** By collecting and joining data from multiple different sources a unified view of the company can be generated and analysis to be conducted. This would not be possible with an OLTP system alone.

**Data Quality and Governance:** When transforming the data through the ETL (Extract Transform Load) or ELT (Extract Load Transform) process the data should be cleaned and checked. This significantly improves data quality and can support better governance. Databases should all have data dictionaries holding important metadata. Allowing the lineage to be traced, personally identifiable fields to be tracked and data sources logged.

While OLTP systems provide essential data processing for day-to-day operations a Data Warehouse allows the collected data to be analysed and put to strategic use for the betterment of the business.

## Data Warehouse Architecture and ETL

A data warehouse is a collection of databases and the data processing technique which connects them all together is called an ETL. This stands for extract transform and load. In recent years with the introduction of new technologies to the data landscape this has also been altered to ELT swapping the placement of the transform and load steps.

### Extract

A Data warehouse has three distinct steps in which the data is processed. The data must first come from a source system, this could be one of the internal OLTP systems which are used within the business or taken from one of the external systems [2]. It is then taken to a staging database or alternatively a staging server depending on the amount of data required for processing. A staging area is where the data is loaded so that in the event of a complete data refresh through the system the OLTP server will not need to be used slowing its performance down. The data is stored in the staging area and mirrors that of the source system.

### Transform

It is then aggregated and loaded into the Data Warehouse. The data is transformed based on business needs. Aggregates could be collected; different data sources could be joined together to generate business insights, or operational or strategic views of data can be created.

### Load

Data can be loaded into data marts which can be considered as small specific data clusters answering a specific business question or need [3]. For example, a data mart based on the property listings could be created and linked with the financial database data to create an asset and asset valuation data mart. This could be used for forecasting property valuations over time.

The final step but is not guaranteed is visualising this data in a way that is easily digestible and visually appealing. This could be done using a reporting tool such as SSRS (SQL Server Reporting Services), PowerBI or Tableau. This can help the stakeholders easily understand the data from a glance and make business decisions.

Very often Data Warehouses are loaded once daily processing the previous days data. In certain instances, with new cloud technologies and things such as stream analytics it is possible for real-time or close to real-time data reporting. Which is high specialised and not required for most operations.

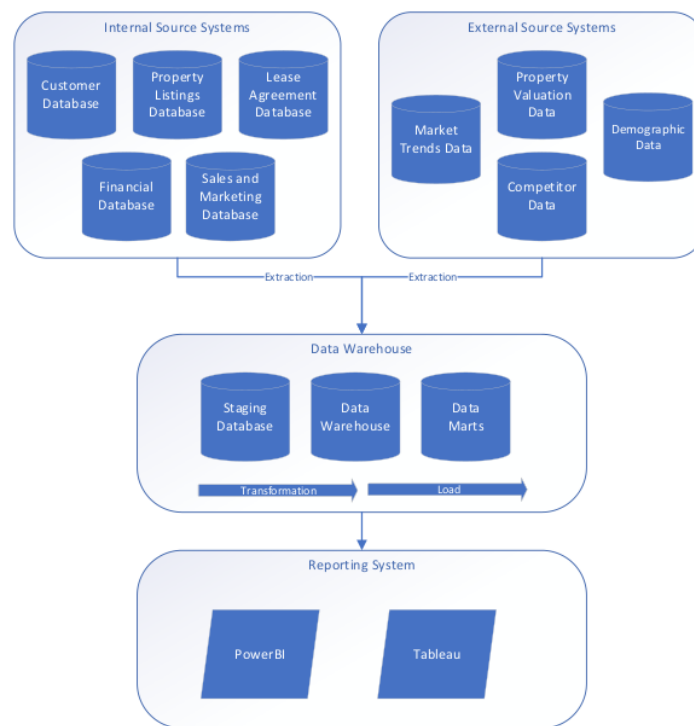


Figure 1 (ETL Process)

## Data Marts for Management

As mentioned above data marts are a specific subset of aggregated data and can be department, concept or role specific. It provides targeted data which meets the specific needs of a person or department. [4]

### Managing Director

A managing director is one of the highest-ranking employees of the business, they would need a comprehensive view of every aspect of the business and performance across all operational areas. They would need access to customers satisfaction data, aggregated sales data, and operational efficiency metrics for the business. This would aid within strategic planning and understanding the overall state of the business.

### Finance Director

The finance director would mainly be interested in a financial data mart with such metrics as revenue, expenses, cash flow and profitability. This would aid in understanding and being able to budget, planning for the fiscal year and identifying cost savings. By analysing trends over time, they would be able to manage risks to the industry more effectively.

### Sales and Marketing Director

The sales and marketing director would be responsible for marketing strategies, customer engagement and sales tactics. To aid with this a data mart which collates customer demographics, sales transactions, marketing campaign results and market analysis. This would help with creating targeted marketing strategies based on customer demographic engagement. Insights on sales

performance and market trends can also inform product investment and ensuring supply and demand can be met.

## Normalized vs. Dimensional Structures

There are two main types of data modelling which are prevalent in industry. Normalisation for OLTP systems getting data into a database, and Dimensional modelling for reporting purposes getting the data back out.

### Normalised Model

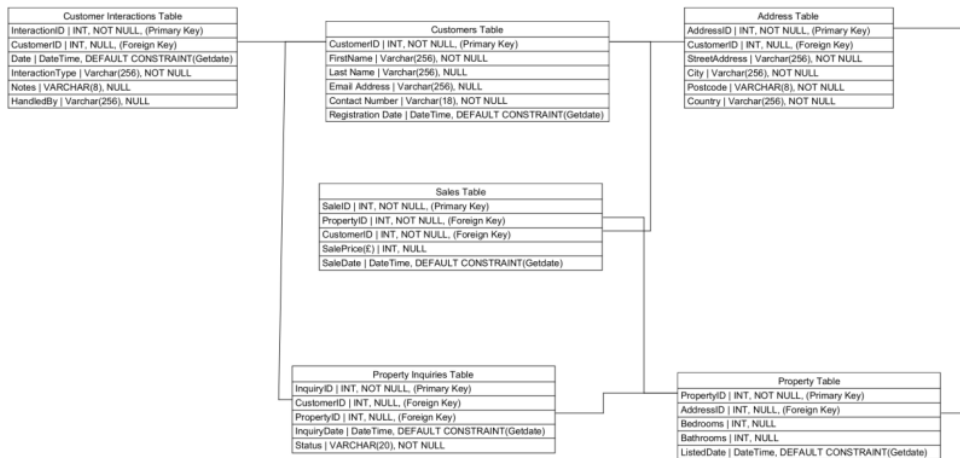


Figure 2 (Normalised Properties Database Example)

A normalised database model aims to reduce data redundancy for efficient storage, ensuring data integrity, and optimisation of transactional processing [5]. The tables above have been separated into 3NF (Third Normal Form), there are up to five normal forms within this data model, but industry standard considers data to be normalised when it matches the requirements for the third. Tables are connected via primary and foreign keys called relationships. The rules for normal forms are as follows:

#### First Normal Form

**Atomicity:** Each cell in the table must contain an atomic value, one piece of data per cell.

**Uniqueness:** Each row must be unique, the main method this is done uses a primary key.

**No Repeating Groups:** The table must not contain repeating groups of columns.

#### Second Normal Form

**Meet all 1NF Requirements:** The table must be in 1NF.

**No Partial Dependencies:** All non-key attributes must be fully functionally dependent on the primary key. This means that no attributes should depend on only a part of a composite primary key.

#### Third Normal Form

**Meet all 2NF Requirements:** The table must be in 2NF.

**No Transitive Dependency:** A non-key attribute must not be dependent on another non-key attribute. All attributes must be explicitly dependent on a primary key.

### Dimensional Model



Figure 3 (Dimensionalised Properties Database)

Dimensional Modelling opposes Normalisation as a method of data modelling which has an emphasis on reporting, efficient querying and analysis [1]. Data is split into FACT and Dimension tables. FACT tables hold measurable quantifiable metrics such as keys, dates, flags and integers. Dimension tables hold contextual descriptive data such as strings. The data model as in figure 3 should resemble a star and this is aptly named a star schema. A central fact table with dimensions surrounding. This is the most efficient method of optimising for querying, some more complex models use a snowflake schema in which a dimension can be joined to a dimension, but this greatly reduces efficiency and is not recommended.

To make this data easily accessible a reporting view should be created off this data which has all business logic and fields contained within. No processing of data should be done within reporting tools as this can cause significant performance issues.

A Data Warehouse would benefit Cambrian Real Estate by providing the functionality of insight to sales trends, property valuations, customer preferences based on demographics and operational dashboards and would aid with business level strategic decision making.

### Source System Identification

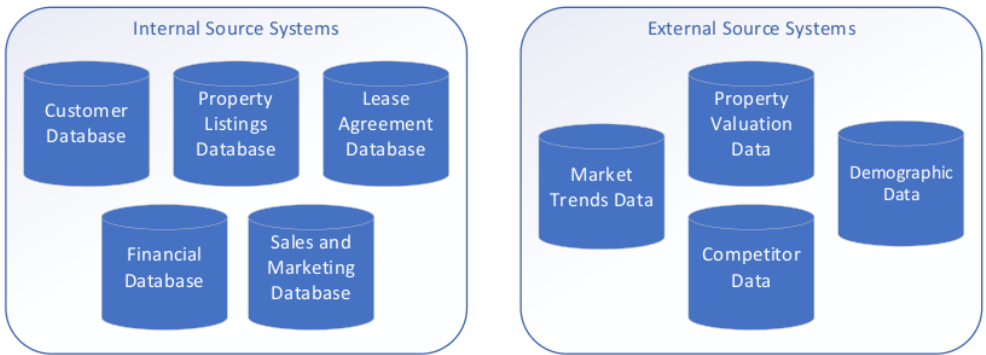


Figure 4 (Identified Source Systems)

The source systems can be split out into internal and external data sources. The internal data sources are from systems which are managed by Cambrian Real Estate whereas the external sources could be open source or government statistics data.

These source systems would be integrated into a staging area through an ETL system. By staging the data separate from the source system it creates an easy method of reloading data in the case of a complete system error. Depending on the method of processing the ETL it could also speed up the ETL processing time by accessing directly from the staging and not the source.

## Star Schema Design

| EventLog (Audit Table)                | Column_Dictionary (Data Dictory)                            |
|---------------------------------------|-------------------------------------------------------------|
| EventID   INT, NOT NULL (Primary Key) | Name   Varchar(256), NOT NULL (Composite Primary Key)       |
| UserID   Varchar(256), NOT NULL       | LogicalName   Varchar(256), NOT NULL                        |
| Command   Varchar(64), NOT NULL       | LogicalDescription   Varchar(256), NULL                     |
| SQLCommand   Varchar(MAX), NOT NULL   | DataType   Varchar(64), NOT NULL                            |
| DateTime   DateTime, NOT NULL         | Bytes   INT, NULL                                           |
|                                       | Precision   SmallInt, NULL                                  |
|                                       | Scale   SmallInt, NULL                                      |
|                                       | ColumnID   INT, NOT NULL                                    |
|                                       | ObjectName   Varchar(256), NOT NULL (Composite Primary Key) |
|                                       | ObjectID   int, NOT NULL (Composite Primary Key)            |
|                                       | SchemaName   Varchar(127), NOT NULL                         |
|                                       | OpenDate   Date, NOT NULL (Composite Primary Key)           |
|                                       | ClosedDate   Date, NULL                                     |
|                                       | Lineage   Nvarchar(4000), NULL                              |
|                                       | ContainsPII   BIT, NOT NULL                                 |

| BusinessDictionary (Audit Table)               |
|------------------------------------------------|
| BusinessSystemID   INT, NOT NULL (Primary Key) |
| SourceSystem   Varchar(256), NOT NULL          |
| DataOwner   Varchar(256), NOT NULL             |
| DataSteward   Varchar(256), NOT NULL           |
| DataDomain   DateTime, NOT NULL                |

Figure 5 (Metadata Tables)

Metadata is a vital part of any Data Warehouse; having a robust and comprehensive metadata management strategy involves multiple different tables and data collection systems. This follows classic data governance principles improving data transparency and will improve data process and efficiency.

### Data Dictionary

A data dictionary is one of the first tables which should be created on any database [6]. It should have an automated stored procedure or function behind it which runs daily to UPSERT (INSERT and UPDATE) the available metadata for all objects from system tables. The logical fields, lineage and PII (Personally Identifiable Information) fields would need to added via a human. However, there are new technologies such as Microsoft Purview and Databricks with Unity Catalog that use AI to fill these out automatically.

Data Dictionaries are a bridge between development and data governance and literacy within the organisation. It catalogues all data elements in a warehouse with their meanings, relationships and usage within the organisation. From a business perspective it promotes data consistency across reports and analysis removing ambiguous or misunderstood data definitions.

### Business Dictionary

A business dictionary tracks the lineage of data and aids in data ownership and stewardship. Within complex environments having data stewards and data ownership aids in data quality and compliance. The table facilitates accountability and provides clear communication routes.

## Event Log Table

The Event Log table is an audit table trail for all significant actions performed on the data warehouse. A database trigger will run a set of code after a certain event occurs. In this case when a DDL (Data Definition Language) or DML (Data Manipulation Language) is run in the warehouse it will be logged in a table. This aids in auditing, security, compliance and real-time monitoring. This log identifies unauthorised or erroneous data changes and what code has been run to roll-back.

## Incorporation of ETL\_LoadDate

An ETL load date can be incorporated into the data to provide a timestamp of when data was loaded. It allows for data management when dealing with incremental loads and data archiving. Auditing data process timelines and lineage.

## Data Warehouse Design (Aggregate Tables)

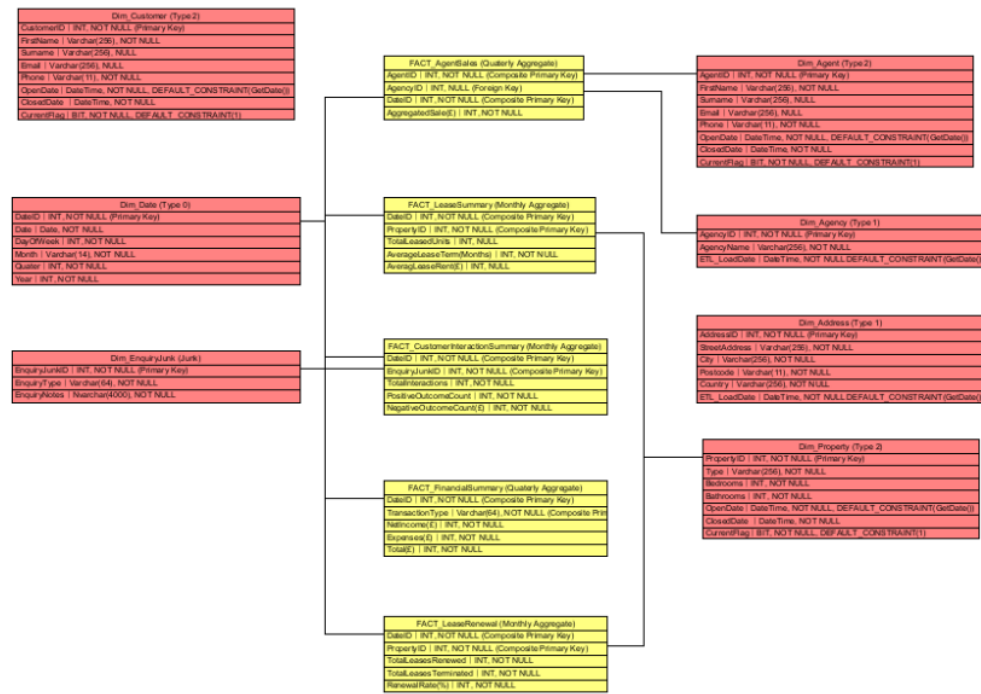


Figure 6 (Example Aggregated Data Warehouse Facts)

Aggregate tables are a summarised view of data created using detailed base data from FACT tables. Aggregate tables store precomputed totals and averages of a granular event. The main aim of this is improves query performance, as it reduces the amount of data that must be processed to answer a question.

Having a separate aggregate FACT table opposed to basing them off existing granular FACT tables optimise performance. Pre-aggregating the data as part of the ETL process reduces processing time for end-user queries as they are not done ad-hoc. In cases where significantly large amounts of data are being processed this is highly beneficial as real-time queries can hamper performance. Example aggregate tables based on the available source data have been created looking at lease and sale summaries and aggregated customer interaction data for marketing and business image analysis.

The ERD in figure 7 is a Dimensionalised Star schema showing an extract of a Data Warehouse model. Yellow tables represent FACT tables in this case many of them being aggregates, and red tables represent dimensions. There are many types of dimensions which can be used when modelling after Kimballs dimensional model, a few have been used in this example:

#### *Type 0 Dimension*

A type 0 dimension is a static data table, which almost no expected changes to the data context. Examples of this could be a date dimension, as dates are not going to change once the data is in the table it should not be updated or amended. This dimension should be checked on each incremental load to look for new data, if it is found it should be validated and added to the dimension.

#### *Type 1 Dimension*

A type 1 dimension is a rapidly changing dimension, where a history of changes is not kept, and data is highly likely to be changed and amended after each incremental load. This can be fully automated with any new or changed data added per load. This has been attributed to the address dimension as addresses cannot change. People who live or work at an address can change which address they are associated with, but the actual granular address data will not change.

#### *Type 2 Dimension (SC2)*

Type 2 Dimensions are considered slowly changing dimensions, they maintain a history of a records previous state and should be checked and updated on each incremental load. If a new change to a record is found, the previous record should be closed off with a flag and a date, and a new record opened. This can be used for point in time reporting as historical data can still be accessed and attributed to the FACT.

#### *Junk Dimension*

A junk dimension is a collection of miscellaneous data or low cardinality flags or indicators. There are multiple methods of using a Junk dimension. There can be one per mart or warehouse, in this case we have one per mart or data subset where we store large amounts of free text data in the form of customer enquiry logs. This is best stored in a separate table to the enquiry dimension as it can have its own indexing and statistics improving query performance.

#### *Star Schema Design Considerations*

A star schemas design is so that each table should only be one join away from the data table increasing optimisation and data visibility. Each dimension table should provide a certain set of business context grouped together, address information, customer information or agency information.

## Data Marts Schema

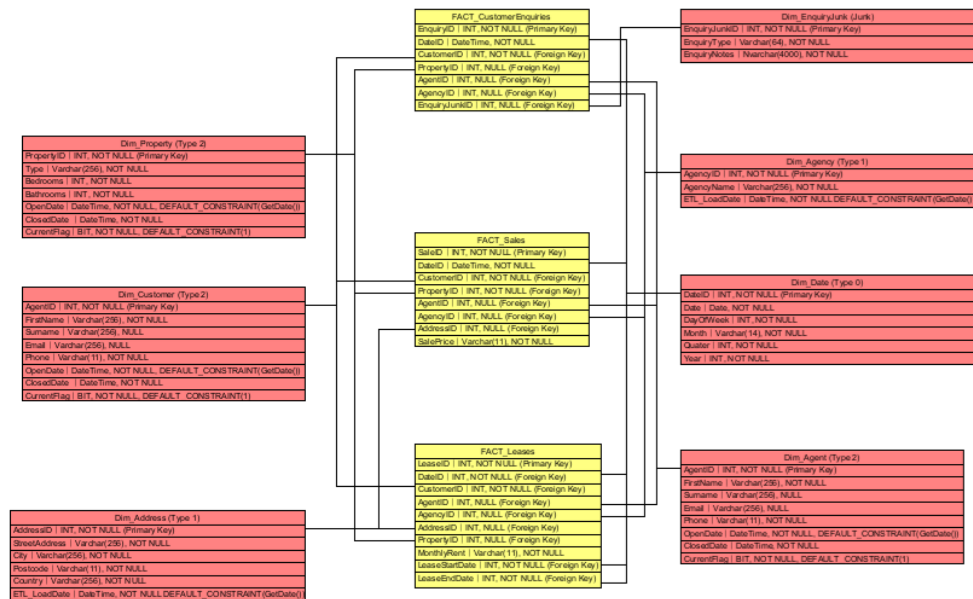


Figure 7 (Example Customer Data Mart)

Figure 6 shows an example customer data mart, the dimensions here can be considered conformed dimensions, as they will likely be in multiple data marts and be a part of the overarching data warehouse. These could be within a conformed schema, quickly showing it is part of multiple ETL processes or alternatively have extended properties on the table marking it as a conformed dimension.

The FACT tables are also at their lowest granularity of transactional data, all leases, all sales and all customer enquiries. Aggregated tables could be built off these however it would be best to create REPORTING-Views. Which hold business logic and sit on top of the fact tables. Views are not persisted objects and do not store data within them. For simple queries which would not process large amounts of data a view is best practice. When the aggregation starts to affect query performance of the warehouse an aggregated table like shown in figure 5 should be created with an incremental load to reduce performance hits.

A data mart was pioneered by Ralph Kimball who decided that opposed to Bill Inmons' method of completely planning and implementing a full warehouse from the start. [7] By creating smaller data marts and building off subsets of data, business output can be reached faster, and in a world of changing requirements the iterative and block nature of a data mart is more flexible and provides correct business logic for longer. As by the time a full data warehouse is planned or built in a larger organisation many of the requirements may have changed making it obsolete.

## Simplified ETL Plan

### Extract

An ETL stands for Extract Transform Load with the aim of receiving data from various internal and external sources. The internal databases such as sales, leases and financial transaction data can be accessed on off peak hours minimising impact to operational services. Larger databases can be

incrementally imported daily checking for deltas (new and changed data). The external sources will likely require API calls or direct database connections.

### Transform

Often data is not clean when it is imported to a Data Warehouse, it will need to be cleaned and standardised during the ETL process. Data should be checked for common data quality issues such as data inaccuracies and inconsistencies. It should also be standardised into consistent data formats ensuring accuracy and reliability. In some cases, data can be aggregated for fact tables ready for loading.

### Load

The data warehouse then needs to be loaded with the transformed data. The dimension tables need to be loaded first with the appropriate cleansed data ensuring that all foreign key relationships are available for populating the FACTs. Best practice is to check and load dimensions in order of their types. Once all dimensions are loaded, the FACT tables need to be loaded next. Where possible bulk loading can be used to optimise performance, or incremental loads for larger data operations.

## Data Quality and Governance

Data Governance is an integral part of any data warehousing work, and the Data Engineering/Analysts and Data Governance teams should collaborate frequently. Standards should be created for data profiling and master data management. This should all be held in a shared area for all people who work with data to access. A master data source system should be identified, and this should be synchronised across all systems [8].

The Data Governance team should develop and create data policies to standardise data use throughout organisations. They should identify and train data stewards and owners who will be responsible for management of certain datasets. A data management lifecycle should also be developed and disseminated throughout the organisation defining the archiving, retention and data purging in line with GDPR and data protection regulations.

## Conclusion

The establishment of a Data Warehouse at Cambrian Real Estate can only improve the decision making, analytical capabilities and operational functionality of the business. ETLs and Data Warehousing are pivotal in utilising stored data. It offers a competitive advantage in the real estate market and uplifts and informs decision-making. By creating a Data Warehouse and associated data marts it will not only improve the data governance and quality of the business it will yield a comprehensive platform for analysis and customer insight.

## Bibliography

- [1] R. Kimball and M. Ross, *"The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modelling"*. 3rd ed. 2011.
- [2] E. Rahm and H. H. Do, *"Data cleaning: Problems and current approaches"* IEEE Data Engineering Bulletin, vol. 23, no. 4, pp. 3-13, 2000.
- [3] P. Vassiliadis, *"A survey of extract-transform-load technology"* International Journal of Data Warehousing and Mining, vol. 5, no. 3, pp. 1-27, Jul.-Sep. 2009.

[4] M. Golfarelli, S. Rizzi, and I. Cella, "Beyond data warehousing: What's next in business intelligence?" in *Proc. 7th ACM Int. Workshop on Data Warehousing and OLAP*, Washington, DC, USA, Nov. 2004, pp. 1-6.

[5] Beynon-Davies, P. (2004). "Normalisation. In: *Database Systems*". Palgrave, London.  
[https://doi.org/10.1007/978-0-230-00107-7\\_18](https://doi.org/10.1007/978-0-230-00107-7_18)

[6] P. P. Uhrowczik, "Data Dictionary/Directories" in *IBM Systems Journal*, vol. 12, no. 4, pp. 332-350, 1973, doi: 10.1147/sj.124.0332.

[7] W. H. Inmon, *Building the Data Warehouse*. 4th ed. 2005.

[8] Harrison T, F. Luna-Reyes L, Pardo T, De Paula N, Najafabadi M and Palmer J. *The Data Firehose and AI in Government*. *Proceedings of the 20th Annual International Conference on Digital Government Research*. (171-176). <https://doi.org/10.1145/3325112.3325245>

# Data Warehousing

## GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

80/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13