Business Rule Solutions

Ronald G. Ross
Principal, Business Rule Solutions, LLC
Executive Editor, Business Rules Journal
www.BRCommunity.com
Author: *Business Rule Concepts: Getting to the Point of Knowledge*

Rross@brsolutions.com

Four Very Useful Constructs for Concept Models: *Developing a Structured Business Vocabulary*

A missing ingredient in most current approaches to business process improvement and IT requirements is developing a standard business vocabulary, a concept model. Every BPM professional should be familiar with the technique – it's simply about clear thinking and unambiguous communication. What are basic constructs in developing a concept model? This Column discusses four prefabricated elements of structure, ones that will enable you to build a complete and robust business vocabulary.

Got functional silos across your business processes? It's worse than you think – you have *semantic* silos. How will you ever operate effectively in a knowledge economy if you're effectively living in a Tower of Babel?!

The remedy is a concept model. A concept model is a *structured business vocabulary*, the set of terms and their definitions, along with special wordings, that organize operational business know-how. Think of a concept model as a *semantic blueprint* for supporting highly complex business communication (such as business requirements and business rules).

Certain elements of structure for concept models come in handy, pre-defined shapes. This discussion illustrates use of four of these 'shapes', as presented in Table 1. These elements of structure are based on the standard SBVR (*Semantics of Business Vocabulary and Business Rules*). Our approach to depicting and developing them is called *ConceptSpeak*™.

About the Term *Instance* – Food for Thought

In this discussion I will take a few liberties with the SBVR concept of *instance*. In SBVR, instances are always in the real world, not in a model. For example, you can't put the real country *Canada* into a model. Wouldn't exactly fit(!). In a concept model, you can include only concepts (like the one that stands for Canada).

If you're used to thinking about instances being organized within or by a model (e.g., to be stored in a database), however, that gets a little confusing. So in this discussion, I will use the term *instance* a bit more loosely. Just remember, when business people talk about real-world things, they're *not* talking about instances in some model(!).

Table 1. Elements of Structure for Concept Models.

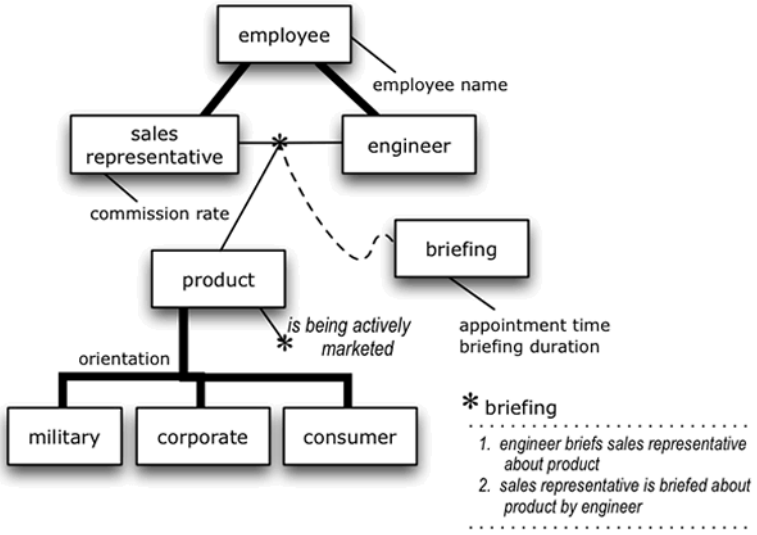
Special-Purpose Element of Structure	General Form	Example	Use in a Sample Statement
categorization	(Class of thing$_1$) *is a category of* (class of thing$_2$).	'Corporate customer' *is a category of* 'customer'.	A customer is always considered corporate if the customer is not an individual person.
property	(thing$_1$) *has* (thing$_2$)	order *has* date taken order *has* date promised	An order's date promised must be at least 24 hours after the order's date taken.
composition (whole-part or partitive structure)	(whole) *is composed of* (parts) (part) *is included in* (whole)	chair *is composed of*: • legs • seat • back • armrests	A chair may be ordered without armrests.
classification	(Instance) *is classified as a* (class of thing).	Canada *is classified as a* country. Canadian dollar *is classified as a* currency.	An order may be priced using the currency 'Canadian dollar' only if the customer placing the order is located in Canada.

Categories and Categorizations

A *category* is a class of things whose meaning is more restrictive than, but otherwise compliant with, some other class of things. For example, *male* is a category of *person*. Each male is always a person, but not every person is a male. A male can have properties that would not apply to any person who is not a male. In general, a category represents a kind, or variation, within a more general concept.

Representing one class of things to be a category of another class of things is called *categorization.* Figure 1 illustrates several categories using the ConceptSpeak convention of heavy lines.

Figure 1. Illustration of categories.



The following *categorizations* are illustrated by Figure 1.

- Both *sales representative* and *engineer* are recognized as categories of a more general concept *employee*. Note the property *employee name* is indicated for *employee*. Since all sales representatives and engineers can have names — indeed, *any* employee can — the *name* property is indicated only for *employee.* Remember that all sales representatives and engineers *are* employees in this business, so the *name* property pertains as a matter of course to both *representative* and *engineer*. It does not need to be re-specified for them; applicability (inheritance) of the property is assumed. On the other hand, commission rates apparently pertain only to sales representatives — not to all employees (e.g., not to engineers) — since *commission rate* is indicated only for *sale representative*.
- *Product* has three categories — *military*, *corporate*, and *consumer* — forming a group. This group of categories is organized on the basis of a categorization scheme named *orientation* — more about that later. Note that (as always for categories) *military*, *corporate,* and *consumer* must be products. Indeed, unless everyone reading the ConceptSpeak diagram is thoroughly familiar with categorization, better labels would probably be *military product*, *corporate product*, and *consumer product*. The boxes represent that anyway, but these revised labels would emphasize the point.

Any category can have categories; any category of a category can have categories, and so on. Multiple levels of categorization are not uncommon in concept models. Indeed, such refinement or narrowing of meaning as you go 'deeper' yields a high degree of precision or selectivity for making statements about the business (e.g., expressing business rules). For example, a business rule might be expressed for software engineers, a potential category of *engineer*, which does not apply either to other kinds of engineers or to employees in general.

Properties

Merriam-Webster Unabridged defines *property* as *a quality or trait belonging to a person or thing.* Figure 1 indicates *employee name* to be a property of *employee*, and *commission rate* to be a property of *sales representative*. In ConceptSpeak a thin line is used to attach each to the appropriate box (noun concept). Exactly what does the thin line represent?

- The thin line does *not* indicate that every member of a class of things actually has an instance of the property, only that it *can*. If each member of a class of things must have an instance of the property, an explicit business rule is required (e.g., *An employee must have an employee name.*).

- The thin line is actually shorthand for a *binary verb concept*. The *wording* for this binary verb concept defaults to *(thing$_1$) has (thing$_2$)*. The important word here is *has*. The verb *to have* is very general — not specific or descriptive at all. *Has* makes very poor wording for verb concepts not specified as properties. For properties, on the other hand, a *has* default is often convenient.

Can properties be worded using verbs other than *has*? *Yes.* For example, the *commission rate* property of *sales representative* might be worded *sales representative is compensated at commission rate*.

The property shown at the end of the line is often actually a role of some other noun concept. For example:

- Suppose commission rates are always percentages (in this business). Then the *commission rate* property of *sales representative* actually represents the verb concept worded *sales representative is compensated at [commission rate] percentage*.
- Similarly, the *employee name* property of employee might actually represent the verb concept worded *employee has [employee name] name*.

Note on ConceptSpeak Notation

Why bother with a graphical shorthand for properties? The answer has to do with scaling up. If you were to treat all properties as 'regular' verb concepts, the concept model would become hopelessly cluttered with connections having to do with such things as numbers, names, dates, units of measure, and much more. Such connections are of secondary importance to the business. Avoid that!

Figure 1 actually includes several other properties, as follows.

- Two properties for the objectification *briefing* have been indicated using a *single* thin line –another ConceptSpeak shorthand to reduce clutter.
- *Orientation*, which can be seen just above the crossbar for the categorization of *product,* is also a property, albeit a special kind. *Orientation* is the name of the *categorization scheme* used to organize the three kinds of product. Since *orientation* is a property of *product*; we can say *product has orientation*. (That's like saying *person has gender*, meaning *male* and *female*.) Is it required that every product fall into at least one of the three categories: *military*, *government*, or *consumer*? In other words, must every product have an orientation? (Or perhaps *exactly* one?) Never assume so — that would require some explicit business rule(s).

Compositions — Whole-Part (Partitive) Structures

Many things in the real world are composites, made up of several other kinds of thing. For example, an automobile (simplistically) is composed of an engine, a body, and wheels. A mechanical pencil is made up of a barrel, a lead-advance mechanism, pencil lead, and eraser. An address (simplistically) is made up of a street number, a street, an apartment number, a city, a state/province, a country, and a zip code / postal code.

Sorting out the terminology and composition of such *whole-part structures* is often quite useful. Before looking at a graphical example, let's address some relevant questions:

- Is every instance of the whole in a whole-part structure required to have at least one instance of each part? *No.* For example, not every address has an apartment number. If every instance of the whole *is* required to have some part(s), an explicit business rule must be given.
- Can an instance of a whole have more than one instance of a kind of part? *Yes.* An automobile must have at least three wheels (a business rule). But use caution here. A

whole-part structure usually works best where there is only one, or a small number of, each part.

- Can the specification of a whole-part structure indicate only one kind of part? *Yes.* However, exercise common sense! For example, is it really useful to consider the verb concept worded *order includes line item* to be a whole-part structure? ConceptSpeak does not favor that practice.
- Can a part itself be a whole composed of other parts? *Yes.* Multiple levels of composition are possible.
- Can both the whole and the parts be selectively involved in verb concepts on their own? *Yes*.
- Can an instance of a part exist independently from an instance of the whole? *Yes* (unless business rules disallow it). A wheel, for example, can be removed from an automobile.
- Can an instance of a part be in more than one instance of a whole at the same time? *Yes* (again, unless business rules disallow it). A power source, for example, can be part of more than one circuit.

Figure 2 illustrates a composition of *briefing* using the ConceptSpeak convention of a tree structure of thin lines to indicate the parts. The wording for this verb concept, not shown explicitly, is assumed to be: *briefing is composed of: introduction, main body, conclusion*. (Or, as they sometimes say in the military, *tell 'em what you're gonna to tell 'em, tell 'em, and tell 'em what you told 'em*.)

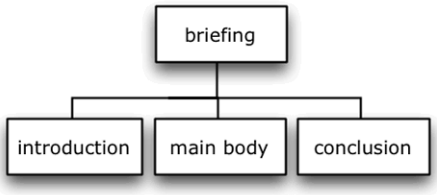


Figure 2. Example of a composition (whole-part structure).

Classifications

A central focus in concept modeling is on identifying, defining, and naming the classes of things important to basic business operations. Most often the business cannot possibly know in advance what all the instances will be of a class of things. For example, most businesses cannot predict all their future customers.

For certain classes of things, however, the business can identify or prescribe in advance some or all of the instances, especially for those classes where the instances are relatively stable. For example, we know all the European countries at the present time. Moreover, the business will need to pre-define instances when it has some business rule(s) that pertain selectively to them — for example: *A shipment may be made only to the European countries United Kingdom or The Netherlands.*

Representing the connection between an instance and its particular class of things is called *classification*. Figure 3 illustrates. In ConceptSpeak a line with the double-wavy hatch mark indicates a classification connection from the class of things *European country* to some of its instances.

Note on ConceptSpeak Notation

The double-wavy hatch mark indicates that a meta level is crossed. To avoid clutter, we recommend ample use of neighborhoods to depict instance-level terminology.

Figure 3. Example of classification.

Some additional examples of classifications:

- *Health care:* All recognized health services — e.g., *consultation*, *office visit*, *hospital admission*, *surgery*, and so on.
- *Ship inspection:* All recognized parts of a ship — e.g., *bulkhead*, *hatch cover*, *railing*, *deck*, and so on.

These examples were chosen deliberately to illustrate that classifications can be multi-level. For example, the instances *bulkhead, hatch cover*, etc. of the class of things *ship part type* might themselves be viewed as classes of things with respect to *specific* bulkheads, hatch covers, etc. These specific bulkheads, hatch covers, etc. probably have serial numbers and would be found on a given ship or in a given shipyard. Business rules might be targeted toward any of these levels.

Summary

Certain elements of structure useful for concept modeling come in handy, pre-defined 'shapes'. This discussion has illustrated four of these special-purpose elements of structure: *properties, categorizations, compositions,* and *classifications*. These special connections between noun concepts extend the reach and precision of the concept model significantly. They also allow statements to be written with great precision — for example, giving business rules or writing very precise requirements. In the final analysis, it's simply all about clear thinking and unambiguous communication.

JUST RELEASED!

Excerpted from Chapter 6, Business Rule Concepts: *Getting to the Point of Knowledge* (4th ed.), by Ronald G. Ross, April, 2013. ISBN 0-941049-14-0
http://www.brsolutions.com/b_concepts.php

BPTrends Linkedin Discussion Group
We created a BPTrends Discussion Group on Linkedin to allow our members, readers and friends to freely exchange ideas on a wide variety of BPM related topics. We encourage you to initiate a new discussion on this publication, or on other BPM related topics of interest to you, or to contribute to existing discussions. Go to Linkedin and join the **BPTrends Discussion Group.**