

The Oracle Academy logo is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom.

ORACLE

Academy

Database Design

8-1

Modeling Historical Data

ORACLE
Academy



Copyright © 2020, Oracle and/or its affiliates. All rights reserved.

Objectives

- This lesson covers the following objectives:
 - Identify the need to track data that changes over time
 - Construct ERD models that incorporate elements of “data over time”
 - Identify the UID of an entity that stores historical data; explain and justify the choice of UID

In this lesson, we will discuss the factors that influence the decision to model historical data, create a new entity to track it, and define a UID for such an entity.

Purpose

- How tall were you at age 5? How tall were you at age 10? How tall are you right now?
- If your parents wrote this down when you were young, they were keeping track of historical data
- Most businesses need to track some historical data
- This helps them find trends and patterns that are the basis for business innovations or process improvements
- For example, rental history of a movie is useful to a video store. It tells managers which movies are popular and which should be moved to the back shelf

Model Data Over Time

- When is it necessary to model data over time?
- Ask your client:
 - Is an audit trail required?
 - Can attribute values change over time?
 - Can relationships change over time?
 - Do you need to produce reports on older data?
 - Do you need to keep previous versions of the data? If so, for how long?

Historical data: Data that records changes over time.

Audit trail: A step-by-step record of data changes.

Validate requirements for storing historical data with the user. Storing unnecessary historical data can be costly. It takes up space – whether it's in a filing cabinet or in the database. It takes up employee resources – someone has to be responsible for updating the historical records with the latest data, for making sure that there is enough space as the data grows, and for allocating more space or moving data around if necessary.

Data Over Time Example

- An organization needs to keep data about employees' salaries
- All employees are paid weekly
- Initially, the following EMPLOYEE entity was modeled
- Additional requirements now specify that the organization needs to keep a historical record of how and when employees' salaries have changed during their employment

EMPLOYEE

Id

* First name

* Last name

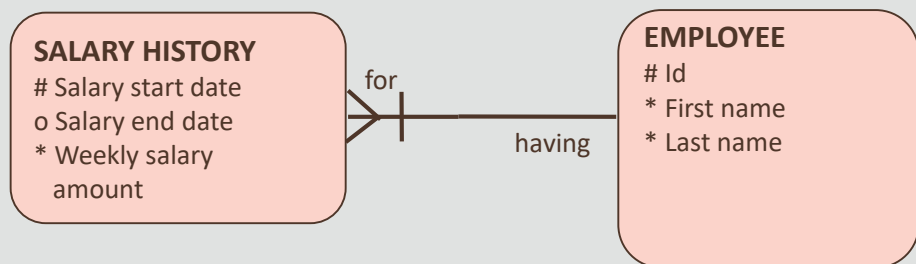
* Weekly salary amount

* Salary start date

The model shown, will not work based on the new requirements, as only one value for salary can be stored. If the salary changed, there is no way of recording the values before they were updated. If we stored all an employees previous salary amounts and start dates , this would be a repeating group of multivalued attributes, thus violating First Normal Form.

Model Salary Changes

- To model salary changes over time, add a SALARY HISTORY entity
- The UID of the SALARY HISTORY entity is the related EMPLOYEE id and the salary start date

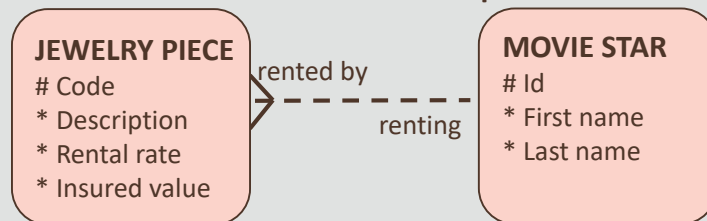


The salary end date is optional because if it is the current salary, there is no end date (yet).
The current salary is the one whose end date is null.

There would also need to be a programmatic constraint to ensure that an employee has exactly one salary at a time: the start date of a new salary cannot be earlier than the end date of the previous salary.

Model Rental Over Time

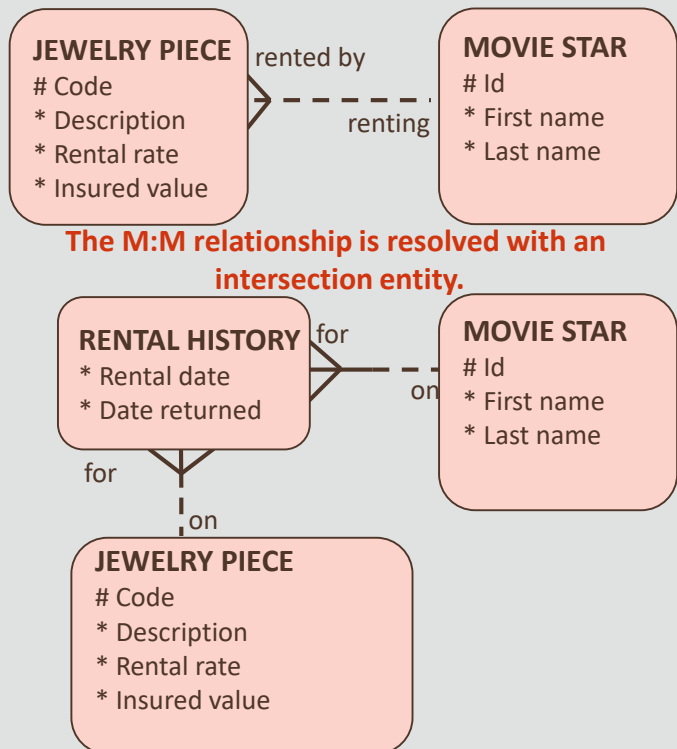
- A jewelry store rents pieces (necklaces, bracelets and so on) to movie stars for special occasions, such as award ceremonies or movie premieres
- They would like to track the rental history of a jewelry piece
- The following ER model will only track the current renter of a piece of jewelry
- How would you revise the relationship to track history?



To track the rental history, we need a M:M relationship between JEWELRY PIECE and MOVIE STAR.

Resolve M:M

- The relationship between JEWELRY PIECE and MOVIE STAR should be revised to a M:M, which is then resolved with an intersection entity RENTAL HISTORY
- Next we need to determine the UID of RENTAL HISTORY

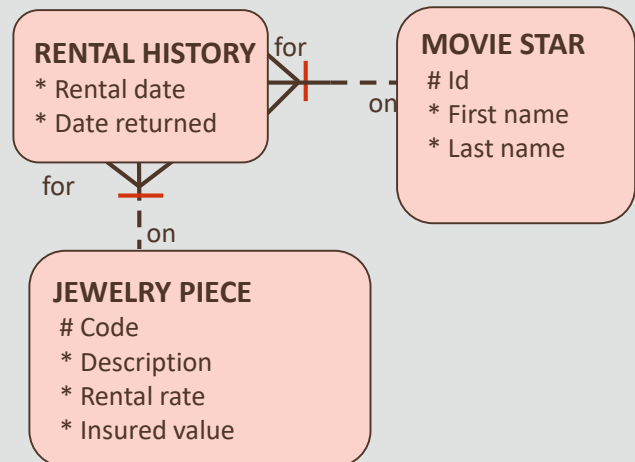


We also add the attributes rental date and date returned to our RENTAL HISTORY entity.

Determine UID

- Option 1: Barred relationship

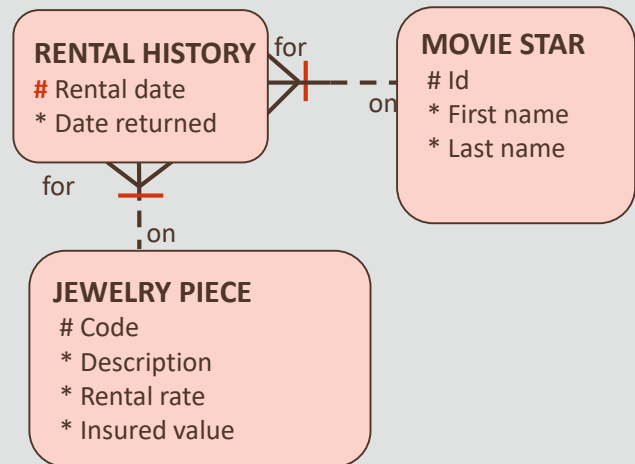
- Drawing a Barred relationship is not a suitable UID here, as this would not allow a MOVIE STAR to rent the same JEWELRY PIECE on different dates



If Scarlet Johansson rented the Princess necklace on 14-Jun, she would not be able to rent it on another date, as this UID would NOT allow the same combination of MOVIE STAR id and JEWELRY PIECE code to be entered more than once.

Determine UID

- Option 2: Barred relationship and Rental Date
 - Adding rental date to the UID would allow a MOVIE STAR to rent the same JEWELRY PIECE on different dates, but would also permit different MOVIE STARS to rent the same JEWELRY PIECE on the same date!

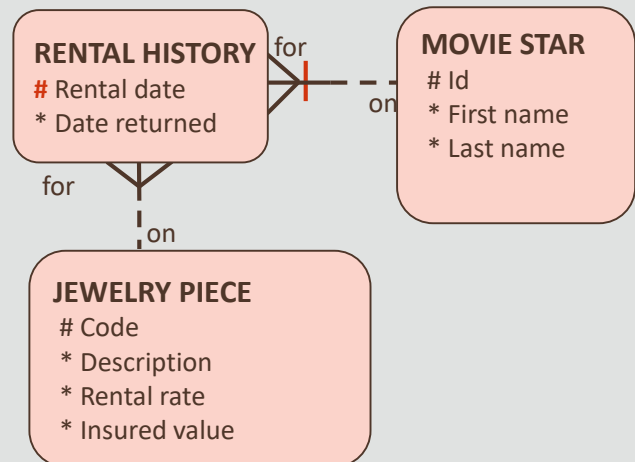


IF Scarlet Johansson rented the Princess necklace on 12-May, this UID would also allow Cameron Diaz to rent the Princess necklace on 12-May.

The combination of MOVIE STAR id, JEWELRY PIECE code, and date rented are unique for each instance, so the model would allow this. However, we know that, in reality, this cannot happen.

Determine UID

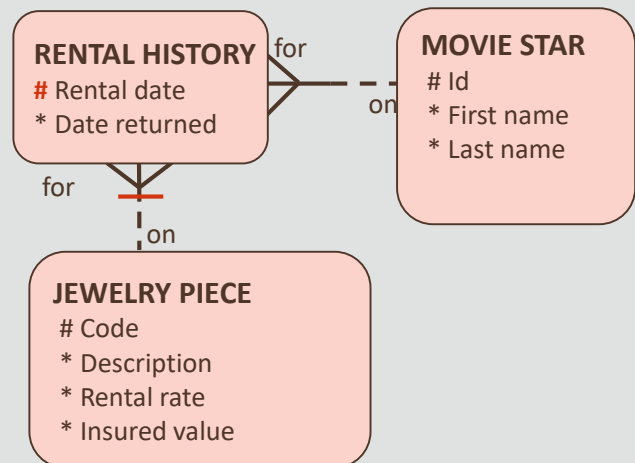
- Option 3: Barred relationship between MOVIE STAR and RENTAL HISTORY with Rental Date
 - This model would not permit the same MOVIE STAR to rent more than one JEWELRY PIECE on a given day



If Angelina Jolie wanted to rent both the Princess necklace and the Maharajah bracelet on 20-Jul, she could not, because MOVIE STAR id plus date rented must be unique.

Determine UID

- Option 4: Barred relationship between JEWELRY PIECE and RENTAL HISTORY with Rental Date
 - This model says that a JEWELRY PIECE can be rented only once on the same date



Cameron Diaz wants to rent the Princess necklace on 05-May. Angelina Jolie can rent it on 12-May. The combination of JEWELRY PIECE code and date rented is unique.

Terminology

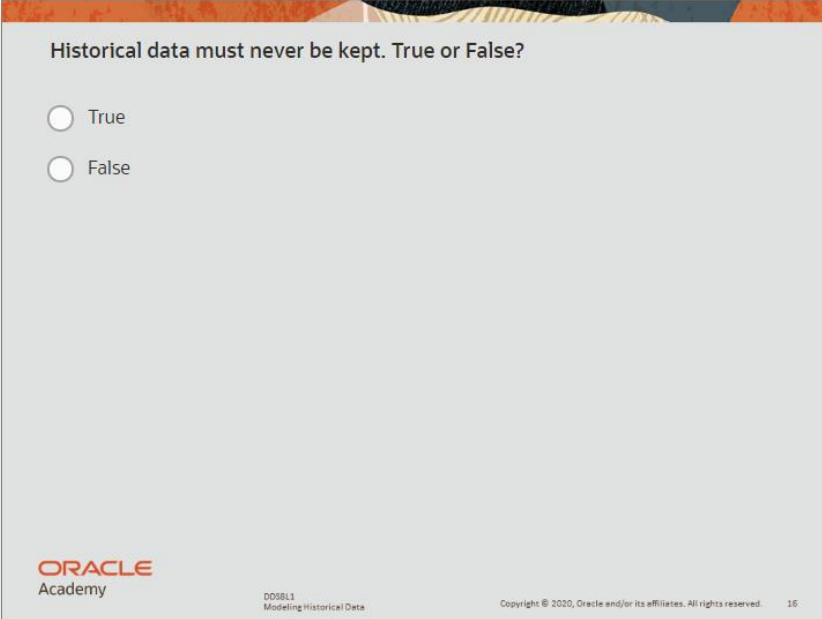
- Key terms used in this lesson included:
 - Audit trail
 - Historical data

Summary

- In this lesson, you should have learned how to:
 - Identify the need to track data that changes over time
 - Construct ERD models that incorporate elements of “data over time”
 - Identify the UID of an entity that stores historical data; explain and justify the choice of UID

Quiz

Click the **Quiz** button to edit this object

A screenshot of a quiz interface. At the top, there is a decorative header with a wavy pattern in orange, yellow, and black. Below the header, the question "Historical data must never be kept. True or False?" is displayed in a dark font. Two radio button options are listed: "True" and "False". The "True" option is selected, indicated by a small orange dot inside the radio button. At the bottom of the quiz area, there is a footer containing the Oracle Academy logo on the left, the text "D098L1 Modeling Historical Data" in the center, and "Copyright © 2020, Oracle and/or its affiliates. All rights reserved. 16" on the right.

Historical data must never be kept. True or False?

☒ True

☐ False

ORACLE
Academy

D098L1
Modeling Historical Data

Copyright © 2020, Oracle and/or its affiliates. All rights reserved. 16

The logo for Oracle Academy. The word "ORACLE" is in a bold, orange, sans-serif font. Below it, the word "Academy" is in a smaller, dark gray, sans-serif font. The entire logo is centered on a light gray background, which is framed by dark gray horizontal bars at the top and bottom.

ORACLE

Academy