Mapping Multi-valued attributes

Lecture 5 slide – page 8 – Properties of relation contains the following sentence “Each cell of relation contains exactly one atomic (single) value.”. Which means you can’t have multi valued attributes in logical diagram. In Figure 1, **Phone is a multivalued attribute**.

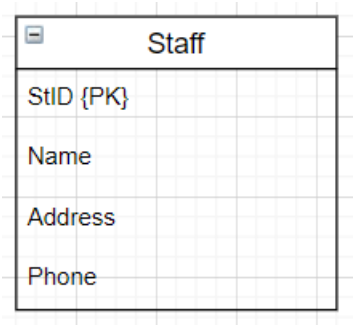

Figure 1

Figure 2 shows how Multi-valued attributes are mapped.

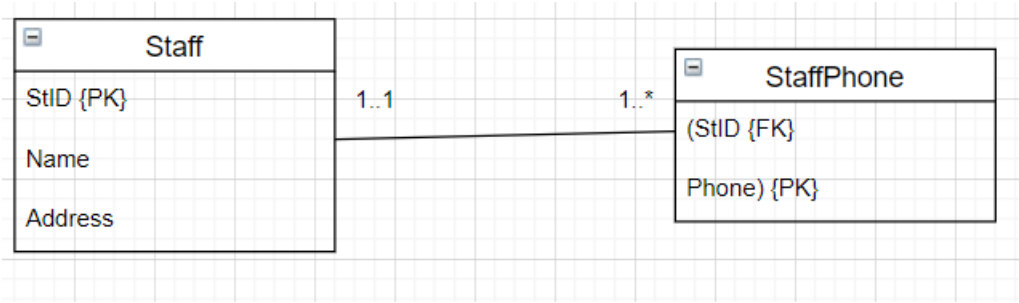

Figure 2

Mapping Generalization with one Entity

Figure 3 shows Generalization with only one entity

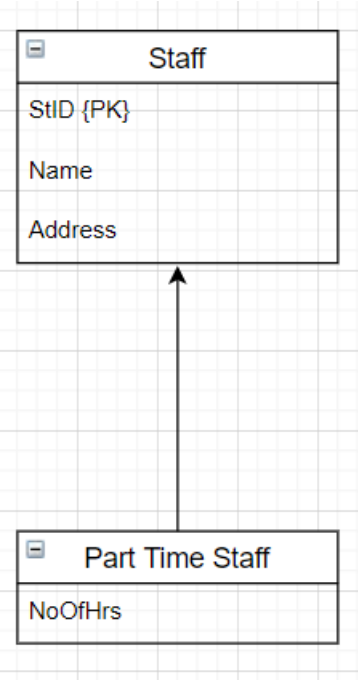

Figure 3

Figure 4 show how Generalization with only one entity is mapped

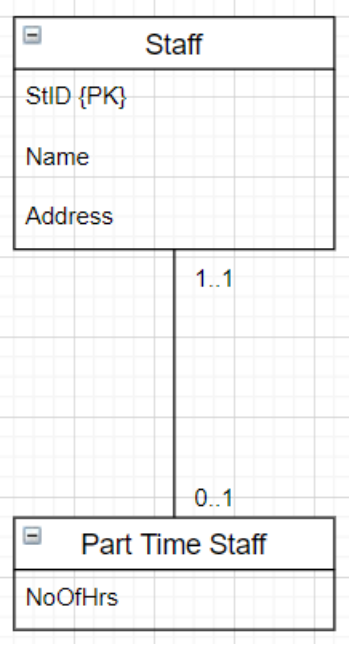

Figure 4

Mapping Recursive (Unary) relationship (1:M)

Figure 5 shows a 1:M recursive relationship.

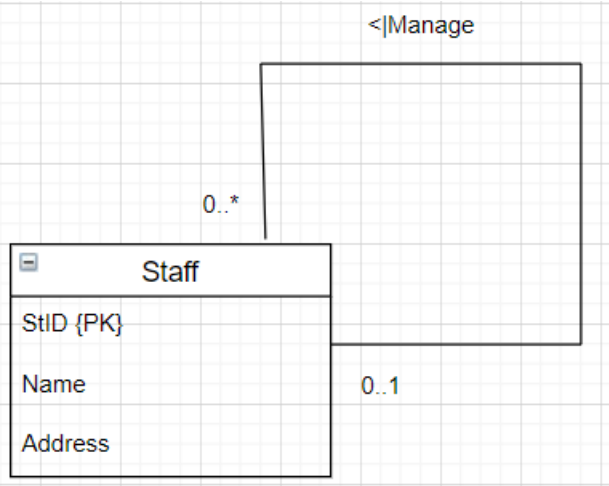

Figure 5

Figure 6 shows how 1:M recursive relationship is mapped. For 1:M relationship, the foreign key should be placed on the 'Many' side which is also the Staff entity. Therefore, foreign key is placed in the **same entity.** But **Two columns can't have the same name**. So it **must be renamed**.

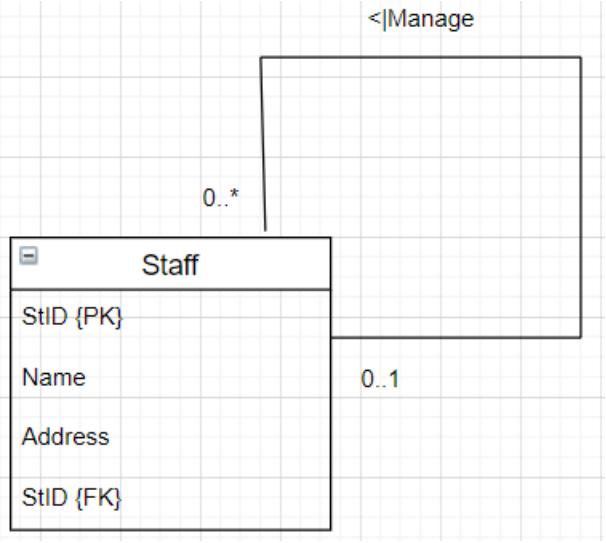

Figure 6

See next Page

Figure 7 shows the renamed attribute and It shows how 1:M recursive relationship is mapped.

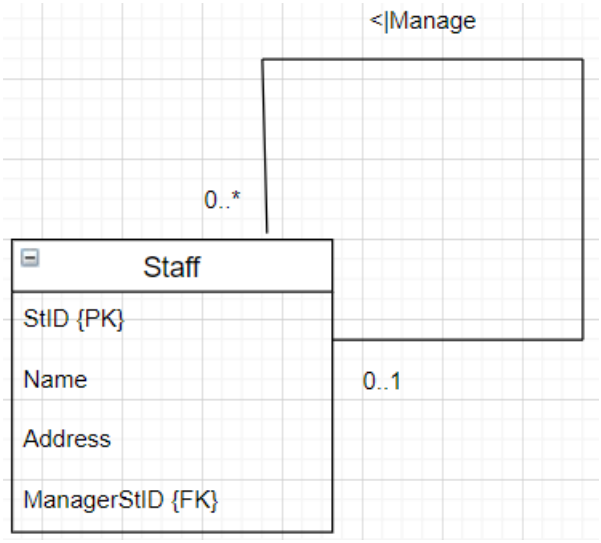

Figure 7

Mapping Recursive (Unary) relationship (M:M)

Figure 8 shows M:M recursive relationship.

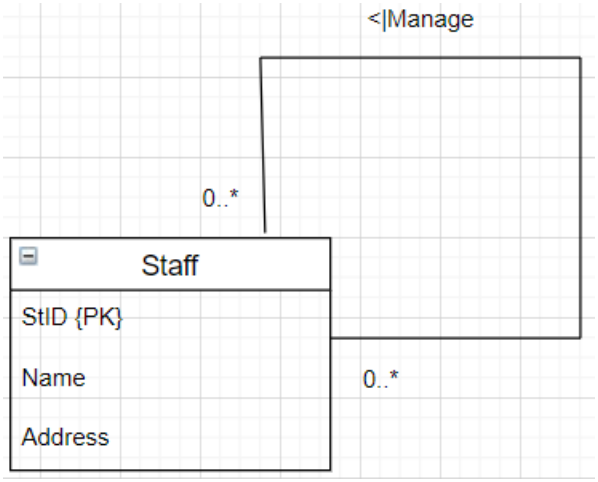

Figure 8

See next Page

Figure 9 shows how M:M recursive relationship is mapped.

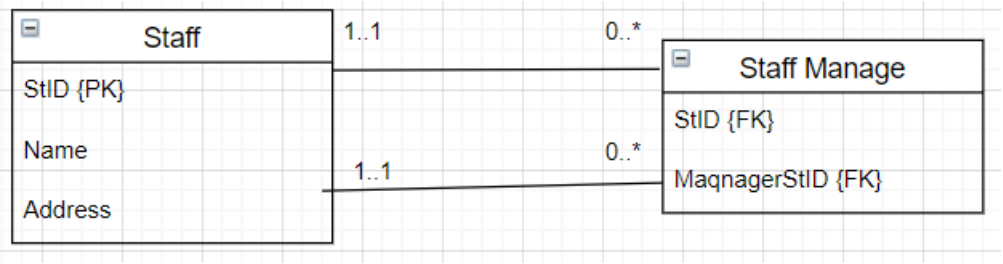

Figure 9

Not null for 1:M relationship

In Figure 10, participation constraint is 0. That means a laptop can exist without a staff. Therefore, **the Foreign key can be null**.

Figure 10

```
CREATE TABLE Laptop
(
        SerialNo INT,
        Description VARCHAR (50),
        StID INT,

        PRIMARY KEY(SerialNo),

        FOREIGN KEY (StID) REFERENCES Staff(StID)
);
```

In figure 11, participation constraint is 1. That means a laptop **must belong to a** staff. Therefore, the **foreign key can NOT be NULL**.

Figure 11

```
CREATE TABLE Laptop
(
        SerialNo INT,
        Description VARCHAR (50),
        StID INT NOT NULL,

        PRIMARY KEY(SerialNo),

        FOREIGN KEY (StID) REFERENCES Staff(StID)
);
```

UNIQUE for 1:1 Relationship

Figure 12 shows a 1:1 relationship which means a Laptop belongs to **only one** Staff. To enforce this logic, the **foreign key must be UNIQUE**.

Figure 12

```
CREATE TABLE Laptop
(
        SerialNo INT,
        Description VARCHAR (50),
        StID INT UNIQUE,

        PRIMARY KEY(SerialNo),

        FOREIGN KEY (StID) REFERENCES Staff(StID)
);
```