Modeling Business Applications with the OODB Ownership Relationship

Michael Halper
Dept. of Math & Computer Science
Kean College of New Jersey
Union, NJ 07083 USA

Yehoshua Perl, Oscar (Ou) Yang, James Geller
CIS Department and CMS
New Jersey Institute of Technology
Newark, NJ 07102 USA

Abstract

Ownership is a very important relationship in the business world. It is endowed with rich semantics and various complexities with respect to both the owner and the property that is owned. In this paper, we present a formal model of "ownership" relationships in the context of an Object-Oriented Database (OODB) system. As our motivation, we employ three scenarios involving various ownership relationships that exhibit a wide range of distinctions. Essential aspects of ownership relationships are their related transactions such as sale, lease, and donation. Since certain of these can be applied with respect to specific kinds of ownership, while others cannot, we need to explicitly model this behavior in order to properly represent ownership in an OODB system. The ownership relationship, at times, exhibits inheritance behavior, where the values of certain attributes are derived with respect to it. With these issues in mind, we have identified and formally defined various characteristics (which we call the dimensions) of ownership. Our ownership model incorporates all these to capture the functionality of ownership's transactions and inheritance.

1 Introduction

Ownership is a very important relationship in the business world. It is endowed with rich semantics with respect to the owner and the property that is owned. As used in the corporate world, ownership can exhibit a hierarchical structure. For example, one company can own other companies.

Because of its complexity, modeling ownership in the context of a database system can be an extremely difficult task. In this paper, we introduce an "ownership" relationship model that can be integrated into an Object-Oriented Database (OODB) system. The use of this relationship greatly facilitates the problem of modeling real-world ownership and of enforcing its associated constraints.

To motivate the need for an ownership relationship and to see what kinds of problems one might encounter when trying to model it, we shall employ three example scenarios, pictured in Fig. 1, 2, and 3. Corresponding database schemata are included in the Figures, which have been drawn using our OOdini graphical notation [9]. Ownership is denoted by a bold, dotted arrow. This symbol was chosen for the mnemonic association between the dots and the "o" in "ownership."

Let us now describe these scenarios. In the first one, Jim and his business partner David own a manufacturing business that produces an item for which Jim holds a patent. The business resides in a building which Jim owns and for which a bank, First Nat'l Trust, holds a lien. Jim and David have a joint business bank account. Jim rents his house from Tom. Jim also uses a car that is legally owned by his business. A car owned by Jim is used by his son John. Jim and David each have individual bank accounts and investment portfolios, consisting of corporate stocks and government bonds. In addition, each possesses a life insurance policy and the appliances in their homes.

In the second scenario, Chrysler owns Jeep, Plymouth, and Dodge, each of which in turn owns subsidiaries, manufacturing plants, industrial equipment, etc. Dodge and Mitsubishi jointly own the Eagle Corporation. Chrysler, being a public company, issues stock that is owned by shareholders who are persons or other corporations.

The third scenario deals with an individual investor. Here, Jack owns several portfolios of investments, including stocks, bonds, and mutual funds. Together, these portfolios represent all of Jack's investments.

Against these scenarios, one is liable to carry out any of the various transactions associated with ownership, such as sales, leases, or donations. Such transactions are often restricted due to complex constraints. For example, under some circumstances, the sale of an object might be disallowed. Ownership also exhibits "inheritance" behavior, where val-

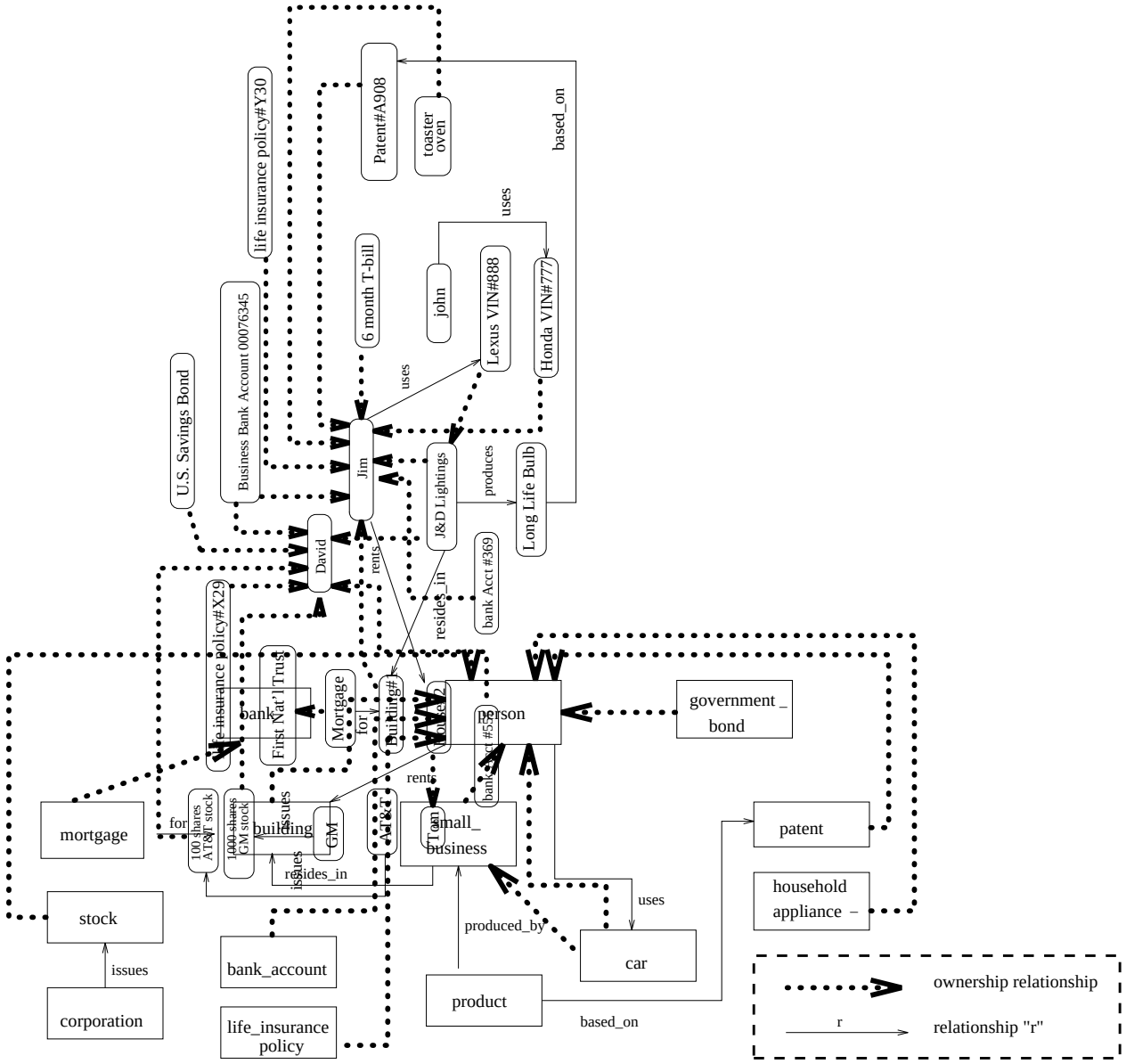

Figure 1: Instance and schema diagrams for first scenario.

ues of certain attributes of objects can be derived from other attributes via ownership relationships. We note that an investor's net worth can be determined directly from the values of his or her portfolios.

As can be gathered from the above, dealing with the issues of ownership can be very complex. To model such behavior in the context of a traditional relational database system [3, 4] would require that some programmer write programs (apart from the database system itself) to ensure that all business transactions are carried out in accordance with the semantics of ownership. However, if one were to bypass such programs, then no guarantee could be made about the database's integrity. Certain "illegal" ownership transactions may be allowed to commit, leaving various constraints violated and the database in an invalid state.

In an OODB system, the programmer could incorporate the additional constraint satisfaction code into the methods of any classes that participate in ownership relationships. However, this would still require manual programming labor. Furthermore, the correctness of the code would be very difficult to verify given all the subtleties of ownership. Ensuring that the programmer has properly accounted for all the necessary constraints in the program code would be nearly impossible.

In order to avoid such *ad hoc* programming approaches and remove the burden from the programmer, we present an ownership relationship model that expands and enhances an existing OODB data model. This relationship serves to encapsulate the rich semantics of ownership and its related trans-

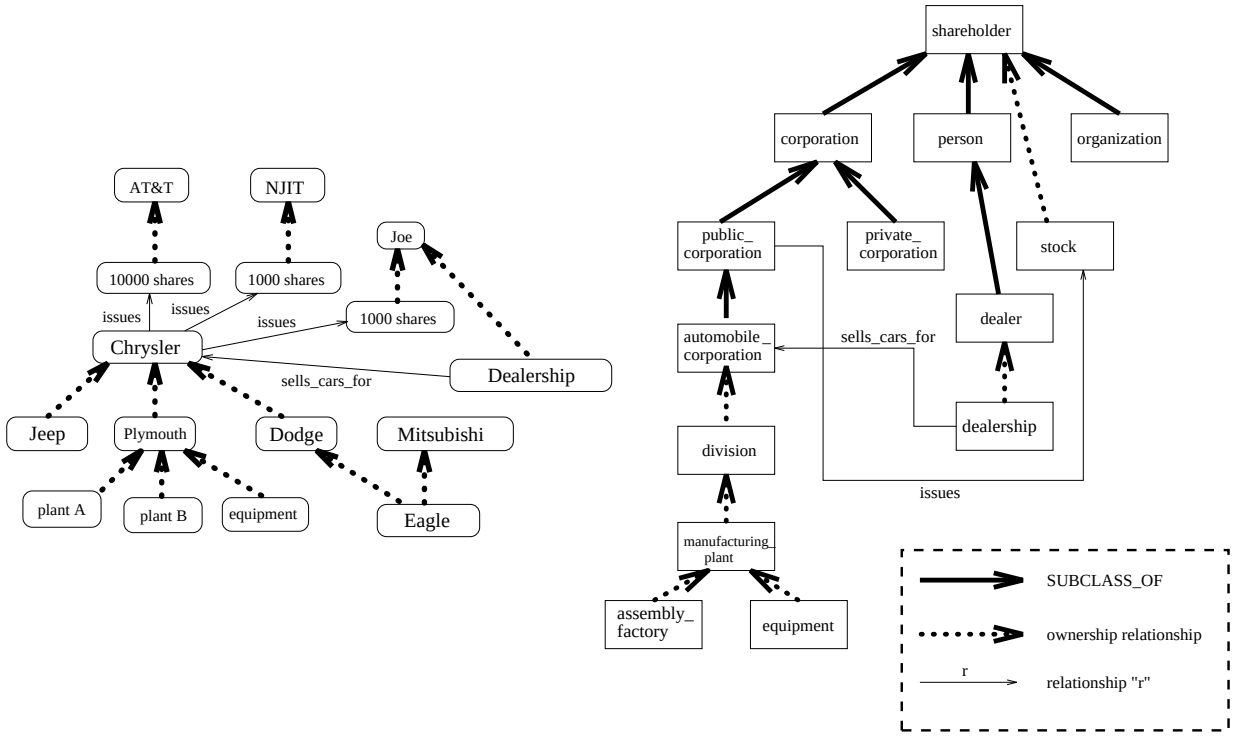

Figure 2: Instance and schema diagrams for second scenario.

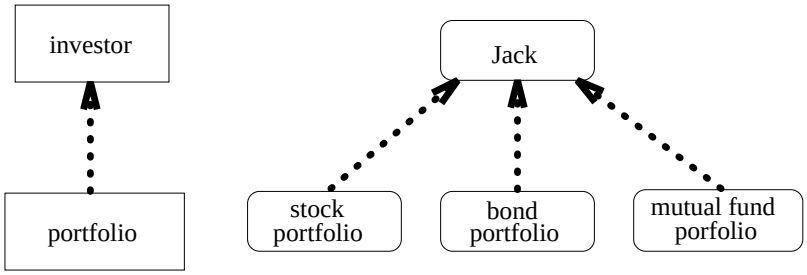

Figure 3: Schema and instance diagrams for third scenario.

actions. Using it, one can declaratively specify the desired behavior and then allow the OODB system to enforce it. In this sense, the ownership relationship is a "semantic relationship," meaning that its interpretation does not lie solely "in its name" [13] but rather in its constraint-satisfaction and inheritance mechanisms. In this light, our ownership relationship can be viewed as a modeling primitive of an OODB system with built-in semantics. In previous research, the IS-A (or SUBCLASS) relationship [2, 12] and the PART-OF relationship [6, 7, 8, 10, 11] have been modeled as semantic relationships.

The remainder of this paper is organized as follows. In Section 2, we discuss the legal definition of ownership. In Section 3, we formally define the ownership relationship and describe its "characteristic" dimensions, which capture the wide range of distinctions exhibited by ownership. Section 4 contains concluding remarks.

2 Definition of Ownership

When we describe a state of "ownership," we must, in general, include the following three features: (1) The owner, (2) the property that is owned, and (3) the characteristics of the relationship between the two. We are interested in identifying what types of objects can fill the roles of (1) and (2), and what the characteristics that distinguish the various kinds

of ownership are.

According to *Webster's Dictionary*, *ownership* is defined as follows:

1. The state or fact of being an owner.

2. Proprietorship; Legal right of property; Legal or just claim or title (to something); in law, the right to use for one's own advantage some property.

The owner referred to above can, by law, be a natural person, a corporation, or an organization. The latter two are, in general, referred to as *legal entities*. Under the law, legal entities are vested with certain powers, some of which are also held by natural persons. Others, like the power to exist in perpetuity, are unique to legal entities. In our databases, we see that Jim as a natural person owns his business. The Chrysler Corporation as a legal entity owns Dodge. In Fig. 1, *bank*, *small_business*, and *corporation* are legal entities. All "owner" classes in Fig. 2, except *person*, represent legal entities.

Ownership of an item is often distributed among persons and legal entities. E.g., Jim and David together own a business *J&D Lightings*, and a business bank account. Also, the Eagle Corporation is a joint venture of Chrysler and Mitsubishi. We describe such a situation as *joint ownership*. It is legitimate for a person and a company to jointly own a property. The ownership need not be divided into equal portions. Stock holdings partition the ownership of a public company into various percentages. Jim, e.g., owns thousand shares of Chrysler.

In law, *property* means the rights which one has in anything subject to ownership, whether it be mobile or immobile, tangible or intangible, visible or invisible. Ownership is used synonymously with rights in property. Thus, a person is said to be the owner of a property if he has certain rights in it. The term ownership is often used to indicate that one has the "highest rights" [1] in a property, but it may be used even when one does not have all the rights; thus, we say that a person is an owner of a house even though he has rented it to a tenant who has exclusive rights to the use of the house during the term of the lease [1].

A property can be classified as *real*, *intellectual*, or *personal*. A *real property* refers to the rights that one has in land or things closely related to it. An *intellectual property* is the rights held on an idea (e.g., the design of an invention) or a creative work (such as a musical composition or a novel). For such property, the rights apply to a potentiality—no claim is made on any tangible item. Copyrights and patents are the ordinary forms of intellectual property. *Personal property* encompasses everything that is not a real or intellectual property.

As examples, Jim's business resides in a building which is his real property. The patent (number A908) for the Long-Life Bulb is his intellectual property. Bank account 369 and the car used by John are his personal property. In Fig. 1, the class *building* denotes a real property. *Patent* is an intellectual property. The remainder of the "property" classes represent personal properties. In Fig. 2, the only real property is *manufacturing_plant*. The rest are personal properties.

One characteristic of the ownership relationship itself centers around the existence of a legal document that verifies the owner's rights to a property. A copyright owner, e.g., is granted a legal certificate giving him exclusive rights to possess, make, publish, and sell copies of his intellectual production, or to authorize others to do so. In contrast, the owner of a household item does not have a legal document to support his ownership, but he has the right to use it as he pleases. We call ownership of the former kind *documented* and ownership of the latter kind *undocumented*. So, Jim's patent is documented, while his ownership of a toaster oven is undocumented.

In Fig. 1, the following ownerships (written as: owner class-property class) are among those that can be classified as documented: *bank-mortgage*, *person-building*, *person-small_business*, *person-bank_account*, *person-patent*, and *small_business-car*. The relationship between the classes *person* and *household_appliance* is undocumented. All ownerships in Fig. 2 are documented.

As a final distinction, some kinds of ownership are acquired by operation of law, while some others are not. We call ownership of the former kind *de jure* and ownership of the latter kind *de facto*.

3 Ownership as an OODB Semantic Relationship

3.1 Transactions and Inheritance

As noted above, the most crucial aspects of ownership are the constraints that it imposes on its related transactions such as sale and lease. Certain transactions can be applied to specific kinds of ownership, while others cannot. For example, in the case of exclusive ownership, the owner can sell his belonging without restriction (and thus the transaction "sale" can be applied freely), while for joint ownership an owner cannot sell the property without

the consent of the other owners (so the use of "sale" must be controlled). When a person has accepted an offer to sell his house, he cannot accept another offer, even though he is still the owner, until that time when the first offer becomes invalid. We call ownership of this kind *action-limited*. Similarly, when one has bought a stock option, the ownership of it may expire after a certain period of time if it is not exercised. In this case, we say that the ownership is *time-limited*. Likewise, when one has ownership of some property like a car or a house, it cannot be sold without its supporting documentation.

Let us consider some of these complexities of ownership transactions in the context of our example scenarios. If Jim wants to sell the business, he needs the consent of David, his partner. If David wants to buy half of the business's building from Jim, then he must have the consent of First Nat'l Trust which owns the mortgage. What would happen if David wanted to sell his half of the company to a new partner? Depending on the partnership agreement, he may need Jim's approval. With respect to their joint checking account, do both Jim and David need to sign every check together? Clearly that depends on the nature of the account. What about the sale of properties that are being used by others? For example, can Tom sell the house that he is renting to Jim? Yes, but the new owner would be unable to occupy the house until the lease expired. Is John allowed to sell his father's car? No, because even though he is using the car, he does not possess the proper ownership documentation required to sell it.

Aside from the transactions, the ownership relationship plays a vital role in more accurately modeling various application domains via its inheritance mechanism, which allows values of certain attributes to be propagated across it. For example, Jack's net worth (i.e., his "value") can be determined directly as the sum of the values of the portfolios that he possesses. Consider also that to calculate Chrysler's profits for 1994, the profits of Dodge, Plymouth, and Jeep must be added together. Furthermore, the profits of Dodge must take into account the profits of Eagle. In all these examples, a value propagation between properties and owners is required.

From the above we see that to properly support transactions and inheritance with respect to ownership, we need to explicitly model the different characteristics (which we call the *dimensions*) of the ownership relationship. Our investigation has revealed six important dimensions. In this section, we will first formally define the ownership relationship and its constituent dimensions. Thereafter, we will examine two of the dimensions, *exclusiveness* and *value propagation*, in some detail. We will then briefly describe the others.

3.2 Formal Definition of the Ownership Relationship

Let $E(C)$ denote the extension of a class C, i.e., the set of all its instances. The ownership relationship between a property class B and an owner class A (denoted $O_{B,A}$) is defined as the following septuple:

$$O_{B,A} =< \Omega_B^A, \lambda, \beta, \alpha, \chi, \delta, \nu >$$

where Ω_B^A is a relation from $E(B)$ to $E(A)$. The pair $(b, a) \in \Omega_B^A$ indicates that the instance b of class B is the property of (i.e., is owned by) the instance a of class A. We will ordinarily express this fact as $b\Omega_B^A a$. The remaining elements of the septuple are the six characteristic dimensions, whose names are *Legality*, *Documentation*, *Limitation*, *Exclusiveness*, *Dependency*, and *Value Propagation*, respectively. For each, we list its domain in the following:

$\lambda \in \{$*de jure*, *de facto*$\}$,
$\beta \in \{$*registration-docum'ted*, *transfer-docum'ted*, *undocum'ted*$\}$,
$\alpha \in \{$*action-limited*, *time-limited*, *action&time-limited*, *unlimited*$\}$,
$\chi \in \{$*exclusive*, *free-joint*, *percentage-joint*, *global-percentage-joint*$\}$,
$\delta \in \{$*owner-to-property*, *nil*$\}$,
$\nu \in \{$*up*, *down*, *upTrans*, *downTrans*, *up&down*, *nil*$\}$.

The values of both δ and ν may be *nil*, indicating that the particular characteristic (dependency or value propagation) is inapplicable. For lack of space, formal descriptions of only two dimensions will be given. The rest of the dimensions are described formally in [5, 14]. (In [14], we had only four dimensions and the notion of transactions is not connected to the dimensions of ownership.) For the following definitions, assume an ownership relationship $O_{B,A}$.

Definition 1: $\forall a \in E(A)$, let $P_{\Omega_B^A}(a) = \{b \mid b \in E(B) \wedge b\Omega_B^A a\}$. $P_{\Omega_B^A}(a)$ is called the *property set* of a with respect to $O_{B,A}$, i.e., the set of instances of B which are properties of a.

Definition 2: $\forall b \in E(B)$, let $N_{\Omega_B^A}(b) = \{a \mid a \in E(A) \wedge b\Omega_B^A a\}$. $N_{\Omega_B^A}(b)$ is called the *owner set* of b with respect to the ownership $O_{B,A}$, i.e., the set of instances of A of which b is a property.

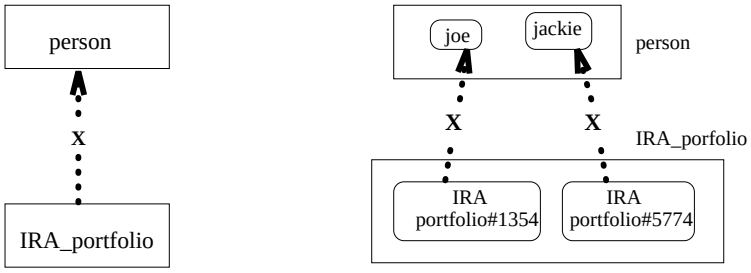

Figure 4: An example of exclusive ownership.

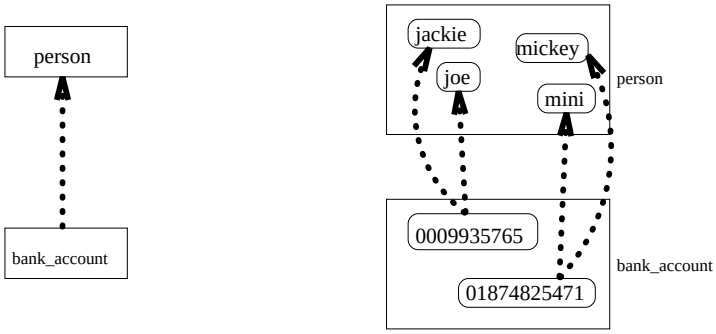

Figure 5: Jointly owned bank accounts.

3.3 Exclusiveness Dimension

Ownership can be classified as exclusive or joint. In other words, a property may be owned by one owner or jointly owned by several owners. The formal definition for the exclusive ownership relationship follows:

Definition 3: For the ownership relationship $O_{B,A}$, χ = *exclusive* implies that $\forall b \in E(B)$, $|N_{\Omega_B^A}(b)| \leq 1$. In other words, a property cannot have more than one owner.

To represent this graphically, we add an **X** to the dotted arrow to denote e**X**clusive (Fig. 4).

Those ownership relationships which are not exclusive are referred to as *joint*, in which case a property may be either jointly owned freely, i.e., there is no explicit partition of the rights of the joint owners in the property (e.g, a joint bank account is freely shared by a couple—we call this *free joint*), or jointly owned such that each owner takes a certain percentage of the ownership (e.g., husband and wife each own 50% of their house—we call this *percentage joint*). We call the case where all owners have the same percentage *equal joint*. Although the exclusiveness dimension has been included in some OODB models (e.g., SHOOD [11] and our part relationship model [7, 8]), percentage joint is unique to ownership. Percentage joint plays an important role in economic activities. A shareholder has the right to receive his percentage of dividends.

In our graphical notation, a plain dotted arrow indicates free joint (Fig. 5). Percentage joint and equal joint are denoted by labels of **P** and =, respectively (Fig. 6).

Definition 4: For the ownership relationship $O_{B,A}$, χ = *free-joint* implies that $\forall b \in E(B)$, $O_{B,A}$ does not impose any constraints on $|N_{\Omega_B^A}(b)|$. That is, each instance b may have any number of owners.

Definition 5a: For the ownership relationship $O_{B,A}$, χ = *percentage joint* implies that $\forall b \in E(B)$, each of its owners a has an associated number $p_{b,a}$ $(0 < p_{b,a} \leq 100)$ indicating a's percentage of ownership of b. The percentages $p_{b,a}$ associated with all the owners of b must total 100%.

Definition 5a defines the *percentage joint* ownership relationship when the property class has only one associated owner class. At times, the ownership of an object may be distributed among owners from different classes. This case is defined as follows.

Definition 5b: The ownership relationships O_{B,A_1}, $O_{B,A_2}, \ldots, O_{B,A_n}$ are *global percentage joint* if $\forall b \in E(B)$, each of its owners (regardless of their classes) own percentages of b totaling 100%.

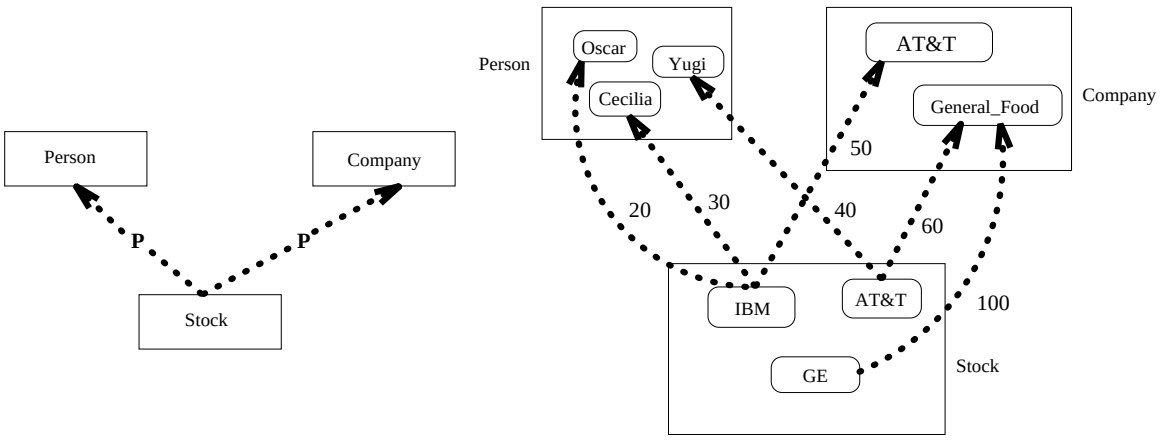

Figure 6: Stocks are owned (percentage) jointly by person and company.

To better understand Definition 5b, refer to Figure 6, where $O_{Stock,Person}$ and $O_{Stock,Company}$ are two *global percentage joint* relationships. For any instance of class *Stock*, the ownership is distributed among its owners such that each of them takes a certain percentage and the sum of the percentages is 100%. In Figure 6, the IBM stock owners are Oscar and Cecilia of class *Person*, and AT&T of class *Company*, with 20, 30, and 50 percent of the ownership, respectively.

3.4 Value Propagation Dimension

There are times when a certain feature of a property is naturally assimilated as a feature of its owner, or vice versa. E.g., the address of a person may be modeled as the address of his house rather than as an intrinsic attribute of the person. Likewise, the name that appears on the passport can be taken to be the name of its owner. In the former case, the value of *address*, rather than being duplicated, should be stored solely with the house and propagated upward on demand. *Address*, in this sense, is a derived attribute of person.

As another example, Jack's net worth can be determined directly from his portfolios. Specifically, Jack's net worth (denoted as his "value") is just the sum of the values of his various portfolios. As these fluctuate on a minute-to-minute basis, so too should Jack's worth. Therefore, it does not make sense to store this value statically. Rather, it should be derived dynamically from the appropriate sources on demand. The ownership relationship can automatically (i.e., without the need for manual programming) perform the necessary retrieval and computation.

Definition 6: Let $\pi_B : E(B) \rightarrow \tau$ be an attribute of B. The ownership relationship $O_{B,A}$ is said to be *invariant upward propagating* if it defines a property π_B on the class A such that the value of π_B for an instance $a \in E(A)$ is identically the value of π_B for that $b \in E(B)$ which is owned by a.

For example, if the property *address* is propagated from the class *house* to the class *person*, then the ownership relationship would define the property *address* on class *person* as follows:

$$address(a) = \begin{cases} address(b), & \text{if } \exists b \in P_{\Omega_B^A}(a), \\ \text{undefined}, & \text{otherwise.} \end{cases}$$

Thus, the address of a person is identically that of the house that he or she owns. Invariant propagation in the other direction is defined analogously (see [5]).

Transformational upward value propagation is designed to take contributions for the value of the propagated (or inherited) attribute from any number of objects that are owned. The multiple values are transformed into a single value of the attribute's data type.

Definition 7: Let $\pi_B : E(B) \rightarrow \tau$ be an attribute of B. The ownership relationship $O_{B,A}$ is said to be *transformational upward propagating* if it defines a property π_B on the class A such that the value of π_B for an instance $a \in E(A)$ is derived by applying some transformation collectively to the values of π_B for all $b \in E(B)$ such that b is owned by a.

Here, instead of being identical to a value at a single "property" object, the value of the propagated attribute is derived through a transformation of values from many owned objects. For the example of the net worth of an individual, the propagated property *value* would have the following definition:

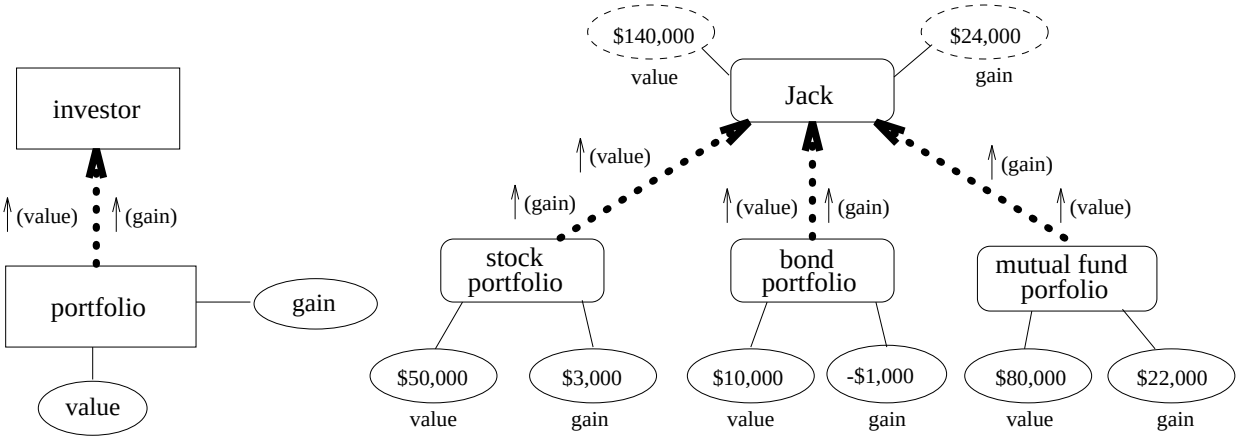

Figure 7: An example of value propagation.

$$value(a) = \begin{cases} \sum_{i=1}^{n} value(b_i), & \forall i,\ value(b_i) \text{ defined} \\ \text{undefined}, & \text{otherwise,} \end{cases}$$

where a is an investor and $b_1, b_2, \ldots, b_n$ are his portfolios. The above is shown graphically in Fig. 7, where we also show the specific example of Jack obtaining his net worth from his three portfolios. Another derived attribute, an investor's total gain (which is just the sum of the gains of the portfolios), is shown in the figure as well.

3.5 Additional Dimensions

Due to space limitations, we mention the issues of several other dimensions only briefly. For details, see [5, 14]. The dependency dimension regulates the semantics of deletion of owner class A or property class B. It defines when deletion of one should cause deletion of the other. Ownership can be either *documented*, or *undocumented*. Documented ownership always has a supporting legal document, while undocumented ownership does not.

Some kinds of ownership are acquired "by operation of law," i.e., through a formal legal procedure. We call such ownership *de jure*. Others are not, and are called *de facto*. These are the values for the legality dimension. Ownership is often used to indicate the "highest rights," but it may be used when one does not have all the rights. In other words, ownership may be limited in some aspects. For example, if the owner of a house has accepted an offer to sell that house to someone, then he cannot sell it to some other person, even though he is still the owner, unless the offer becomes invalid.

4 Conclusion

We have addressed the issue of representing ownership relationships in OODBs with a model that captures a variety of semantics. In particular, we have distinguished a number of aspects for the roles of the owner and property in such relationships. These aspects define notions like exclusive and joint owners. Formal definitions for various ownership relationships were presented. To complement these, we have presented graphical symbols for each of the ownership relationships which expand the OODini graphical schema representation language for OODBs [9]. We have also investigated the interaction between the various ownership transactions and the ownership relationship's characteristic dimensions. We plan to integrate the ownership relationship that we have defined here into a commercial OODB system.

Acknowledgment

We thank Stewart Klein for reading an earlier version of this paper and for his many important remarks regarding the legal issues surrounding ownership.

References

[1] R. Anderson, W. Kumpf, and R. Kendrick. *Business Law — Principles and cases.* South-Western Publishing Co., Cincinnati, OH, 1971.

[2] R. J. Brachman. What IS-A is and isn't: An analysis of taxonomic links in semantic networks. *Computer*, 16(10):30–36, Oct. 1983.

[3] E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387, 1970.

[4] C. J. Date. *An Introduction to Database Systems*, volume 1. Addison-Wesley Publishing Co., Inc., Reading, MA, fourth edition, 1986.

[5] J. Geller, M. Halper, O. Yang, and Y. Perl. Exploring the semantics of ownership relationships. In preparation, 1995.

[6] M. Halper. *A Comprehensive Part Model and Graphical Schema Representation for Object-Oriented Databases.* PhD thesis, CIS Department, New Jersey Institute of Technology, 1993.

[7] M. Halper, J. Geller, and Y. Perl. An OODB "part" relationship model. In *Proceedings of the First International Conference on Information and Knowledge Management*, pages 602–611, Baltimore, MD, 1992.

[8] M. Halper, J. Geller, and Y. Perl. Value propagation in object-oriented database part hierarchies. In *Proceedings of the 2nd Int'l Conference on Information and Knowledge Management*, pages 606–614. Washington, DC, 1993.

[9] M. Halper, J. Geller, Y. Perl, and E. J. Neuhold. A graphical schema representation for object-oriented databases. In R. Cooper, editor, *Interfaces to Database Systems*, pages 282–307. Springer-Verlag, London, 1993.

[10] W. Kim, E. Bertino, and J. Garza. Composite objects revisited. In *Proc. of the 1989 ACM SIGMOD International Conference on the Management of Data Portland, Oregon, appeared as SIGMOD RECORD*, pages 337–347, 1989.

[11] G. T. Nguyen and D. Rieu. Representing design objects. In *AI in Design'91.* Butterworth-Heinemann Ltd., 1991.

[12] A. Snyder. Encapsulation and inheritance in object-oriented programming languages. In *Proc. OOPSLA-86*, pages 38–45, 1986.

[13] W. A. Woods. What's in a link: Foundations for semantic networks. In D. G. Bobrow and A. M. Collins, editors, *Representation and Understanding*, pages 35–82. Academic Press, New York, NY, 1975.

[14] O. Yang, M. Halper, J. Geller, and Y. Perl. The OODB ownership relationship. In *Proceedings of the Int'l Conf. on Object Oriented Information Systems*, pages 389–403, London, England, 1994.