

AC 2008-929: TEMPORAL EXTENSIONS FOR ENHANCED ENTITY RELATIONSHIP NOTATION

Curtis Welborn, Utah Valley State College

Reza Sanati-Mehrizy, Utah Valley State College

Temporal Extensions for Enhanced Entity Relationship Notation

Abstract

An organization can have many business rules to implement in their daily operations. When these rules deal with the planning of business operations, there can be a strong need to specify the temporal relationships between business objects. Software engineers are seldom educated as to the use of temporal logic though it is often needed to accurately explain time-based relationships. Temporal logic[1],[2] defines a basic set of primitive relationships that can exist between intervals in time. These same primitive relationships can be used to express temporal relationships between business objects. The Enhanced Entity Relationship (EER) notation allows business rules to be shown in a graphic form using action assertions which keep the business rule at a conceptual level without specifying how the rule will be implemented. In this paper we will show how the EER notation can be augmented to allow a software engineer to specify temporal-based business relationships in a relational data model.

Introduction

There are three common types of business rules[6]: structural assertions, action assertions and derivations. A structural assertion is concerned with statements that express an aspect or relationship about the structure of a business. A derivation is concerned with statements that can be used to derive additional information about the business, whereas an action assertion is a statement that controls or limits the actions of the business. Action assertions are important as they define constraints that a business[5] should or must operate under. A business often has many operating constraints that will be implemented in various users' application programs. Capturing and documenting business rules in an application program can lead to consistency and manageability issues that ultimately leave the database in an inconsistent state[3].

A more modern and more reliable approach is to define the business constraints (action assertions) at a conceptual level without specifying how the rules will be implemented. The Enhanced Entity Relationship (EER) notation has been used to specify business rules. The EER notation was invented to allow more business rules to be shown in graphical form than the simpler ER notation[4]. Associating business rules with the data that it applies to can be very natural, as the rules are all about the data[8][10].

While business constraints come in many different forms, an important type of constraint on a business is a timing-based or temporal constraint. As an introduction to temporal constraints, let us review a problem that has been expressed in EER notation and explore how it could potentially be augmented with temporal constraints. We will skip any formal definition at this time and simply try to present an intuitive introduction to temporal constraints.

Temporal Constraint Problem

EER notation can be used to express the following business constraints about the maintenance of aircraft[3]:

1. “a mechanic can only provide a maintenance service if he has received all the training required for that service”,
2. “a maintenance service must be done using required tools”,
3. “a maintenance service is only provided in a hanger”,
4. “a service can only be provided by a hanger if there are two mechanics that can provide the same service”.

Let us consider a few ways that the problem could be modified by looking at how time could affect the constraints. In order to introduce a number of different temporal concepts to this single example, we may stretch a constraint in such ways that you question how valid the constraint would actually be in the real world. Please understand that our intention is to introduce temporal concepts using an understandable example, not define actual Federal Aviation Administration (FAA) rules.

Temporal Constraint c1

Let us consider constraint one first. Often training is not a one-time event but something that an individual must renew regularly. If we consider that some training is only valid for a specific duration of time, we can rewrite the constraint as “a mechanic can only provide a maintenance service during the time when his/her training is certified”. We will refer to this new temporal constraint as c1.

Temporal Constraint c2

Assume that tools can include materials that may expire over time. Then constraint 2 can be modified to indicate that a required tool can only be used to perform a service if it is used before its expiration date. The new constraint c2 will be rewritten as “a maintenance service must be completed using required tools during the usable life of the tool”. This constraint could also be expressed as “a maintenance service must be completed using required tools before the tools expire”. While both forms of the constraint are valid, it is a little more intuitive to maintain the interval in which a tool is usable than to maintain all the intervals in which a tool is not usable.

Temporal Logic

Hopefully the temporal constraints we have placed on the original aircraft problem have been enough to show that business rules should often be augmented with temporal constraints. At this time let us take a brief but more formal look at temporal logic[1][2]. When an action occurs, the action can be viewed as either being instantaneous or occurring over some period of time. An instantaneous action is viewed as happening at a single point in time, such as “John arrived at 2:32p.m.” This action can be referred to as a point-based temporal action. An action that occurs over time such as “John drove from home to work from 7:03a.m.to 7:22a.m.”, is known as an interval-based temporal action.

There are seven basic relationships that can be defined for temporal intervals. Of these seven relationships, six have an inverse relationship. The thirteen possible primitive temporal relationships are shown in FIGURE 1. It is important to note that the relationships are mutually exclusive.

<i>Relation in English</i>	<i>Inverse Relation</i>	<i>Visual Example</i>
X equals Y	none	XXXXXXXXXX YYYYYYYYYY
X before Y	Y after X	XXXXX YYYYY
X meets Y	Y Met by X	XXXXXYYYYY
X overlaps Y	Y overlapped by X	XXXXXXXXXX YYYYYYYYYY
X during Y	Y contains X	XXXXX YYYYYYYYYY
X starts Y	Y started by X	XXXXX YYYYYYYYYY
X finishes Y	Y finished by X	XXXXX YYYYYYYYYY

FIGURE 1. Thirteen Possible Primitive Temporal Relationships

- X equals Y; intervals X and Y begin at the same time and end at the same time, they are the same interval.
- X before Y; interval X ends before interval Y begins, and there is always an interval between when X ends and Y begins.
- X meets Y; interval X ends before interval Y begins, and there is not an interval between when X ends and Y begins.
- X overlaps Y; X starts before Y starts and continues until after Y has started but ends before Y is finished.
- X during Y; X starts after Y has started and ends before Y is finished.
- X starts Y; X and Y start at the same time but X will finish before Y ends.
- X finished Y; X begins after Y has started but both X and Y end at the same time.

Capturing Business Rules

Using the primitive temporal relationships, we will now define an EER diagram to capture our problem with temporal constraints. FIGURE 2 is the original EER diagram [3] used to define constraint 1.

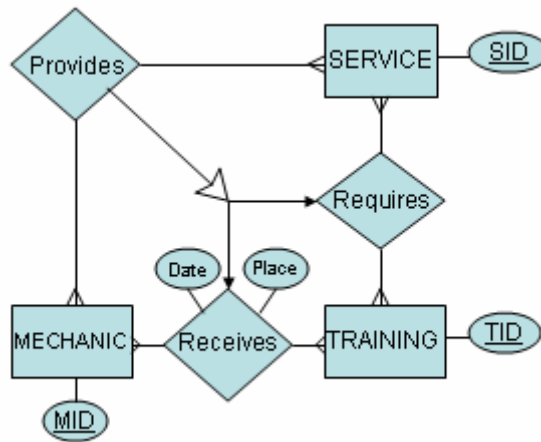


FIGURE 2. Original EER for Constraint 1

FIGURE 3 shows how the original diagram can be modified to capture the new temporal constraint c1. What was missing from FIGURE 2 is the constraint that a mechanic can only provide a service during the interval when he/she is certified. To accommodate the additional temporal constraint the relationship *Receives* has been renamed *Is Certified* and the attribute *Interval* was added to indicate the interval of time in which a mechanic is certified to provide a specific service.

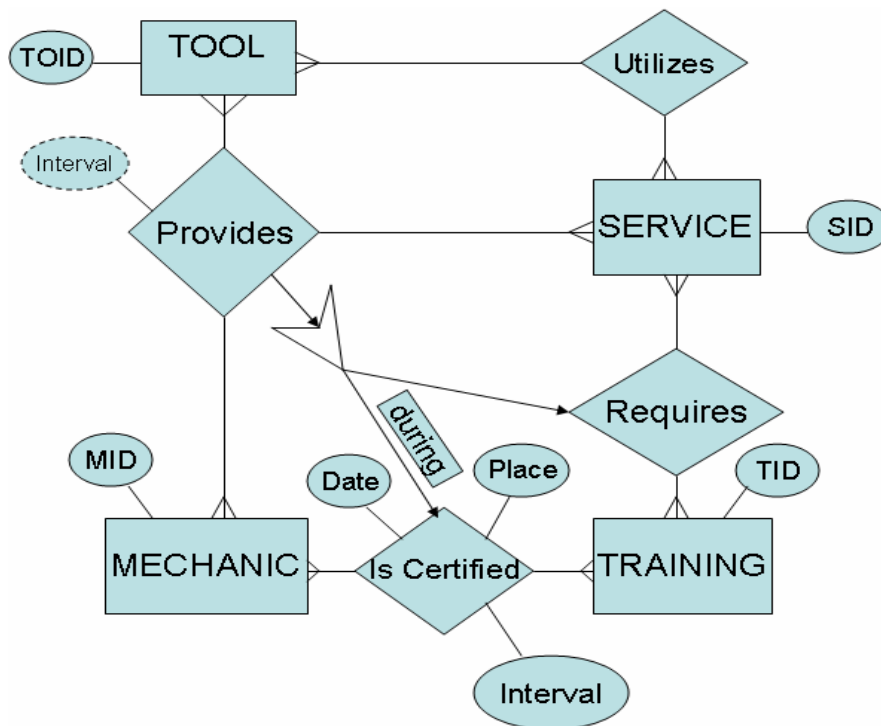


FIGURE 3. New EER for Constraint c1.

Note that the arrow from the action assertion on the relationship *Provides* to the relationship *Is_Certified* is labeled “during” and has not explicitly specified any interval

as is shown in the FIGURE 1, “X during Y”. The interval of the anchor object (the derived attribute of Provides in this example) for the action assertion will always be the first (i.e., X) interval specified for the “during” constraint whereas the interval of the corresponding object (the attribute of Is Certified in this example) will always be the second (i.e., Y) interval for the “during” constraint. In another word, the X interval in this instance is the derived attribute of the Provides relation and the Y interval is the interval attribute of Is_Certified, which is explicitly stated. The derived interval will be derived by checking the intervals when mechanics are certified. Thus our diagram has established a constraint on how long a Provides relationship can exist or, at most, how long it can be used in our database based upon business rules. The action assertion can now be classified [4] as follows:

- The type of result is: Condition
- The type of form is: Timer
- The type of rigor is: Controlling

Next let us look at how temporal constraint c2 can be captured in an EER diagram. FIGURE 4 is the original EER diagram [3] used to define constraint 1 and 2. What is missing from FIGURE 4 is the temporal constraint that only non-expired tools can be utilized to provide a service.

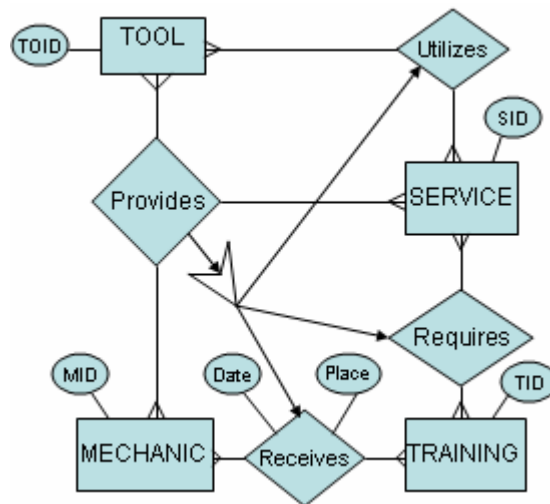


FIGURE 4. Original EER for Constraint 2.

FIGURE 5 is used to show how the diagram is modified to capture the missing temporal constraint c2. Note that the Utilizes relationship has been renamed Is_Usable and the attribute Interval was added to indicate the interval of time during which the tool can be used to provide a service. A tool that does not have an expiration date would have an interval value with an end date of infinity. The arc between the action assertion of the Provides relationship and Is_Usable relationship is labeled “during”. As in FIGURE 3 the first interval is the derived interval of Provides relationship, the anchor object. The second interval is the Interval attribute of Is_Usable, the corresponding object.

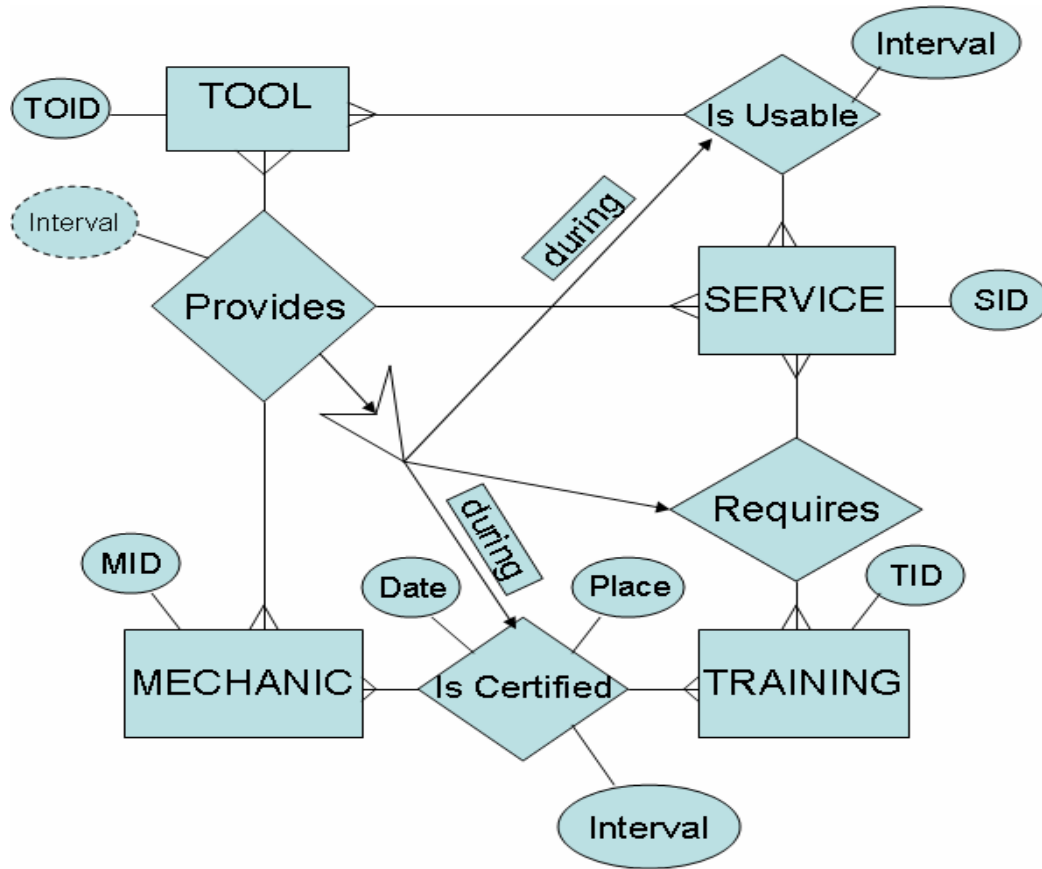


FIGURE 5. New EER Constraint c2.

The figure 5 is interpreted as the mechanics provide many services using many tools that are not expired and the mechanics are still certified for the required trainings to provide those kinds of services.

Curriculum Enhancement

At our institution, the Computer Science department offers an area of specialization within the Computer Science program. This area of specialization offers courses such as Database Theory, Database Construction, Advanced Topics in Database and Enterprise Architecture. Our students in these classes are exposed to these enhanced EER diagrams and are encouraged to improve these diagrams in their team projects. This will enhance our database curriculum and improves our students' education considerably.

Future Work

While we have been able to express some business rules that contain a temporal constraint using a simple extension to EER notation, we have found other business rules that are not as easy to express using our extension to EER notation. As an example, we will present temporal constraint c4. The original 4th constraint is "a service can only be provided by a hanger if there are two mechanics that can provide the same service".

Figure 6 shows the original EER diagram. And the new temporal constraint c4 will be “a service can only be provided by a hanger if there are two mechanics that can provide the same service who work the same shifts”. The reasons that c4 is difficult to capture in EER notation are as follows:

- First we must capture that pairs of mechanics must work the same shifts.
- Next we must capture that mechanics working the same shift must be certified to perform the same service.
- Lastly we must capture that a service can only be provided during the interval in which two certified mechanics are working the same shift.

Extending EER notation to support the definition of a derived table (in a similar fashion to a derived attributes) helps our problem but does not solve our problem. As such, we will continue to investigate solutions for representing this business rule in EER notation.

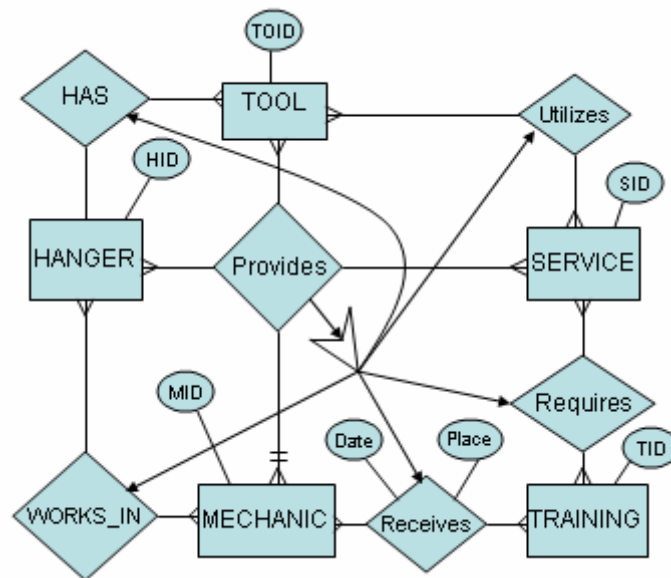


FIGURE 6. Original EER for Constraint 4

Conclusion

Temporal constraints are often integral to the correct operation of a business, yet they are seldom captured in a high-level notation such as EER. This means that the fundamental interaction of temporal constraints on the business are often only captured in detailed requirements or, even worse, only in code.

James Allen [2] defines a non-primitive relationship “in” which allows one interval be wholly within another interval. The “in” relationship would be an even more appropriate relationship than the “during” relationship to be used for our examples. The “during” relationship was used because it is a primitive relationship and the introduction of non-primitive relationships would add unnecessary complexity to this paper.

References

- [1] James F. Allen, "Maintaining Knowledge about Temporal Intervals", Communications of the ACM, pp. 832-843, 1983.
- [2] James F. Allen, "Towards a General Theory of Action and Time", Artificial Intelligence 23 pp. 123-154, 1984.
- [3] Reza Sanati Mehrizy, Curtis Welborn, Afsaneh Minaie, "Representing and Enforcing Business Rules in Relational Data Model", American Society for Engineering Education (ASEE) 2006.
- [4] J. A. Hoffer, M. B. Prescott and F. R. McFadden, "Modern Database Management", Seventh Edition, Prentice Hall, 2005.
- [5] A. Perkins, "Business Rules = Meta Data", The proceedings of the: Technology of Object-Oriented Languages and Systems, IEEE, 2000.
- [6] J. Widom and S. Ceri, "Active Database Systems", Morgan Kaufmann, 1996.
- [7] E. Baralis, S. Ceri, and S. Paraboschi, "Modularization techniques for active rules design", ACM Transactions on Database Systems, 21(1):1-29, 1996.
- [8] G. Ronald Ross, "Business Rule Concepts", Business Rule Solutions Inc., 1998.
- [9] The Business Rules Group, "Defining Business Rules – What Are They Really?", <http://www.BusinessRulesGroup.org>, Feb. 2006.
- [10] B. von Halle, "Building a Business Rule System, Part 1", Data Management Review, Faulkner & Gray, January 2001.