

APRIL 3, 2023

# SPACES AND THE CHEN NOTATION

DAT601 ASSIGNMENT 1

KIRA BYRNE

ID# 13509995

NMIT

## Table of Contents

|  |           |
|--|-----------|
| <b>DATA MODELLING IN INFORMATION SYSTEMS .....</b> | <b>2</b>  |
| <b>CONCEPTUAL MODELLING .....</b>                  | <b>3</b>  |
| <b>SPACES CONCEPTUAL MODEL .....</b>               | <b>8</b>  |
| <b>DATABASE DICTIONARY .....</b>                   | <b>11</b> |
| <b>BUSINESS RULES .....</b>                        | <b>24</b> |
| <b>REFERENCES .....</b>                            | <b>26</b> |

# DATA MODELLING IN INFORMATION SYSTEMS

## Introduction

Data modelling is used to create a visual and logical representation of all components being recorded and stored in a proposed or existing database (Corbo, 2023). The main components seen in a data model are:

- **Entities** – A category that holds a set of data (e.g. Student, Order, Train)
- **Attributes** – A definitive labelling of the data that is being stored within an entity (e.g. A STUDENT has a *student id*, *first name*, *last name*, *email*, etc).
- **Relationships** – The connection two entities have with each other. A relationship should clearly show the cardinality and optionality each entity has within a connection.

Each entity and their attributes are the essence of a database. Between them are the relationships that form the structure and narrative. Relationships define not only the connections but the optionality and cardinality between entities.

Take the example below, a database for the New Zealand Land Transport Agency (NZTA). A road worthy car that is being sold in a legal manor must be registered with the (NZTA), who have many (N) cars within their database. A car does not always have a specific owner, such as if the car was in a dealership, it is not registered to an owner, rather a business. However, an owner of a car must register themselves with NZTA to legally own a car. An owner can have many cars, but only one (1) owner can be registered to a car within NZTA's database, in which they have many (N) members.

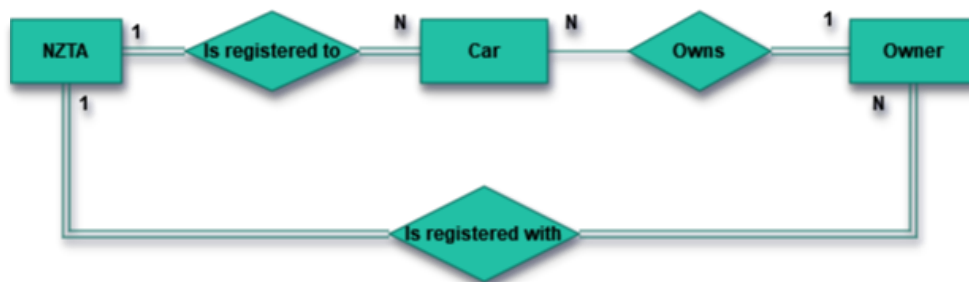


Figure 1: Car registration data model

Through data modelling, anyone within the project team should be able to understand the purpose and goals of a database. Typically, a proposed database will be modelled in stages as the project advances, beginning with highly visual and abstract drawings and ending with technical and physical representations as they would be seen from the perspective of software.

## Conceptual Modelling

When a database is proposed, in need of major changes, or lacks documentation, conceptual modelling is typically used as a visual guide to the architecture of a database in a “big picture” view. In the context of a proposed database (such as SPACES), this type of model is generated early in the project design in a way that can be easily understood by most viewers. Most importantly, anyone on the project team. It can be created on any drawing medium like a whiteboard, paper, or modelling software and have the ability to be quickly changed as it lacks complexity (DataAcademy.in, 2020). The goal of this early modelling is to showcase the conceptual structure of all entities and simple connective relationships with little technicality outside of the chosen style (ERD, CHEN, etc). It is not yet concerned with the data that is being stored, making it highly abstract and modular.

## Logical Modelling

Once entities and their connections have been established in the conceptual design, the project team can begin to model the fields of data that are to be stored. Logical models are typically developed as a platform-independent representations of the values an entity will have, the data that will be stored, and the relationships between them all (Rithka, 2022). The addition of keys is the logical focus of how the entities in a database are connected through primary and foreign keys. At this stage, only candidate keys are highlighted. These are values that can uniquely identify an individual set of data within a table, such as personal email, order number, or any other field that is entirely unique and traceable per dataset.

The creation of a logical model is often produced by a database architect and used by a business analyst to provide an organisation with insight pertaining to the limitation of the current technologies. (Agar, 2021), Logical models are made without foresight into software of coding language. Their purpose is to prepare a database architect for defining further detail to the modelling prior to implementation.

## Physical Modelling

The final stage of data modelling prior to the coded construction of a database is the physical model, which focuses on datatypes, their respective lengths, tables, and columns that provide the structure of the data an entity will hold. The design of the physical modelling considers the systems requirements and constraints (Keys, size, scalability, etc) and analyses the feasibility of the data systems proposed in the conceptual and logical models. This stage is the most crucial in the design of a proposed database, therefore, documenting the format in which the data will be physically stored is vital (cloud-base, SQL, NoSQL, hard drive, etc). Following from this, the database architect can tweak the concept of the database to meet the requirements of particular database management systems (DBMS) (Nalimov, 2021). The end goal of the physical model is to provide a finalised design that can be engineered into an implementable relational database.

## CONCEPTUAL MODELLING

### Chen's Extended Notation

An entity relationship diagram (ERD) is a method of modelling data that showcases all aspects of a database, primarily the entities, their attributes, and how they relate to one and other. An extension of the ERD is the Chen Notation. Made by Dr. Peter Chen in 1976, the purpose of the Chen Notation is to extend the logical model of an ERD through making changes to the:

- Associative shapes
- The complexity of entities
- How attributes are attached to an entity
- How relationships are shown
- Cardinality




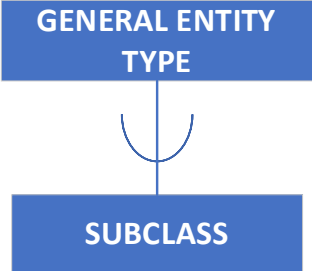
Chen Notation diagrams maintain the abstract look of a conceptual model while containing more detail similar to that of a logical model. In this style, entities have a level of complexity which allows a data analyst to further understand the context behind an entity, such as it's reliance or association with other entities.

As mentioned in the introduction, relationships make up the structure of a data model, providing connections and context between all entities. A standard ERD would use crow's foot methodology to

represent a relationship and its cardinality. In Chen Notation, a relationship is a diamond shaped textbox that connect to entities, with the cardinality shown through the connective lines (Dybka, 2014). This addition of text provides context for a relationship and forms a sentence to define it (e.g., “One OWNER has many DOGS”). The aim of Chen’s Notation is to construct a natural view of how the real world comprises entities and the relationships that exists between them (CS, Odessa, 2023).

## Chen Notation Glossary

### Entities

|   |  |
|---|--|
|    | <p>An ENTITY represents a thing which has attributes and relationships with a database.</p> <p>E.g. A student, teacher, class room</p>   |
|    | <p>A WEAK ENTITY represents a thing or instance in which its existence is dependant on a relationship with another entity</p> <p>E.g. A subscription cannot exist without a subscriber</p>   |
|  | <p>An ASSOCIATIVE ENTITY represents an instance that occurs in a many-to-many relationship.</p> <p>E.g. Many owners can own many houses, so the associative entity would be the ownership of a particular house.</p>   |
|  | <p>A SUBCLASS is a an entity that takes on attributes of a general entity type. This is known as generalisation and specialisation and helps to minimise redundant data (Yue, 2017). SUBCLASSES can be used to create a tier group of subclasses that extend from one after the other.</p> <p>E.g A CAR SERVICE has a GOLD service that includes everything from the SILVER and BRONZE services with extra values that would be outlined in the GOLD subclass entitiy.</p> |

## Attributes

### ATTRIBUTE

An ATTRIBUTE is a field detailing stored information about an entity.

E.g. A DOG has a BREED.

### KEY ATTRIBUTE

A KEY ATTRIBUTE is a primary attribute that uniquely identifies a singular entity.

E.g. An ACCOUNT HOLDER is identified by an ACCESS NUMBER.

### PARTIAL KEY ATTRIBUTE

A PARTIAL ATTRIBUTE is used to uniquely identify a weak entity through combination with the main entity's key attribute.

E.g. A SUPPLIER (e) with a SUPPLY CODE (a) processes an ORDER (w/e) that has an ORDER CODE (p/k/a).

### MULTIVALUED ATTRIBUTE

A MULTIVALUED ATTRIBUTE is made up of multiple values, each being distinct from others within a single table.

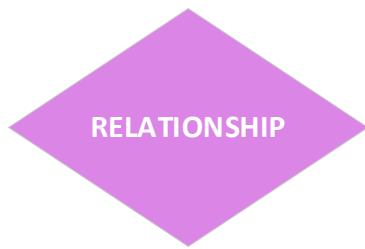
E.g. An ALBUM can fit many GENRE/s.

### DERIVED ATTRIBUTE

A DERIVED ATTRIBUTE represents an attribute that is calculated from the values derived from other attributes.

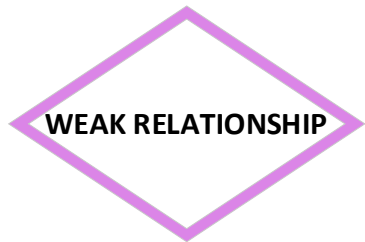
E.g. the total PRICE of an ITEM can be found when calculated with the GST and the DISCOUNT.

## Relationships



A RELATIONSHIP represents the connection between two entities. Usually, the text within the diamond shape provides context.

E.g. DEFENDANT *is registered to* TERRITORY



A WEAK RELATIONSHIP is used to identify a relationship between parent (strong) and child (weak) entities, where one entity's existence is dependant on another.

E.g. A CLIENT (s/e) *has a* SUBSCRIPTION (w/e)

## Optionality



A MANDATORY relationship connection represents a relationship that is solidified without optionality. The two entities must be related to each other but not all entries in an entity have total participation, only partial participation (Dybka, 2014).

E.g. A STUDENT **must** be enrolled in a COURSE, but a COURSE **does not** have to have any enrolled students.



An OPTIONAL relationship connection is represented with a dotted line. Entities with this connection are not required to have a relationship.

E.g. A PATIENT can *purchase* MEDICINE and can **optionally have** a PRESCRIPTION.



A TOTAL PARTICIPATION connection represents a relationship where all subjects in an entity must participate in a relationship with another entity.

E.g. An INPATIENT **must** be admitted to a WARD.

### Cardinality

#### **One to One 1:1**

A relationship where only one entity is in participation with only one other entity.

E.g. 1 PRINCIPAL directs 1 SCHOOL.

#### **One to Many 1:N**

A relationship where only one entity has multiple instances of participation with another entity.

E.g. 1 SUPPLIER makes many ORDERS

#### **Many to One N:1**

A relationship where an entity has multiple instances of participation with another entity, however that other entity on has one instance or participation.

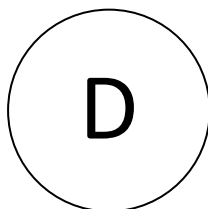
E.g. Many ACCOUNTS are registered to 1 OWNER.

#### **Many to Many M:N**

A relationship where both entities have multiple instances of participation. This relationship is usually represented as a it's own entity with attributes.

E.g. Many INGREDIENTS are used in MANY RECEPIES.

### Extended

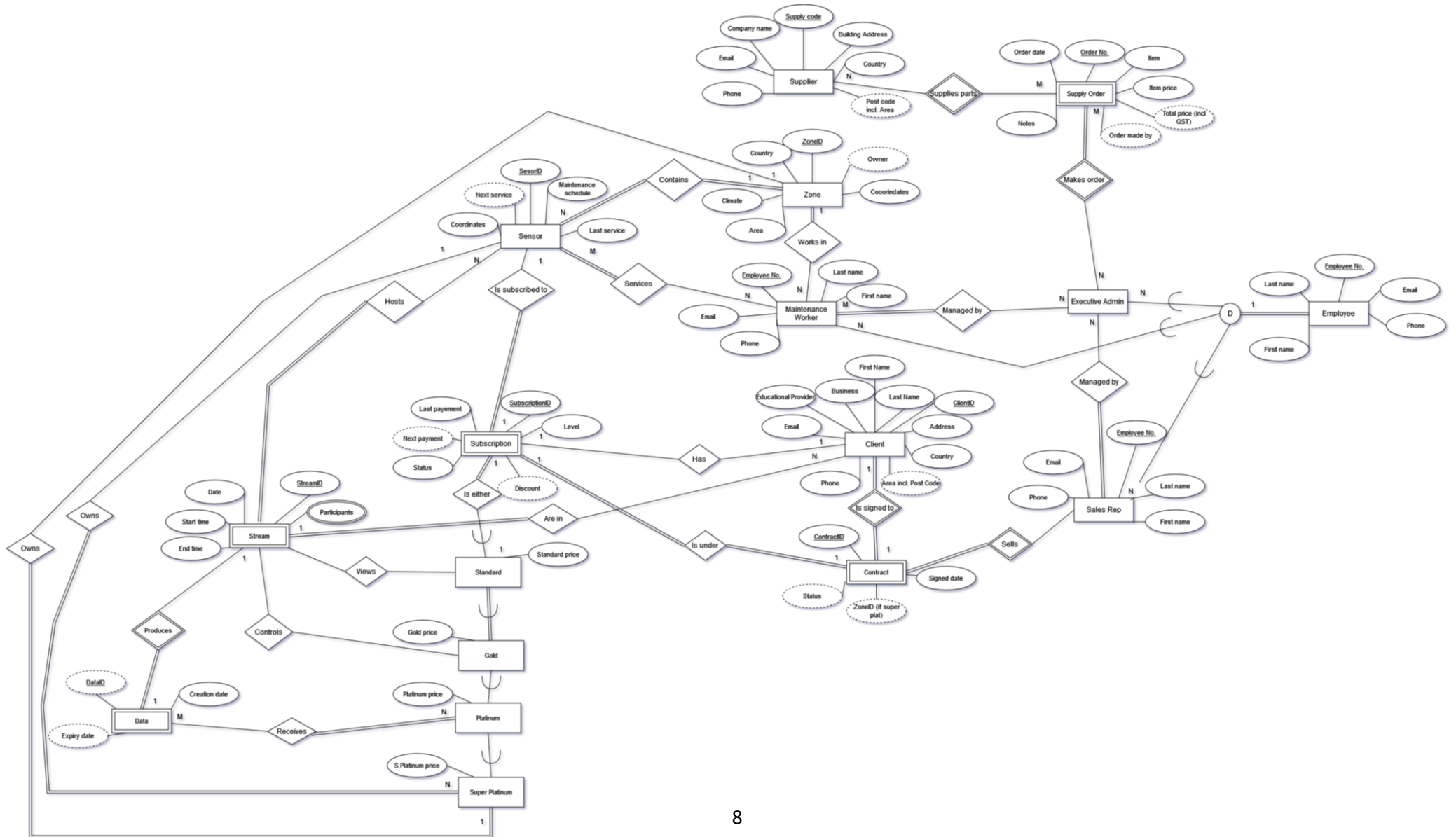


A DISTINCT icon is used to separate a set of subclasses from each other. It exists between the connect of a superclass and a subclass to show that the subclasses receive the given attributes but do not extend from each other. They are distinct.

E.g All EMPLOYEES have an ID, a name, and an email but their roles are distinct.



# SPACES CONCEPTUAL MODEL



## Diagram Rationale

All EMPLOYEEs of SPACES contain essential information about each individual including their unique id, their first and last names, and work-related contact details. Extending from the EMPLOYEE attributes are the three distinct roles.

1. An EXECUTIVE ADMIN is responsible for managing other employees and making SUPPLY ORDERS for the sensors which are fulfilled by SUPPLIERS. They have full view, update, and delete abilities within the database. Not every EXECUTIVE ADMIN manages all employees by all employees must be managed by an EXECUTIVE ADMIN.
2. A MAINTENANCE WORKER is responsible for maintaining sensors. They are permanently allocated to a zone where they service many sensors. With sensor data, they have view and update abilities.
3. A SALES REP is responsible for forming and maintaining client relationships. They sell and sign contracts, attend to client needs, organise subscriptions, discounts, and contact clients when contracts are nearing void status and/or deletion.

A CLIENT is an individual who has a CONTRACT with SPACES. Similar to the EMPLOYEE, the CLIENT table stores essential information about the individual that is relevant to the business structure. This includes, their full name, contact details, physical address, and if they are using SPACES for educational or business-related purposes. If they are, then they are entitled to a 3% discount on the price of their subscription.

A CLIENT discusses and signs a CONTRACT with a sales rep. The importance of the CONTRACT is that it stores long-term, stable information about the CLIENT and their SUBSCRIPTION. This information is only subject to change under certain circumstances such as a contract becoming void (status), CLIENT details changing, or the SUBSCRIPTION level changing. The CONTRACT is a weak entity as its existence is reliant on both the CLIENT and the SALES REP who are involved.

A SUBSCRIPTION stores both stable and transient information about a CLIENTs use of a SENSOR. This includes the level of SUBSCRIPTION, the ID of the sensor they are connected to, the status of the SUBSCRIPTION (subscribed/unsubscribed), if a discount applies, and the dates of the last subscription payment and the next due payment. Each SUBSCRIPTION is reliant on the validity of a CONTRACT (whether it is current or void), therefore making it a weak entity. Each SUBSCRIPTION has a level which defines the CLIENTs abilities with a sensor. The level exists in a tier formation, meaning the abilities of the level prior will continue to greater tiers.

- STANDARD, being the first level, is the least expensive and allows a CLIENT to view a stream.
- GOLD will allow the CLIENT to control the view of the stream, changing camera angles and being able to “look around” a ZONE.
- PLATINUM allows the CLIENT to receive recorded DATA after the STREAM is over.
- SUPER PLATINUM is negotiated when a CLIENT is forming a CONTRACT. This level of subscription allows the CLIENT to secure a private ZONE and SENSOR. The ownership of a ZONE is negotiated and recorded in the CONTRACT.

A STREAM is a live presentation, similar to a zoom call. When in the STREAM, it is a constant live video that cannot be rewound or fast forwarded. This can only happen once the STREAM has ended, and the SENSOR has exported a recording of the STREAM as DATA, receivable by PLATINUM and SUPER PLATINUM subscribers. A STREAM is reliant on at least 1 participating

client to exists, making it a weak entity. This extends to the DATA. If a STREAM does not exists, there is no DATA to output.

SENSORS are the physical hardware that are placed in ZONEs to host a STREAM and project the CLIENTs into a ZONE. Each SENSOR has a maintenance schedule that is adhered to by MAINTENANCE WORKERS. When a SUPPLY ORDER is made by an EXECUTIVE ADMIN, it contains information about the ID, contents, price, and reason for order including the ID of the staff member who made it. The SUPPLY ORDER is a weak entity as its existence only occurs when it is being made by an EXECUTIVE ADMIN.

A SENSOR is located by coordinates which are ascertained at the time of the SENSOR being installed in a ZONE. A ZONE is a specific area in which a SENSOR will be installed and used to host a STREAM. A ZONE is defined by its climates (urban, snowy, forest, etc). It identified with a unique ID, given general location data (county, area) as well as coordinates for specific locating. If a ZONE is owned by a SUPER PLATINUM subscriber, their CLIENT ID will be recorded as well.

## DATABASE DICTIONARY

### Entities

| Entity Name        | Description  | Aliases             | Occurrence |
|--------------------|--|---------------------|------------|
| Employee           | An employee of spaces.   | E.Employee          | Many       |
| Executive Admin    | Members of the executive administration team at SPACE. Responsible for managing sales reps, maintenance workers, and making supply orders.   | EA.ExecutiveAdmin   | Many       |
| Sales Rep          | Employee of SPACE that work with clients to procure contracts. Manage client contact.  | SR.SalesRep         | Many       |
| Maintenance Worker | Performs maintenance work on the sensors for a particular zone.  | M.MaintenanceWorker | Many       |
| Supplier           | Supplies the items for a supply order for SPACES.  | SU.Supplier         | Many       |
| Supply Order       | The order made by executive admins and fulfilled by a supplier.  | SO.SupplyOrder      | Many       |
| Zone               | An area where sensors are placed to create the environment for a stream.   | Z.Zone              | Many       |
| Sensor             | The device used to perform a stream. Subscribed to by clients.   | SE.Sensor           | Many       |
| Stream             | The live video performed by a sensor in a zone that clients partake in. A client's abilities vary within a stream depending on their subscription level.   | ST.Stream           | Many       |
| Data               | The recorded video after a stream is complete. Only accessible to platinum and super platinum subscribers.   | D.Data              | Many       |
| Client             | A customer with a contract with SPACES.  | C.Client            | Many       |
| Contract           | A contract between SPACES and a client that states their subscription details.   | CO.Contract         | Many       |
| Subscription       | A contracted subscription to a sensor. Prices vary depending on level of subscription. Each level will take on the abilities of the previous tier. Price can be discounted if client is using SPACES for business or educational purposes. | SB.Subscription     | Many       |
| Standard           | The base level of a subscription. Allows a client to partake in a stream   | S.Standard          | Many       |

|                |  |          |      |
|----------------|--|----------|------|
|                | on a public sensor.  |          |      |
| Gold           | Allows a client to control the view of a sensor while partaking in a stream. | G.Gold   | Many |
| Platinum       | Client receives recorded data from a stream.                                 | P.Plat   | Many |
| Super Platinum | Client can negotiate the ownership of a private zone and sensor.             | SP.Splat | Many |

### Entity Relationships

| Entity Name        | Cardinality | Participation | Relationship     | Participation | Cardinality | Entity Name        |
|--------------------|-------------|---------------|------------------|---------------|-------------|--------------------|
| Supplier           | 1           | Strong        | Provides items   | Strong        | 1           | Supply Order       |
| Employee           | 1           | Mandatory     | Is either        | Strong        | 1           | Executive Admin    |
| Employee           | 1           | Mandatory     | Is either        | Strong        | 1           | Sales Rep          |
| Employee           | 1           | Mandatory     | Is either        | Strong        | 1           | Maintenance Worker |
| Executive Admin    | 1           | Strong        | Makes order      | Total         | 1           | Supply Order       |
| Executive Admin    | N           | Strong        | Manages          | Total         | M           | Sales Rep          |
| Executive Admin    | N           | Strong        | Manages          | Total         | M           | Maintenance Worker |
| Maintenance Worker | N           | Strong        | Works in         | Total         | M           | Zone               |
| Maintenance Worker | N           | Strong        | Services         | Total         | M           | Sensor             |
| Zone               | N           | Total         | Contains         | Total         | M           | Sensor             |
| Sensor             | 1           | Strong        | Hosts            | Total         | M           | Stream             |
| Stream             | 1           | Strong        | Creates          | Total         | 1           | Data               |
| Sales Rep          | N           | Strong        | Sells            | Total         | M           | Contract           |
| Client             | 1           | Total         | Signs            | Total         | 1           | Contract           |
| Client             | 1           | Strong        | Is contracted to | Strong        | 1           | Subscription       |
| Client             | N           | Strong        | Are in           | Total         | 1           | Stream             |
| Subscription       | 1           | Total         | Is (either)      | Strong        | 1           | Standard           |
| Subscription       | 1           | Strong        | Is (either)      | Strong        | 1           | Gold               |
| Subscription       | 1           | Strong        | Is (either)      | Strong        | 1           | Platinum           |
| Subscription       | 1           | Strong        | Is (either)      | Strong        | 1           | Super Platinum     |

|                |   |        |             |        |   |        |
|----------------|---|--------|-------------|--------|---|--------|
| Standard       | 1 | Strong | Can view    | Strong | 1 | Stream |
| Gold           | 1 | Strong | Can control | Strong | 1 | Stream |
| Platinum       | 1 | Total  | Receives    | Strong | N | Data   |
| Super Platinum | 1 | Total  | Owns        | Strong | 1 | Sensor |
| Super Platinum | 1 | Total  | Owns        | Strong | 1 | Zone   |

| Entity Name            | Attributes  | Description   | Domain     | Aliases      | Composite | Derived                     | Nulls    | Key?          | Default Value  |
|------------------------|-------------|---|------------|--------------|-----------|-----------------------------|----------|---------------|----------------|
| <b>Employee</b>        | Employee No | The unique employee number used to identify individual employees of SPACES. | Int        | E.EmpNo      | N         | N                           | Not null | Primary Key   | Increment by 1 |
| <b>Employee</b>        | First Name  | The first name of the employee  | Char (50)  | E.firstname  | N         | N                           | Not null | N             | None/Not Null  |
| <b>Employee</b>        | Last Name   | The surname of the employee   | Char (50)  | E.lastname   | N         | N                           | Not null | N             | None/Not Null  |
| <b>Employee</b>        | Email       | The work email of the employee  | Char (100) | E.email      | N         | EA.Firstname<br>EA.lastname | Not null | Candidate key | None/Not null  |
| <b>Employee</b>        | Phone       | The work phone of the employee  | Char (20)  | E.Phone      | N         | N                           | Not null | Candidate key | None/Not null  |
| <b>Executive Admin</b> | Employee No | The unique employee number used to identify individual employees of SPACES. | Int        | EA.EmpNo     | N         | N                           | Not null | Primary Key   | Increment by 1 |
| <b>Executive Admin</b> | First Name  | The first name of the executive admin                                       | Char (50)  | EA.firstname | N         | N                           | Not null | N             | None/Not Null  |
| <b>Executive Admin</b> | Last Name   | The surname of the executive admin  | Char (50)  | EA.lastname  | N         | N                           | Not null | N             | None/Not Null  |

| Entity Name            | Attributes                 | Description  | Domain     | Aliases        | Composite | Derived                     | Nulls    | Key?          | Default Value |
|------------------------|----------------------------|--|------------|----------------|-----------|-----------------------------|----------|---------------|---------------|
| <b>Executive Admin</b> | Email                      | The work email of the executive admin                          | Char (100) | EA.email       | N         | EA.Firstname<br>EA.lastname | Not null | Candidate key | None/Not null |
| <b>Executive Admin</b> | Phone                      | The work phone of the executive admin                          | Char (20)  | EA.Phone       | N         | N                           | Not null | Candidate key | None/Not null |
| <b>Supplier</b>        | Supply Code                | Code used to identify a supply company on their order receipts | Int        | SU.SupplyCode  | N         | N                           | Not null | Primary Key   | None/Not null |
| <b>Supplier</b>        | Company Name               | Name of the supply company                                     | Char (100) | SU.CompanyName | N         | N                           | Not null |               | None/Not null |
| <b>Supplier</b>        | Email                      | General email address of the supply company                    | Char (100) | SU.Email       | N         | N                           | Y        | Candidate Key | None          |
| <b>Supplier</b>        | Phone                      | Phone number of the supply company                             | Char (20)  | SU.Phone       | N         | N                           | Not null | Candidate Key | None          |
| <b>Supplier</b>        | Address                    | Address of the Supply building or office                       | Char (100) | SU.Address     | N         | N                           | Not null | Candidate Key | None          |
| <b>Supplier</b>        | Country                    | Country the supplier is located in                             | Char (50)  | SU.Country     | N         | N                           | Not null | N             | None          |
| <b>Supplier</b>        | Area (including post code) | Area and postal code of the                                    | Char (50)  | SU.Area        | Y         | From Country +              | Not null | N             | None          |



| Entity Name         | Attributes                  | Description  | Domain     | Aliases       | Composite | Derived                                     | Nulls    | Key?        | Default Value                             |
|---------------------|-----------------------------|--|------------|---------------|-----------|---|----------|-------------|---|
|                     |                             | supplier   |            |               |           | Area  |          |             |   |
| <b>Supply Order</b> | Order Number                | Unique identifier of a supply order  | Int        | SO.OrderNo    | N         | N   | Not null | Primary Key | None                                      |
| <b>Supply Order</b> | Order Date                  | The date the order was made (DD-MM-YYYY)                                   | Date       | SO.Date       | N         |   | Not null | N           | The date the order was finalised          |
| <b>Supply Order</b> | Order made by               | The first and last name of the executive admin that made the order         | Char (100) | SO.MadeBy     | Y         | From EA.FirstName + EA.LastName             | Not null | N           | The name of the person writing the order. |
| <b>Supply Order</b> | Item                        | Names or descriptions of an item   | Char (100) | SO.Item       | N         | N   | Not null | N           | None                                      |
| <b>Supply Order</b> | Item price                  | Price of an individual item  | Decimal    | SO.ItemPrice  | N         | N   |          | N           | \$0.00                                    |
| <b>Supply Order</b> | Total Price (including GST) | Total price of the order including GST                                     | Decimal    | SO.TotalPrice | N         | GST derived from total price excluding GST. | Not null | N           | \$0.00                                    |
| <b>Supply Order</b> | Notes                       | Descriptive notes for the executive admin to use (e.g. order for sensorID) | Text       | SO.Notes      | N         | N   | Y        | N           | None                                      |
| <b>Sales Rep</b>    | Employee ID                 | The unique identifier of each  | Int        | SR.EmpID      | N         | N   | Not      | Primary     | Increment from 1                          |

| Entity Name               | Attributes  | Description  | Domain     | Aliases      | Composite | Derived | Nulls    | Key?          | Default Value    |
|---------------------------|-------------|--|------------|--------------|-----------|---------|----------|---------------|------------------|
|                           |             | sales rep  |            |              |           |         | null     | Key           |                  |
| <b>Sales Rep</b>          | First Name  | The first name of the sales rep                    | Char (50)  | SR.FirstName | N         | N       | Not null | N             | None             |
| <b>Sales Rep</b>          | Last Name   | The surname of the sales rep                       | Char (50)  | SR.LastName  | N         | N       | Not null | N             | None             |
| <b>Sales Rep</b>          | Email       | The employee email given to the sales rep          | Char (100) | SR.Email     | N         | N       | Not null | Candidate key | None             |
| <b>Sales Rep</b>          | Phone       | The employee phone number given to the sales rep   | Char (20)  | SR.Phone     | N         | N       | Not null | Candidate key | None             |
| <b>Maintenance Worker</b> | Employee ID | The unique identifier of each maintenance worker   | Int        | MW.EmpID     | N         | N       | Not null | Primary Key   | Increment from 1 |
| <b>Maintenance Worker</b> | First Name  | The first name of the sales rep                    | Char (50)  | SR.FirstName | N         | N       | Not null | N             | None             |
| <b>Maintenance Worker</b> | Last Name   | The surname of the maintenance worker              | Char (50)  | SR.LastName  | N         | N       | Not null | N             | None             |
| <b>Maintenance Worker</b> | Email       | The employee email given to the maintenance worker | Char (100) | SR.Email     | N         | N       | Not null | Candidate key | None             |
| <b>Maintenance Worker</b> | Phone       | The employee phone number given to the             | Char (20)  | SR.Phone     | N         | N       | Not null | Candidate key | None             |

| Entity Name   | Attributes             | Description                                  | Domain     | Aliases      | Composite | Derived   | Nulls    | Key?          | Default Value    |
|---------------|------------------------|--|------------|--------------|-----------|---|----------|---------------|------------------|
|               |                        | maintenance worker                           |            |              |           |   |          |               |                  |
| <b>Client</b> | Client ID              | The unique identifier of each client         | Int        | CL.ClientID  | N         | N   | Not null | Primary Key   | Increment from 1 |
| <b>Client</b> | First Name             | The first name of the client                 | Char (50)  | CL.FirstName | N         | N   | Not null | N             | None             |
| <b>Client</b> | Last Name              | The surname of the client                    | Char (50)  | CL.LastName  | N         | N   | Not null | N             | None             |
| <b>Client</b> | Email                  | Email address of the client                  | Char (100) | CL.Email     | N         | N   | Not null | Candidate key | None             |
| <b>Client</b> | Phone                  | Contact phone number of the client           | Char (20)  | CL.Phone     | N         | N   | Not null | Candidate key | None             |
| <b>Client</b> | Address                | Mailing address of the client                | Char (100) | CL.Address   | N         | N   | Not null | N             | None             |
| <b>Client</b> | Country                | Country the client resides in                | Char (100) | CL.Country   | N         | N   | Not null | N             | None             |
| <b>Client</b> | Area (incl. Post Code) | Area the client lives in include postal code | Char (100) | CL.Area      | N         | Postal code derived from address + area + country | Not null | N             | None             |
| <b>Client</b> | Educational Provider   | School/institution a client may be           | Char (100) | CL.Education | N         | N   | Y        | N             | None             |

| Entity Name     | Attributes      | Description  | Domain     | Aliases       | Composite | Derived   | Nulls    | Key?        | Default Value    |
|-----------------|-----------------|--|------------|---------------|-----------|---|----------|-------------|------------------|
|                 |                 | part of  |            |               |           |   |          |             |                  |
| <b>Client</b>   | Business        | Name of the business/company a client may be part of   | Char (100) | CL.Business   | N         | N   | Y        | N           | None             |
| <b>Contract</b> | Contract ID     | The unique identifier of each contract   | Int        | CO.ContractID | N         | N   | Not null | Primary Key | Increment from 1 |
| <b>Contract</b> | Signed Date     | Date the contract was made (signed off by both parties)  | Date       | CO.SignedDate | N         | N   | Not null | N           | DD-MM-YYYY       |
| <b>Contract</b> | Status          | Status of a contract (current or void)   | Char (15)  | CO.Status     | N         | From Subscription Status and Subscription Last payment date | Not null | N           | None             |
| <b>Contract</b> | Negotiated Zone | ID of the negotiated zone owned by a super platinum subscriber. Must be manually updated as contract does not connect to zone. | Int        | CO.Zone       | N         | ZoneID  | Nullable | N           | None             |

| Entity Name         | Attributes      | Description  | Domain    | Aliases     | Composite | Derived  | Nulls    | Key?        | Default Value                            |
|---------------------|-----------------|--|-----------|-------------|-----------|--|----------|-------------|--|
| <b>Subscription</b> | Subscription ID | The unique identifier of each subscription   | Int       | SB.SubID    | N         | N  | Not null | Primary Key | Increment from 1                         |
| <b>Subscription</b> | Level           | Level of the subscription (standard/gold/platinum/super platinum)                          | Char (20) | SB.Level    | N         | N  | Not null | N           | Standard                                 |
| <b>Subscription</b> | Discount        | A discounted percentage from the price if client is using SPACES for education or business | Percent   | SB.Discount | N         | From Client's educational provider or business | Y        | N           | 3% of subscription price                 |
| <b>Subscription</b> | Status          | Current status of the subscription (Subscribed/Unsubscribed)                               | Char (15) | SB.Status   | N         | N  | Not null | N           | None                                     |
| <b>Subscription</b> | Last Payment    | The last payment made by the client  | Date      | SB.LastPay  | N         | N  | Not null | N           | None                                     |
| <b>Subscription</b> | Next Payment    | The next due payment of a subscription   | Date      | SB.NextPay  | N         | Calculated as one month from last payment      | Not null | N           | One month from last payment (DD-MM-YYYY) |
| <b>Sensor</b>       | Sensor ID       | The unique identifier of each sensor   | Int       | SE.SubID    | N         | N  | Not null | Primary Key | Increment from 1                         |

| Entity Name   | Attributes           | Description   | Domain     | Aliases        | Composite | Derived   | Nulls    | Key?        | Default Value    |
|---------------|----------------------|---|------------|----------------|-----------|---|----------|-------------|------------------|
| <b>Sensor</b> | Coordinates          | The coordinates of a sensor   | GPS        | SE.Coordinates | N         | N   | Not null | N           | None             |
| <b>Sensor</b> | Maintenance schedule | The schedule for a sensor's maintenance (weekly, monthly, yearly depending on zone condition) | Char (15)  | SE.Maintenance | N         | N   | Not null | N           | None             |
| <b>Sensor</b> | Last service         | Date of the last service a sensor received  | Date       | SE.LastService | N         | N   | Y        | N           | DD-MM-YYYY       |
| <b>Sensor</b> | Next service         | Date of the next service a sensor receives  | Date       | SE.NextService |           | Calculated from Maintenance schedule + Last service | Y        | N           | DD-MM-YYYY       |
| <b>Zone</b>   | Zone ID              | The unique identifier of each zone  | Int        | Z.ZoneID       | N         | N   | Not null | Primary Key | Increment from 1 |
| <b>Zone</b>   | Coordinates          | Area coordinates of a zone  | GPS        | Z.Coordinates  | N         | N   | Not null | N           | None             |
| <b>Zone</b>   | Country              | Country a zone is located in  | Char (100) | Z.Country      | N         | N   | Not null | N           | None             |
| <b>Zone</b>   | Area                 | Area of a country a zone is located in  | Char (100) | Z.Area         | N         | N   | Not null | N           | None             |

| Entity Name   | Attributes    | Description   | Domain     | Aliases         | Composite                            | Derived | Nulls    | Key?        | Default Value                   |
|---------------|---------------|---|------------|-----------------|--------------------------------------|---------|----------|-------------|---------------------------------|
| <b>Zone</b>   | Climate       | General climate of the zone (Forest, plains, beach, mountain, urban)            | Char (30)  | Z.Climate       | N                                    | N       | Not null | N           | None                            |
| <b>Zone</b>   | Owner         | Full name of the client which may own a zone with a super platinum subscription | Char (100) | Z.Owner         | Client first name + Client last name | Client  | Y        | N           | SPACES (if not owned by client) |
| <b>Stream</b> | Stream ID     | The unique identifier of a stream   | Int        | ST.StreamID     | N                                    | N       | Not null | Primary Key | Increment from 1                |
| <b>Stream</b> | Date          | Date of a stream  | Date       | ST.Date         | N                                    | N       | Not null | N           | DD-MM-YYYY                      |
| <b>Stream</b> | Start Time    | Start time of the stream  | Time       | ST.Start        | N                                    | N       | Not null | N           | 00:00 (24 hour)                 |
| <b>Stream</b> | End Time      | End time of the stream  | Time       | ST.End          | N                                    | N       | Y        | N           | 00:00 (24 hour)                 |
| <b>Stream</b> | Participants  | Full names of each participant in the stream                                    | Text       | ST.Participants | Client first name + Client last name | Client  | Not null | N           | None                            |
| <b>Data</b>   | Data ID       | The unique identifier of a recorded stream                                      | Int        | SE.SubID        | Stream ID + Stream Date              | Stream  | Not null | Primary Key | Increment from 1                |
| <b>Data</b>   | Creation Date | Creation date of the data   | Date       | D.Creation      | N                                    | N       | Not null | N           | DD-MM-YYYY                      |

| Entity Name           | Attributes       | Description                            | Domain  | Aliases  | Composite | Derived                                      | Nulls | Key? | Default Value   |
|-----------------------|------------------|--|---------|----------|-----------|--|-------|------|---|
| <b>Data</b>           | Expiry Date      | Expiry date of the data                | Date    | D.Expiry | N         | Calculated as 1 month from the creation date | Y     | N    | One month from the creation date (DD-MM-YYYY)             |
| <b>Standard</b>       | Standard price   | Price of a standard subscription       | Decimal | S.Price  | N         |  | Y     | N    | Set price of a standard subscription (bar discount)       |
| <b>Gold</b>           | Gold Price       | Price of a gold subscription           | Decimal | G.Price  | N         |  | Y     | N    | Set price of a gold subscription (bar discount)           |
| <b>Platinum</b>       | Platinum Price   | Price of a platinum subscription       | Decimal | PL.Price | N         |  | Y     | N    | Set price of a platinum subscription (bar discount)       |
| <b>Super Platinum</b> | S Platinum Price | Price of a super platinum subscription | Decimal | SP.Price | N         |  | Y     | N    | Set price of a super platinum subscription (bar discount) |



| BUSINESS RULES         |  |
|------------------------|--|
| <b>Title</b>           | Client – SPACES relationships  |
| <b>Description</b>     | <p>Client relationships are established through a contract, not a subscription. A client must have a contract but does not have to have a subscription to a sensor. This way, if a client only needs access to a sensor on an ad-hoc basis then they will be able to maintain their registry with SPACES without having to pay a constant subscription fee.</p> <p>For example, an account owner of <i>Netflix</i> does not need to make a new account every time they opt out of subscribing for a period of time. Their access to the streaming content is merely withheld until a subscription payment can be made.</p>     |
| <b>Tables affected</b> | Client, Contract, Subscription   |
| <b>Fields affected</b> | N/A  |
|                        |  |
| <b>Title</b>           | Void contracts   |
| <b>Description</b>     | <p>Extending from business rule 1, a client may have their contract made void if they have not paid for a subscription within a year of the last payment. A client should be warned 3 months in advance prior to the contract becoming void, giving them 3 monthly opportunities to pay for a subscription. A client can contact a sales representative to extend the void with reason. If not, and the client fails to pay, the contract will be terminated.</p> <p>Keeping data on clients that have potentially abandoned their relationship with SPACES is unnecessary data that is taking up storage in the database.</p> |
| <b>Tables affected</b> | Client, Contract, Subscription   |
| <b>Fields affected</b> | SB.LastPay, SB.Status, CO.Status   |
|                        |  |
| <b>Title</b>           | Discounts  |
| <b>Description</b>     | A discount of 3% can be allotted to a customer that is using SPACES for education or work. This client must register the name of their business or educational institute to be validated by a sales representative for discounting. This promotes accessibility to larger groups of clients all subscribing for a shared purpose versus one-off clients who subscribe for personal interest.   |
| <b>Tables affected</b> | Client, Standard, Gold, Platinum, Super Platinum   |
| <b>Fields affected</b> | CL.Business, CL.Education, S.Discount, S.Price, G.Price, PL.Price,   |

|                        |  |
|------------------------|--|
|                        | SP.Price   |
|                        |  |
| <b>Title</b>           | Super Platinum Zones   |
| <b>Description</b>     | The purpose of the Super Platinum subscription level is so that the client may have a private zone with a private sensor in which they “own”. When a client opts for super platinum subscription, they must negotiate the procurement of a zone to place a private sensor in. This will be recorded in their contract. SPACES will need to provide a zone for them that fits their choice of climate and does not overlap existing zones to maintain privacy when used.  |
| <b>Tables affected</b> | Zone, Super Platinum, Contract   |
| <b>Fields affected</b> | Z.Owner, CO.Zone   |
|                        |  |
| <b>Title</b>           | Employee views   |
| <b>Description</b>     | <p>The three main employees of SPACES have separate view rights to information within the database. Each role may only view or manipulate data that is relevant to their job.</p> <p>A Sales Rep can only view information about clients, contracts, subscriptions, sensors, and zones. They can update all data on clients, contracts, and subscriptions but have no delete abilities.</p> <p>Maintenance workers are able to access information about supply orders, sensors, and zones. They can update the dates and schedules for maintenance for sensors, but they have no delete abilities.</p> <p>Executive Admin can access all information, update, and delete where needed.</p> |
| <b>Tables affected</b> | All tables   |
| <b>Fields affected</b> | All fields   |

## REFERENCES

- Agar, R. (2021, May 19). *Stages and types of data models*. TDAN.com. <https://tdan.com/stages-and-types-of-data-models/28201>
- Corbo, A. (2023, January 3). *What is data modeling?* Built In. <https://builtin.com/data-science/data-modeling>
- DataAcademy.in. (2020, May 1). *Conceptual, Logical & Physical Data Models* [Video]. YouTube. <https://www.youtube.com/watch?v=cY7WZYhyC3o>
- Dybka, P. (2014, August 2). *Chen notation*. Vertabelo. <https://vertabelo.com/blog/chen-erd-notation/>
- Nalimov, C. (2021, July 29). *Quick guide to physical data modeling*. Gleek.io. <https://www.gleek.io/blog/physical-data-modeling>
- Rithika, S. (2022, April 1). *Understanding the stages & types of data models: A comprehensive guide 101*. Learn | Hevo. <https://hevodata.com/learn/types-of-data-model/>
- Yue, K. (2017). *The extended entity relationship model*. SCE Support Center. [https://dcm.uhcl.edu/yue/courses/itec3335/Fall2018/notes/model/Extended\\_ERDiagram.html](https://dcm.uhcl.edu/yue/courses/itec3335/Fall2018/notes/model/Extended_ERDiagram.html)