An Alternative Solution to the Observation Pattern Problem

Kinh Nguyen
Department of Computer and Mathematical Sciences
Victoria University of Technology, Melbourne, Australia 3000
kinh@matilda.vut.edu.au

Tharam Dillon
School of Computer Science and Computer Engineering
La Trobe University, Melbourne, Australia 3083
thdillon@latcs1.cs.latrobe.edu.au

Abstract

We have used Fowler's Observation pattern (published in [Fowler 97a]) in a project involving the maintenance of medical records of patients. The pattern has been modified to suit the project's particular requirements. We end up with a substantially modified model which could be regarded as an alternative solution. In this paper, we present that alternative model and the underlying rationale.

We also suggest, as a separate issue, how the study of analysis patterns can be improved by using a suitable 'minimal' application, an application just big enough to cover the pattern in question.

Keywords: analysis pattern, object business modelling, object oriented analysis, reuse, Object-Z, Smalltalk prototyping.

1 Introduction

Fowler's book, *Analysis Patterns: Reusable Object Models* [Fowler 97a], is an invaluable catalog of analysis patterns.

The Observation pattern, in particular, brings to the reader's attention many important issues. In our experience, they are issues that, unless care is taken, could easily be overlooked. We have faced precisely those issues in a project in which we perform the analysis, design and prototyping of an a Patient History System for a hospital which needs to monitor the effects of treatments of the patients over a period of time.

It turns out that Fowler's analysis pattern does not suit our application context as well as we wish. This is a situation which is not totally unexpected of analysis patterns in general. Consequently, we have modified it to suit our proposes. The modification is sufficiently substantial that it could be regarded, in our opinion, as an alternative solution.

In this paper, we will present Fowler's model (section 3), and describe our context for applying the Observation pattern (section 4), followed by the reasons why we need to modify it (section 5). We then present our solution (section 6), and a discussion of some of the related issues (section 7).

2 The Observation Pattern's Requirements

The Observation pattern is concerned with the recording of various kinds of observations about people or things. The most important characteristics of this pattern is that we have information about both

- the types of observations, and
- the individual observations themselves.

The relationship between the types of observations and the observations is equivalent to that between classes and their instances. However, both the types of observations and the individual observations are maintained as objects, as opposed to classes, in the system.

To put the Observation pattern in a concrete context, we can consider the maintenance of medical records of patients; and for the sake of definiteness, we will follow the requirements given by Fowler in Chapter 3 of [Fowler 97a].

These requirements are actually very close to the requirements of our project. Some of the differences will be pointed out later.

Essentially, we need to record three kinds of observations and one kind of relationship:

- *Measurement observations*, e.g. the height of a patient.
- *Category observations*, e.g. the patient's gender which can be male or female.
- Observation regarding the *presence/absence* of a characteristic, e.g. we may be interested in recording whether a patient suffers from high blood pressure.
- *Sub/super classification relationships* among types of observations. For example, diabetes can be subdivided into *type 1 diabetes* and *type 2 diabetes*. We will say that *diabetes* is a super-classification of *type 1 diabetes* and the latter is a sub-classification of *diabetes*.

  This kind of relationship has an importance consequence: the absence of diabetes implies the absence of *type 1 diabetes* and *type 2 diabetes*, whereas a presence of *type1* or *type 2 diabetes* implies a presence of *diabetes*.

  This kind of relationships also exists among various categories. For example, blood group *A* is subdivided into subgroups *A1* and *A2*.

The question is: What sort of object oriented model should we use to describe the situation?

3 Fowler's Observation Pattern

The solution proposed by Fowler (in [Fowler 97a], Figure 3.9) is reproduced with minor changes in notation in Figure 1.

The diagram is divided into two parts by the dashed line. The part of the model above the line represents the kind of information that does not depend on the particular population of patients, and the part below it represents the kind of information that does. Fowler uses the terms *'knowledge information'* and *'operational information'* to refer to these two types of information respectively. Knowledge information changes very infrequently, whereas operational information changes on the basis of daily operations.

We have found Fowler's distinction between those two levels of information to be very useful.

The way the model is intended to record medical observations is shown in Figure 2. It shows how the model records the fact that Smith is 175 cm tall, has type B blood group and suffers from diabetes. In general, we have:

- A measurement observation will have a phenomenon type and a quantity associated with it. It will not be associated with any phenomenon. A quantity consists of an amount (a numerical value) and a unit.

  For example, the height of a patient is a measurement observation of phenomenon type *Height* and may have the quantity of amount *175* in *cm* unit.
- A category observation will be associated with a phenomenon, which in turn is associated with a phenomenon type.

  For example, the blood group of a patient is a category observation of phenomenon type *Blood-Pressure* and may have the phenomenon *B* say.
- The presence/absence of a "property" is recorded by a *Presence/Absence* instance which has a reference to an instance of *Observation Concept*. An observation concept does not have any *Phenomenon Type* associated with it.

  For example, the presence of diabetes for a patient is recorded by a *Presence* instance which refers to Observation Concept *diabetes* which is not related to any *Phenomenon Type*.

4 Our Context for the Application of Observation Pattern

We are concerned with the analysis, design and prototyping of the Patient History System (PHS) for a private hospital. The following is an extract from the initial requirements statement, which was drawn up in consultation with the doctors involved.

> The PHS is to facilitate the recording and retrieval of the medical histories of the patients. The doctors will be involved from within and outside the hospital, with some providing the primary care and the remainder specialist care.

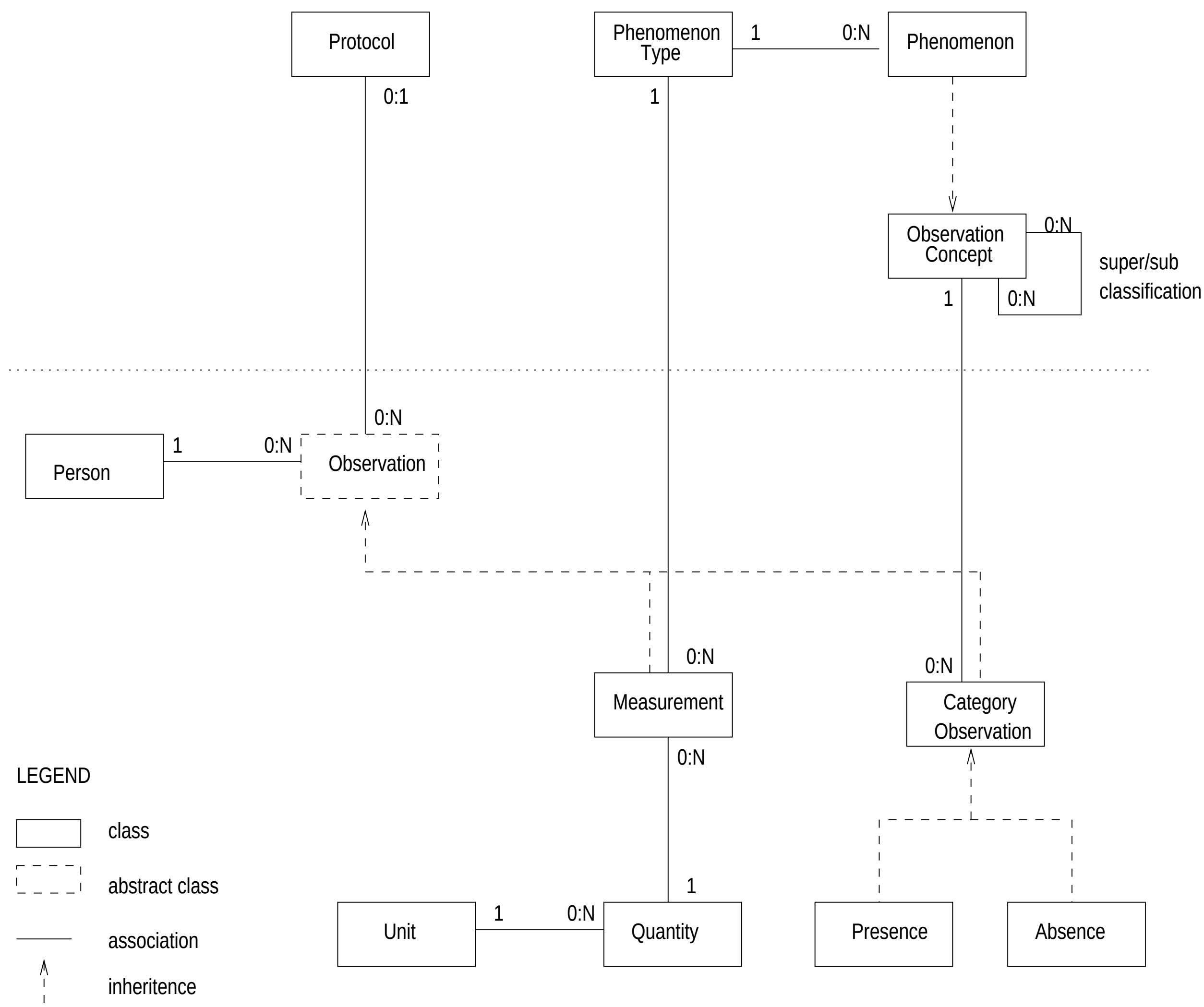

Figure 1: Fowler's Observation Pattern.

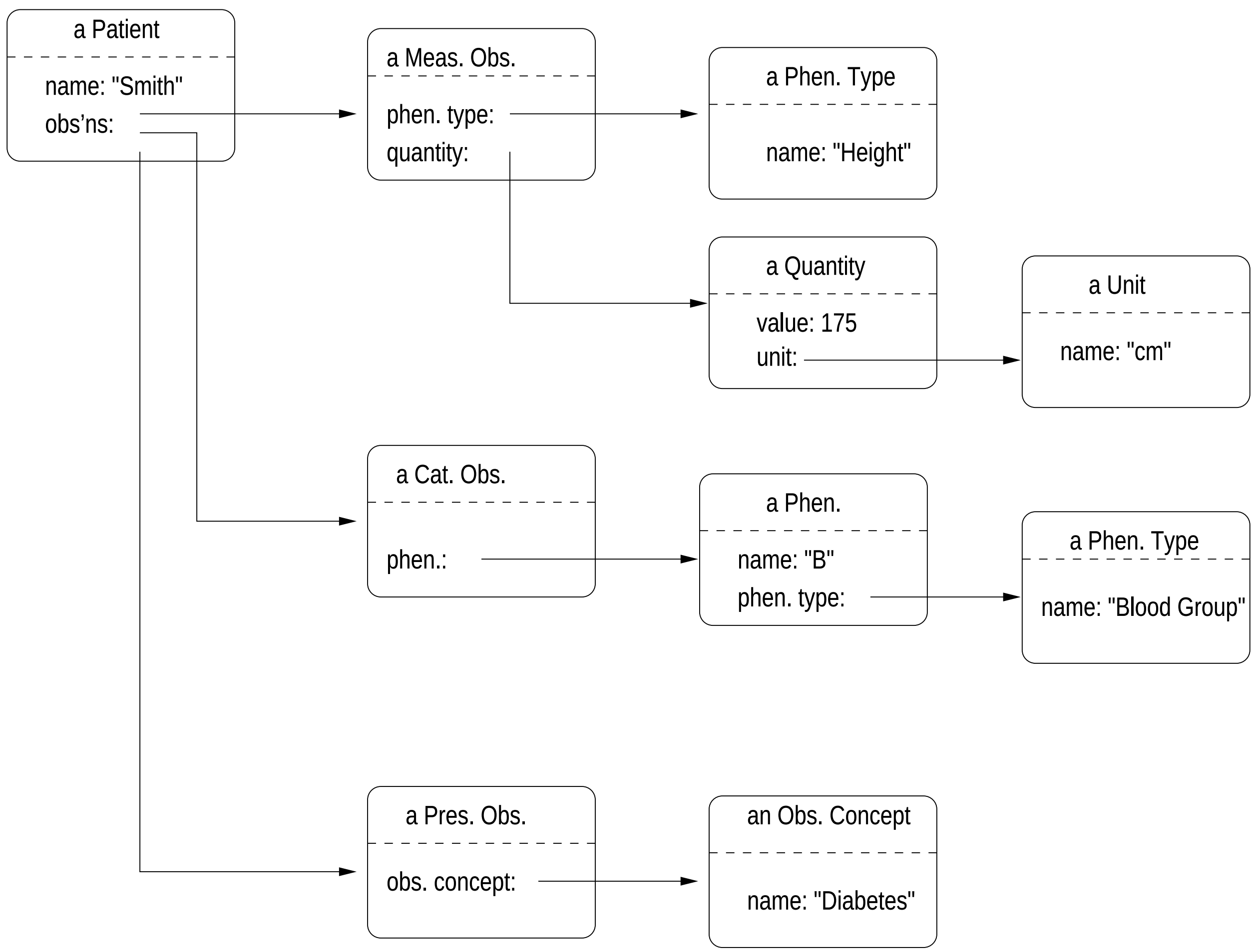

Figure 2: Sample Instance of Fowler's Observation Pattern.

At any time a patient will be in the care of one or two doctors, being placed on one or more medications, perhaps none.

Events in the medical history will include:-

- being placed in / removed from the care of a particular doctor;
- a physiological measurement such as *CD4-cells* or *viral load*;
- the beginning / ending of a particular medication;
- the development of a particular complication such as pneumonia or meningitis.

Data will be entered into the PHS by a clerical assistant. Data-entry will include :-

- creating / deleting a patient or a doctor;
- updating patient and doctor attributes such as address or phone number;
- creating a new physiological measurement;
- creating a new medication;
- entering events in the medical history of a patient.

A prime requirement of the PHS is that the user be able to see immediately the interaction between the physiological measurements and treatments. The various components of the medical history need to be seen in parallel. For example, with time as the horizontal axis, the physiological measurements could be displayed as graphs, with duration of treatments being displayed as horizontal bars below the graphs, and the occurrence of complications by vertical arrows.

As can be seen from the extract, the maintenance of medical observations on the patients is a central part of the project. As mentioned earlier, the requirements for the observations, upon further analysis, turn out to be very close to those described in section 2.

Remarks:

1. Observations that have both a measurement and an associated category do exist in our application domain, however, they have not been included for practical reasons: they will be recorded merely as measurement observations. Such observations are considered in [Fowler 97b], Chapter 11 and are briefly discussed in section 7 of this paper.
2. We also allow each type of measurement observation to have only *one* kind of unit associated with it. This restriction is not quite true in the application domain, but in the project we can impose it without harm. The model we propose can be easily modified if we wish to remove this restriction (as can be seen in section 6).

5 Our Needs to Modify Fowler's Model

Fowler's model allows us to record the three kinds of observations and the sub- super-classification relationship. However, in our project, one of the emphases is to try to capture as much information about the observations at the knowledge level as possible and make use of them for the entry of operational data. It is to fulfill this requirement that we need to modify Fowler's pattern. More specifically,

- *We need to capture more knowledge information about measurement observation types*

  At the knowledge information level, each measurement observation type has some important information associated with it. For example, the *unit* of measurement: a height can be recorded in *cm* or *feet* and *inches* but not in *kilogram.* It is important to record this kind of information at the knowledge level to be able to ensure the validity of observations at the operational level.

  Fowler's model, as it stands, does not allow us to capture this kind of information. And to do that, we need to model measurement observation type explicitly.

- *We need to enforce some relationship constraints that are not enforced by Fowler's model*

  The fact that a measurement observation has no phenomenon associated with it is an important piece of knowledge information. This fact is *permitted* by Fowler's model by the virtue of the relationship multiplicity which states that phenomenon type can have zero or more phenomena. But it is *not enforced* by therein.

  In other words, this model cannot prevent the misuse of having a measurement observation with phenomena attached to it. (NOTE: There are cases where this possibility may be desirable. See for example [Fowler 97a], chapter 3.)

Similarly, the fact that each CategoryObservation instance is associated with a Phenomenon instance is permitted by the model but is not enforced by it.

Again, to enforce such relationship constraints, we find it necessary to model the observation types (as opposed to the individual observations) explicitly.

6 The Alternative Solution

Our alternative solution is given in Figure 3. The diagram is divided into two parts by a dotted line. The left half represents knowledge information and the right half represents operational data.

An example of how the model can be used to record data is given in Figure 4 to record the same facts as those in Figure 2.

It can be seen that in the proposed solution

- The three kinds of observations are modelled explicitly at both knowledge and operational levels.
- The knowledge information about the observation types can be captured clearly and as fully as we wish. This is made possible by the explicit representation of the three kinds of observation types. This explicit representation is necessary because each kind of observation has a *different* "knowledge structure".

  For example, for each measurement observation type, we can record the range of valid values and the associated unit, as shown in Figure 4 On the other hand, each category observation type is associated with a number of categories of its own.
- Each observation is associated with an observation type as shown in Figure 3. Note that we have not shown how this relationship is materialized at the subclass level. That is, we have omitted the clearly intended 'lines of association' between the MeasurementObservation class and the MeasurementObservationType class and so on.

  Naturally, the knowledge information can be used to validate the operational data. For example, we can verify that the value of an observation must fall within the valid range.
- The way we represent the three kinds of observations / observation types seems to conform closely to the way people would think about the real world of the application domain. It presents a view that is natural and easy to comprehend.

  We have also found that this view allows us to record the information in a clear and consistent manner. For example, a data entry operation concerning observations corresponds to the 'creation' of either an observation type or an observation. Then depending on what type of observation we are dealing with, the data entry operation follows a particular pattern, either to completely characterize an observation type or to record an observation which is constrained to ensure compatibility with its observation type.

Remarks:

1. In our project, to each measurement type there associates only *one* unit. The unit is modelled as part of the information about the MeasurementObservationType. Each measurement of such a type is assumed to have the unit already recorded for that type. For that reason, we have not directly associated the unit to the measurement observation.

   If that restriction is removed, then each measurement observation type would be associated with one or more units, and this should be recorded as part of the knowledge data. And then for each measurement observation, we must record which of the permitted units the observation is to associated with.
2. We have chosen not to make the Category class a subclass of the PresAbsObsType class (as in Fowler's model). At the risk of splitting hair, we present below the reasoning behind our decision.

   Consider for example the 'Blood group A' and assume that there are cases in which we want to record the presence or absence of blood group A for a patient. With that assumption, we can envisage the following two scenarios. In the first scenario, we want to determine the blood group of someone and it happens to be the 'Blood group A'. In the second scenario, other case, we want to determine whether the blood group of a patient is A or not. In our opinion, we have in those two cases two entirely different views of 'Blood group A'. In one case 'Blood group A' just happens to be the value among a set of possible values. In the other one, the blood group A the

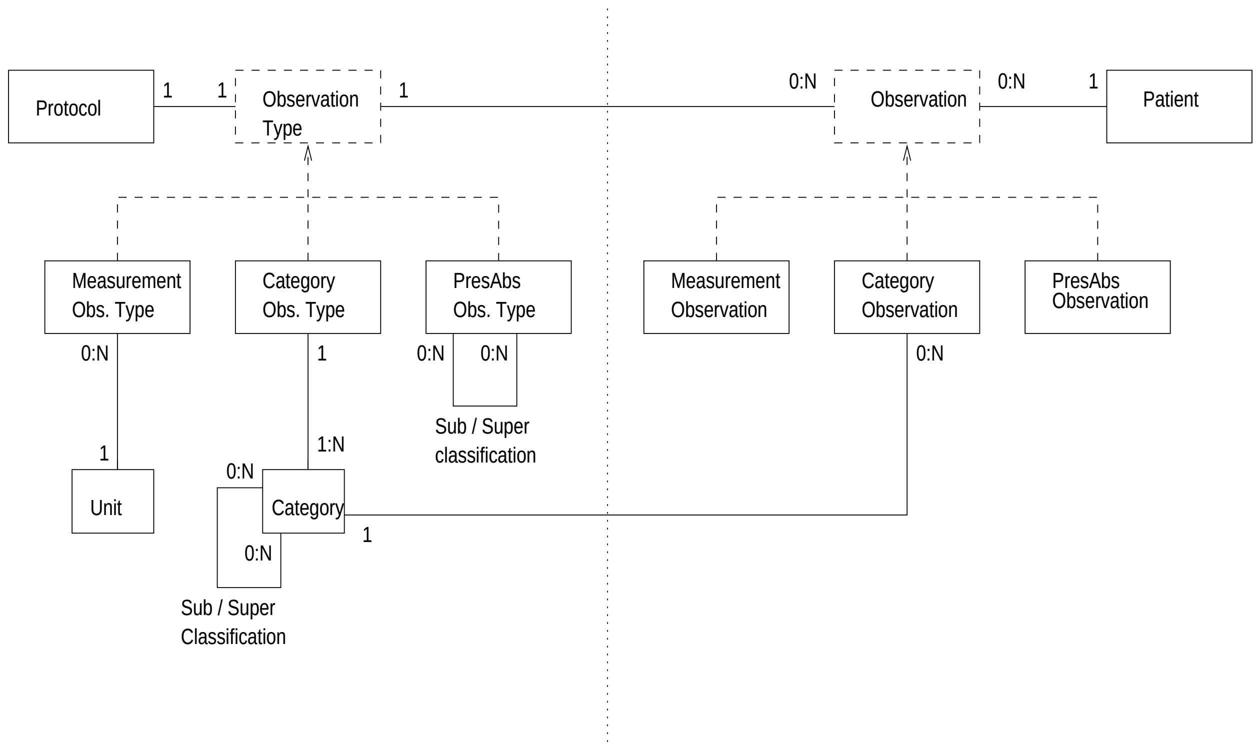

Figure 3: Suggested Observation Pattern.

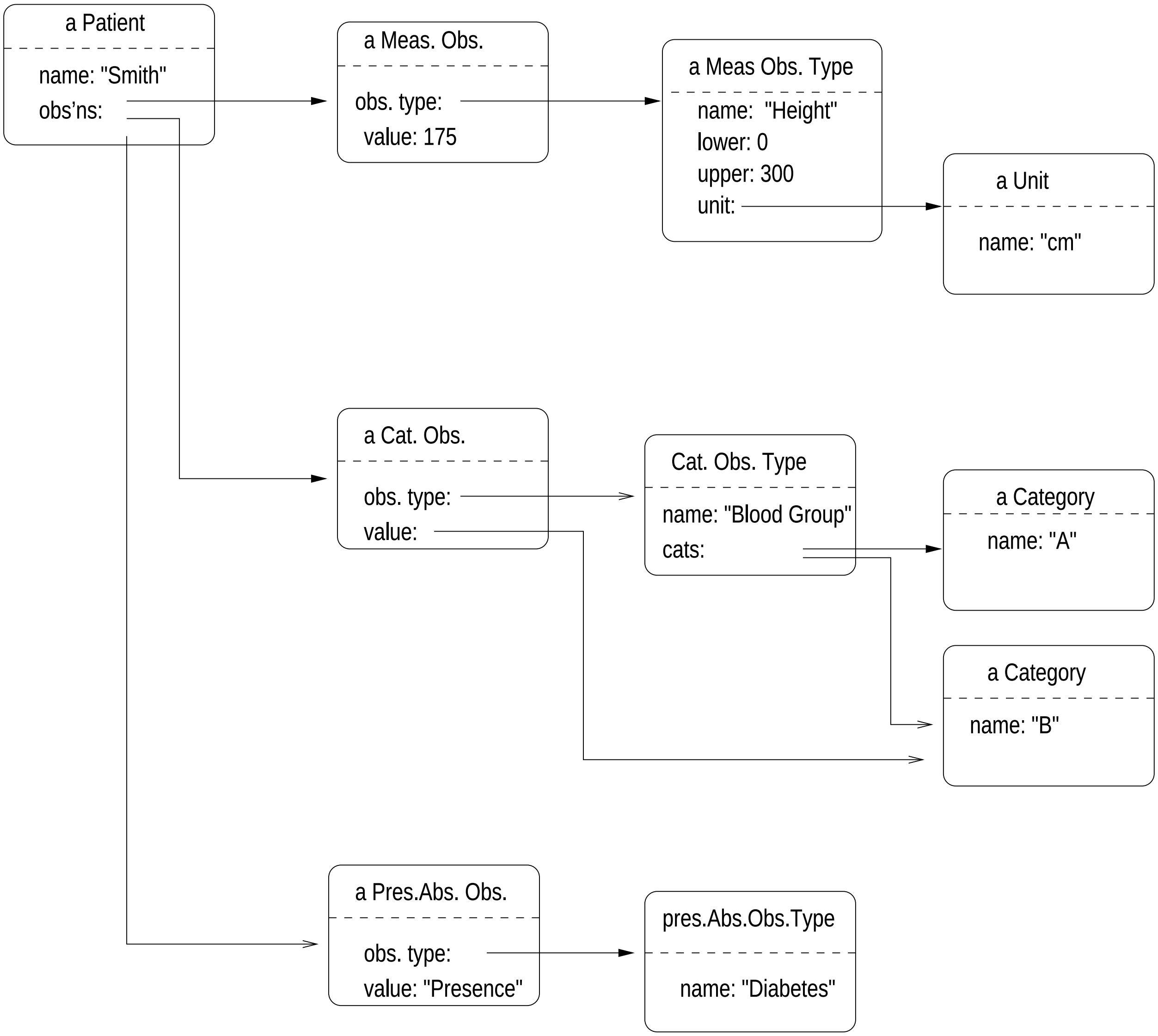

Figure 4: Sample Instance of Suggested Observation Pattern.

subject of investigation. Conceptually, we have two different, though related, concepts. The confusion arises from the fact that we refer to both concepts by the same label, namely 'Blood group A'. To capture the differences, it makes sense to have an instance of the Category class with label 'Blood group A' and an instance of the PresAbsObsType class with the same label. The two 'concepts' may be related to each other by an association, not an inheritance relationship.

7 Further Discussion of Related Issues

We will *briefly* discuss some of the related issues below.

7.1 Extension: Measurements with Associated Categories

In [Fowler 97b], Chapter 11, Fowler considers the case of observations that have both a measurement and an associated category. Figure 11-4 of the cited book takes the heart rate as an example. A patient can have the heart rate of 70 bpm which falls into the category of "normal heart rate", which ranges from 60 to 80 bpm.

- Fowler's model (as given in Figure 2) *can* cater for this case and we can do so only for the operational data.

  The knowledge information about the various categories of heart beat cannot be modelled in the given pattern as far as the class structure is concerned: there is no way to associate categories with ranges of values. Such information can be modelled as rules and implemented by the code of the instance methods.

- With our proposed solution, as it stands, we *cannot* record this kind of observations.

  To do that, we need to create a new kind of observation type which contains: a lower bound and an upper bound (valid range), a unit, and a set of "ranged" categories (each "ranged" category has a lower bound and an upper bound). We can then enforce the constraint that the category ranges have to be disjoint. Note that the change to the model does not affect the overall structure or the readability of the model at all.

7.2 Relating Fowler's Model and Ours

We could also relate the two models conceptually by noticing the following relationship:

> Model in Figure 3 = Model in Figure 1 +
> Rules on Relationship Constraints

For example, the following rule, when applied to the model in Figure 1, would require that a measurement observation will not be associated with any phenomenon:

$$\neg\,(\exists\, m : \mathit{Measurement};\ p : \mathit{Phenomenon} \bullet \\ m.\mathit{PhenomenonType}.\mathit{Phenomenon} = p)$$

It is not clear to us how we should enforce such a rule, given that we follow the model in Figure 1. The only way we can think of is this: whenever we add a new measurement observation, check that the associated phenomenon type is not associated with any phenomenon.

7.3 Minimal Applications for Analysis Patterns

Recognizing the importance of analysis patterns, we wish to study them effectively. To do that, we have adopted a simple, and obvious, approach using what we call a "minimal" application. It is an application which is just big enough to cover the analysis patterns. Most analysis patterns are sufficiently substantial to lend themselves to a reasonably interesting minimal application.

For the Observation pattern (minus the sub-super classification relationships), we could have the following as a minimal application:

> We are to maintain information about a set of patient and their medical observations.
>
> Each patient is identified by a name.
>
> An observation can be: (1) a measurement observation (e.g. the height of a patient); (2) a category observation (e.g. a person blood group); or a presence / absence observation which records the presence or absence of certain property.
>
> Each observation type has a name (e.g. "height", "blood group"). A measurement observation type has a unit associated with it and a range of valid values. Each unit is identified by a name. each category observation type has a number of associated

categories. Each category is identified by a name.

Operations to be performed on the system include: (1) add / delete a patient; (2) add an observation type; (3) add an observation for a patient; etc.

The minimal application provides a well-defined context to demonstrate how the pattern actually works. Note that to provide the context, we have to make some rather arbitrary assumptions (for example, patients are identified by names, each measurement observation type is associated with only one unit, etc.). The minimal application is useful in showing: how the objects are introduced into the system, how they can be updated, how they are related to each other and how the integrity constraints can be enforced.

We can proceed to specify the minimal application formally and/or prototype it. We use Object-Z ([Duke 95], [Lano 94]) and Smalltalk for formal specification and prototyping (see, for example, [Nguyen 97]). We have found this approach to be very useful for studying and presenting analysis patterns.

8 Summary

In developing the Patient History System, we have started with, and then modified, the Observation pattern of Fowler. So far, the alternative pattern has been used only in our project. In the context of that application, the alternative solution satisfies the requirements quite well. In addition, it appears to be natural, easy to understand and to use. We also point out how the two models (Fowler's and ours) can be related to each other.

Finally, we briefly describe our approach to study (and present) analysis patterns. We have found that approach, which makes full use of formal notation and Smalltalk, to be very effective.

Acknowledgment: We express our thanks to Dr. Don Watson, Victoria University of Technology, Melbourne, Australia, for making available the PHS's initial requirements statement which is partly reproduced in section 4.

References

[Duke 95] Duke R., Rose G., Smith G. (1995) Object-Z: a Specification Language Advocated for the Description of Standards, to appear in *Computer Systems and Interfaces*.

[Fowler 96] Fowler M. (1996) *Analysis Patterns and Business Objects*, http://www.tiac.net/user/jsuth/oopsla96/fowler.html.

[Fowler 97a] Fowler M. (1997) *Analysis Patterns: Reusable Object Models*, Addison-Wesley.

[Fowler 97b] Fowler M. (1997) *UML Distilled*, Addison-Wesley.

[Lano 94] Lano K., Haughton H., Ed (1994) *Object-Oriented Specification Case Studies*, Prentice Hall, Hemel Hempstead, UK.

[Nguyen 97] Nguyen K., Dillon T. (1997) A Practical Formally-Based Modelling Method for Object oriented information systems, *1997 International conference on Object Oriented Information Systems*, Springer Verlag, London, G.B.