

Modeling Historical Data

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
- Construct a conceptual model based on a given scenario
- Apply the rules of entity-relationship diagramming to create an ERD that reflects the business rules
- Present and interpret the data model to an audience
- Compose written documentation to accompany the oral presentation and an ERD

Try It / Solve It

1. In the scenario below what words or phrases indicate to you the need to track data that changes over time.

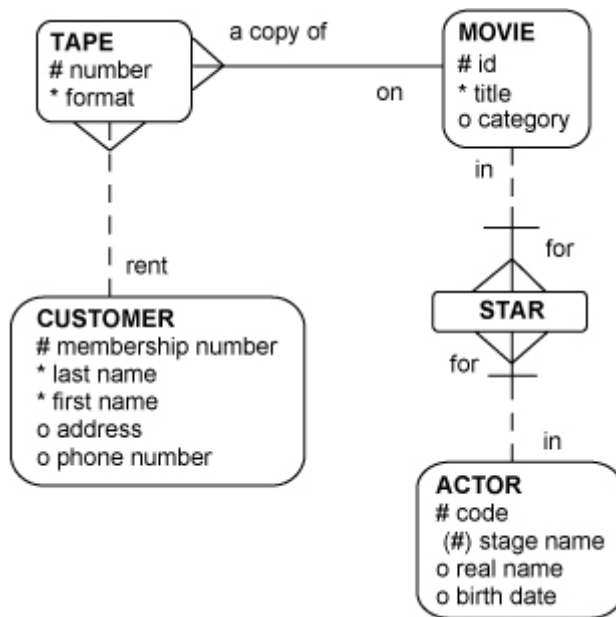
Actors at the local theater sign contracts that specify their pay and working conditions. These contracts are renewed annually. Each year the contracts may change through new language and pay rates. In order to resolve labor disputes, the theater must sometimes go back several years in order to see the contract provisions in place on a particular date.

2. Given the scenario below, construct a conceptual model.

The Seville, Spain soccer association is renovating their soccer arena. They are adding luxury boxes that will be offered to companies. These boxes contain varying numbers of seats. Each box is assigned a section number and level. The boxes are leased by the year, and the prices are set individually according to the location and number of seats in the box. The soccer association needs to keep track of the company name, address, phone, and contact person. Sometimes the companies also provide a logo. Companies that lease a box have the right to renew their leases each year, but they often do not. When a lease for a box opens up, the association contacts companies that previously held leases to see if they want to lease one again.

3. Modify the Video Store ER model below to accommodate the following additional requirements:

“You know, we really need to keep a history of all our rentals. Each time a customer rents a tape, we would like to keep the rental date/time and the return date/time. All our tapes are due back the next day, so we don’t need to keep a due date. Keeping this rental history will allow us to analyze the pattern of our rentals. We will be able to determine how many tapes each customer rents and how many times a customer has returned a tape late. We will also know how many times a particular tape has been used and will then know when to retire each tape. We will also be able to analyze our customers’ movie preferences.”



4. Read through the IOUG Case Study listed below. As a group develop an ER diagram for the situation. Be sure to tag each attribute with its optionality. Each group will be given 10 minutes to present its solution. You will be required to prepare each of the following:

- Statement of the problem: Why has the IOUG approached you? What help do they need?
- Statement of the proposed solution: How can you help them solve their problem? What are your recommendations?
- The information requirements of the business/organization clearly stated: What data do they need to track and what information do they need to get out of the data?
- The business rules as they apply to the information requirements of the company/organization stated as single sentences: What business rules can you formulate from the given scenario?
- The assumptions clearly stated that had to be considered in developing the solution: What assumptions did you have to make in developing your solution?
- Entity relationship model

IOUG CASE STUDY

Our regional Oracle User's Group has grown to include over 200 members. We're an all-volunteer organization of working adults and secondary-school students, and our records are a mess. We need an information system to help us keep track of our business. We definitely need to automate our membership records. For each member, we need to keep the member's name, title, mailing address, office or home phone number, type of membership (individual or corporate), and whether or not the member has paid the dues (is the

member “current”?). We collect dues on a yearly basis, and they are due in January.

We also like to know which company a member works for or which school they attend, but keeping this information current is difficult because our members are always changing companies or addresses or graduating from school. We try to track only a single current employer for each member. Our members come from many different companies, including Sony, EG&G, and Storage Tech, as well as several different local high schools. A few of our members are unemployed. For each company, we keep the company name, address, and type of business. For each secondary school, we keep the school name and address. We have a standard set of types of businesses and high-school codes. We keep only the main company address for each company.

We hold various events during the year, and we’d like to track information about each event. Some of our annual events include the September Meeting, the November Meeting, the annual Training Day in January, and our April Meeting. We also hold special events each year.

For example, we held a special CASE (Computer Aided Systems Engineering) day last May, and Richard Barker from Oracle U.K. came and spoke. We hold our events at several different locations around town, including AT&T, Redrocks Community College, and Skyline University. We’d like to track each event’s date, optional description of the event, number of attendees, where it was held, how much money we spent on it, and any comments on the event. We treat all comments as if they came from an anonymous submitter. A set of comments is a free-form text statement of any length. We number each set of comments, and we frequently get multiple sets of comments for an event.

We also track which members attended which events. Some of our members are really active, and others attend very infrequently or just enjoy receiving our newsletter.

We also need to track what type of computer platforms our members are using. We have a unique, three-digit system identification tag for each type of platform. For example, 001 is for IBM/MVS; 002 is for IBM/VM; 003 is for VAX/VMS; 020 is for OS/2; 030 is for PC/DOS; 050 is for Sun UNIX; and 080 is for other UNIX platforms.

We also like to track which application areas each member is interested in. Sample application areas include accounting, human resources, entertainment, oil and gas, pharmaceuticals, and health systems. The applications should be portable, so we don’t need to know which platforms they run on.

5. (Optional Practice: Global Fast Foods)

This practice is step 9 in the Global Fast Foods project.