Date: October 2019

Semantics of Business Vocabulary and Business Rules (SBVR), v1.5

Annex M - A Conceptual Overview of SBVR and the NIAM2007 Procedure to Specify a Conceptual Schema

---

OMG Document Number: formal/19-10-12

Note: This SBVR Annex is published as a separate document solely for convenience and ease of use. The fact that it is published as a separate SBVR specification document makes no change to its status as part of the SBVR specification, or the way in which it can be updated under OMG Policies and Procedures.

Annex M - A Conceptual Overview of SBVR and the NIAM2007 Procedure to Specify a Conceptual Schema

(informative)

M.1 Introduction

The acceptance of SBVR is a breakthrough in productivity in requirements and knowledge management. It is fundamentally a fact oriented approach, which makes it comprehensible to many people. It so happens that experience with this approach started in Europe in the seventies, and a mature business practice has been developed during the last 35 years. The current version of this practice is called NIAM2007 [Nijs1977, Nijs1978, Nijs1980, Nijs1986, Nijs2006].

In this annex we will primarily concentrate on describing the coherence of the essential concepts of SBVR, using the NIAM2007 methodology, and thus providing the reader with an easy to grasp framework for SBVR. NIAM2007 uses fact type diagrams that combine the advantages of diagrams and natural language statements, by integrating diagrammatic and natural language aspects.

A small part of the EU-Rent example of Annex G will be used to build up, step by step, an understanding of a well-selected subset of the SBVR core concepts and how they interrelate (i.e., their coherence).

For communication purposes we start with a concrete example, which, in the framework of Figure L.1 is at the level called 'Fact population.' From there we move systematically via the domain-specific component of the conceptual schema to the generic component. This is another useful direction compared to clauses 8 through 21 and 24 and especially appreciated by people new to the subject.

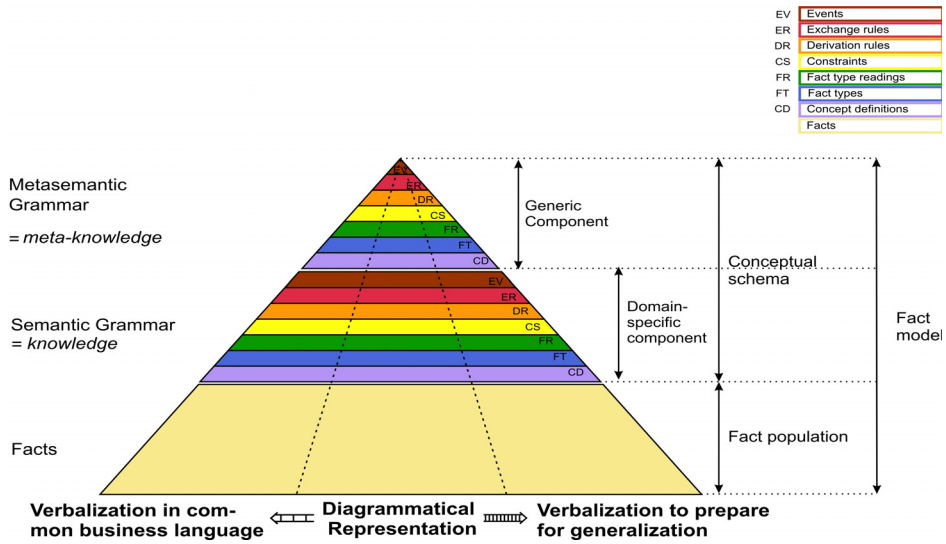

Figure M.1 - Knowledge triangle

The knowledge triangle of Figure M.1 represents the core concepts described in sub clause 10.2.1.2 and a few other concepts, specific for NIAM2007, in diagrammatical coherence.

The knowledge triangle is divided into three vertical lanes. The middle lane represents diagrammatical representations of structured knowledge. The left lane represents verbalizations for business people, which represent the same knowledge in a way familiar to business people. The right lane represents the NIAM2007 ‘verbalizations to prepare for generalization,’ which represents the same knowledge as perceived from the perspective to derive the next level. The latter form of verbalization allows for generalization, which is a step in the procedure of deriving the next level. This will be shown in detail in the following sections.

The knowledge triangle is divided into three levels of knowledge or facts:

I. Facts: facts without a grammatical function, called ground facts in sub clause 24.2.1.2, e.g., ‘The operating country Germany uses the business currency Euro,’ ‘The operating country France uses the business currency Euro,’ and ‘The operating country USA uses the business currency USD.’

II. Semantic Grammars: facts with a domain-specific grammar function, called the domain-specific component of the conceptual schema in sub clause 24.2.1.2, e.g., ‘For each country that is recorded, its currency must also be recorded’ and ‘Each country name recorded in this fact type has to be unique (i.e., the only occurrence of that name).’

III. Metasemantic Grammar: facts with a generic or meta grammar function, called the generic component of the conceptual schema in sub clause 24.2.1.2, e.g., ‘Each fact type must have a role or a sequence of roles for which uniqueness is required.’

In the following sub clauses of this annex, understanding of these concepts will be built up step by step. It will be shown that level II contains the rules and concept definitions for ground facts, and that level III contains meta-rules i.e., rules for rules, including the meta-rules themselves as well as the relevant concept definitions. Thus, level III describes itself. Therefore, these three levels suffice for describing knowledge.

The triangle was chosen as the form to represent structured knowledge to show that there are always more ground facts than rules for them and more level II (domain-specific) rules than meta-rules. This is the intent of defining rules: rules about knowledge are made to make working with knowledge more productive.

In the knowledge triangle the domain-specific as well as the generic component of the conceptual schema are divided in seven related knowledge classes:

- Concept definitions
- Fact types
- Fact type readings (also known as sentential forms)
- Constraints
- Derivation rules
- Exchange rules
- Events

These knowledge classes are part of SBVR as well as NIAM2007, except for exchange rules and events, which are not part of SBVR. Why are they in the knowledge triangle? To facilitate respectful discussions with other communities, such as UML. In the following sub clauses of this annex, all of these knowledge classes will be explained, except exchange rules and events, which fall outside of the scope of this annex. Of course, the concept of 'fact' will also be thoroughly explained.

SBVR is a major step forward towards widespread application of semantics in business and education. SBVR is the first specification in business computing where concept definitions are first class citizens. The concept definitions form the bridge between the formal and the informal world, hence are vital for business communication. One of the 7 knowledge classes at the domain-specific and the generic level, Concept Definitions, form the basis for each of the conceptual schemas, the domain-specific component and the generic component. As various annexes put the major emphasis on rules, this annex puts major emphasis on concept definitions, fact types, and a useful variant of verbalization.

M.2 Use Case EU-Rent 1.1

A substantially reduced version of the EU-Rent Use case presented in Annex G is given below:

EU-Rent Use case 1.1

1. EU-Rent rents cars to its customers. Customers may be individuals or companies. It is obligatory
2. that the rental charge of a rental is calculated in the business currency of the rental.
3. this is a currency in which EU-Rent undertakes financial transactions. A rental has a business
4. currency, if and only if the business currency is the currency of the operating country of the
5. operating company that includes the local area that includes the pick-up branch of the rental.
6. The used business currencies are Euro (EUR), GBP (British Pound) and USD (United States
7. Dollar). Every country only uses one business currency.
8.
9. In each country in which it does business EU-Rent has an Operating Company. EU-Rent's current
10. operating countries are Canada, USA, France, UK, Ireland, Germany, Italy and Switzerland.

Regarding this use case, we first wish to focus on the domain-specific component of the conceptual schema and from there on a core part of the generic component of the conceptual schema to illustrate main concepts of SBVR. First of all, a sample graphical report has to be made regarding the different operating countries of EU-Rent and their respective currencies.

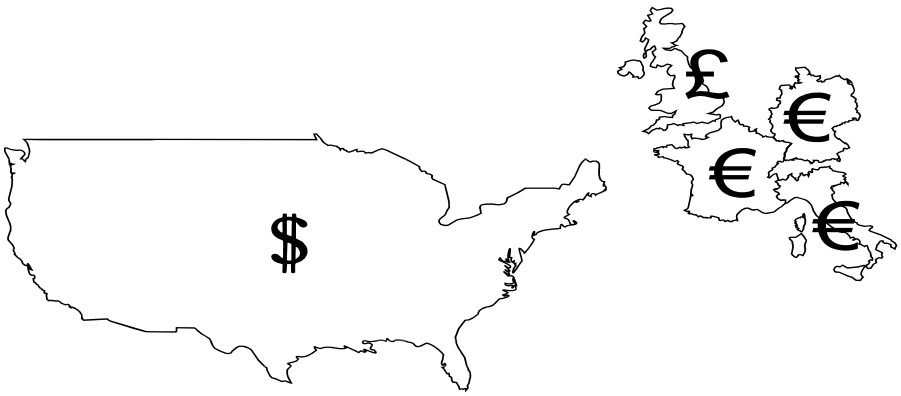

Figure M.2 - Graphical representation of operating countries and their business currency

Figure M.2 is a graphical representation of facts illustrating the use case described above. It is a sample report and could be called a data use case. Since the diagram represents actual facts satisfying the data use case, it is possible to verbalize the contents as if the business professional is talking to a colleague over the phone and writes down the textual representation of the represented facts. This is what is called “to verbalize a graphical representation.” Extensive experience with fact orientation during four decades has shown that starting with a concrete example of a data use case represented in the preferred notation of the business user is *the most productive* way to start requirements engineering and illustrate business processes. It is also recommended for knowledge elicitation.

If we ask a subject matter expert or business person to verbalize the given information of Figure M.2, the following sentences or facts will result:

Table M.1 - The result of verbalization by the domain expert

Operating country	Germany	uses the business currency	Euro
Operating country	UK	uses the business currency	GBP
Operating country	France	uses the business currency	Euro
Operating country	USA	uses the business currency	USD
Operating country	Italy	uses the business currency	Euro

As a first step towards a structured specification or conceptual schema, for every fact or sentence it is indicated where the variable and where the constant sentence elements are located. The result of this operation is presented in Table M.2.

Table M.2 - The result of assigning constant and variable parts

Operating country	Germany	uses the business currency	Euro
” ”	UK	” ” ” ”	GBP
” ”	France	” ” ” ”	Euro
” ”	USA	” ” ” ”	USD
” ”	Italy	” ” ” ”	Euro

As can be seen from the table above, there are two constant elements in each sentence, i.e., elements that are the same in each sentence (in this case “Operating country” and “uses the business currency,” respectively). There are two variable elements in each sentence, i.e., elements that have potentially different counterparts in the other sentences. The first variable element of each sentence is an example of an operating country and the second variable element of each sentence is an example of a business currency.

Like professionals in many other professions, extensive use is made of pattern recognition in the NIAM2007 methodology. From which fact type reading are the five listed facts an instance, an instantiation or a realization? We can conclude that we are dealing with sentences or facts that can be generated from the same fact type reading by filling in a value in two places, while the other elements consist of the same information for every sentence. The places in the fact type reading where the variable elements are to be filled in to form sentences, are called the ‘placeholders’ of the fact type reading in SBVR. By formulating a fact type reading it becomes possible to communicate the contents of a diagrammatical or report representation in a manner suited to a specific audience. The fact type reading which can be formulated based on the sentences listed in Tables M.1 and M.2, is given below and is assigned the number 1.

1: Operating country <Country> uses the business currency <Currency>

This fact type reading has been derived by generalization of five example sentences, or facts. By filling in the placeholder <Country> with the name of an actual operating country (e.g., “Germany”), and the placeholder <Currency> with the name of a business currency (e.g., “Euro”), we obtain a concrete sentence or fact, in this case one of the sentences we started with.

Each placeholder has a counterpart in a fact type, and this counterpart is called ‘role’ in SBVR. This counterpart is shown in Figure M.3 in a diagrammatical form, using a NIAM2007 representation. In the diagrammatical representation of a fact type i.e., the fact type diagram, a role is represented by a rectangle containing the name of the role. This diagram also contains the

fact type reading. In such diagrams, it is advised to include a sample population. In this case, five different pairs of variable elements are filled into the pair of roles, as population of the fact type.

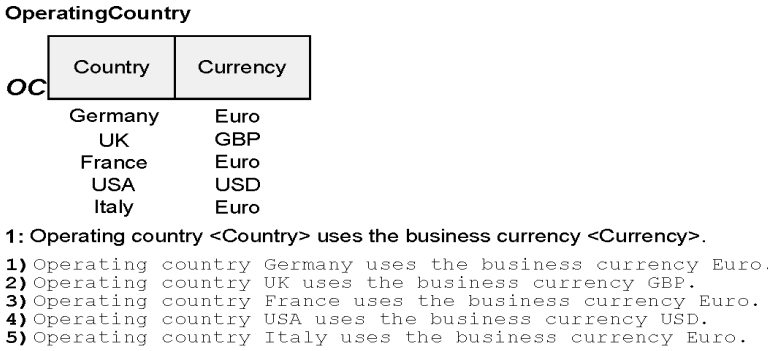

Figure M.3 - Provisional fact type diagram with population

Every fact instance verbalization is given a unique number followed by the symbol ‘)’. The fact instance verbalizations are generated, based on the values in the fact population of the fact type diagram OperatingCountry in Figure M.3.

Every fact type reading is given a unique number or code, in this case the number 1, within the domain-specific component of the conceptual schema. Every fact type is given a name, in this case OperatingCountry as well as a shorter code (here: OC) to facilitate communication.

The fact to be generated from the first record below the roles in the fact type diagram (which is the first record of the population) and the fact type reading can be read as follows: Operating country Germany uses the business currency Euro.

Regarding the structural understanding of the world (or the semantics) of these kinds of fact examples, at least the following terms have to be defined as concept definitions in this domain-specific conceptual schema:

Business currency
{Business currency} is a monetary entity in which EU-Rent undertakes financial transactions.

Operating country
{Operating country} is a country in which EU-Rent does business.

Above, we used a concrete graphical example in which the relevant facts are represented in a diagrammatical manner. We verbalized these diagrammatical representations of facts to get textual representations of the same facts. We made a start in transforming each textual representation into a domain-specific conceptual schema. Until now this transformation has resulted in:

1. two concept definitions;
2. one fact type diagram, as a possible representation of the fact type;
3. one fact type reading.

These three knowledge classes are only a part, although a very important basis, of the desired domain-specific conceptual schema. We therefore have to continue specifying the additional parts of the conceptual schema. We proceed in a structured way to the next part of the conceptual schema, the so-called constraints, a class of business rules.

What is a constraint? A constraint is a rule that limits the populations of the fact types and its population transitions, allowing only populations and transitions considered useful. According to NIAM2007, the most important constraint is the uniqueness

constraint, which is illustrated in the following section. A uniqueness constraint corresponds to the set of independent variables of a function, a major concept in mathematics.

M.3 Uniqueness Constraint

To derive constraints, it is advised to use a precise process for systematic specification. As uniqueness constraints are the major constraints, we first derive these. The precise process ensures that all questions that need to be posed to a business domain expert are systematically composed and expressed in the familiar jargon of the business professional. The result of those processes leads to the following question to the subject matter expert in a language readily understood by the business domain expert:

Is it possible that the following two sentences can exist at the same time in the fact population?

Operating country Germany uses the business currency Euro.

and

Operating country Germany uses the business currency USD.

Or, as recommended by NIAM2007, are the contents of Figure M.4 acceptable to you?

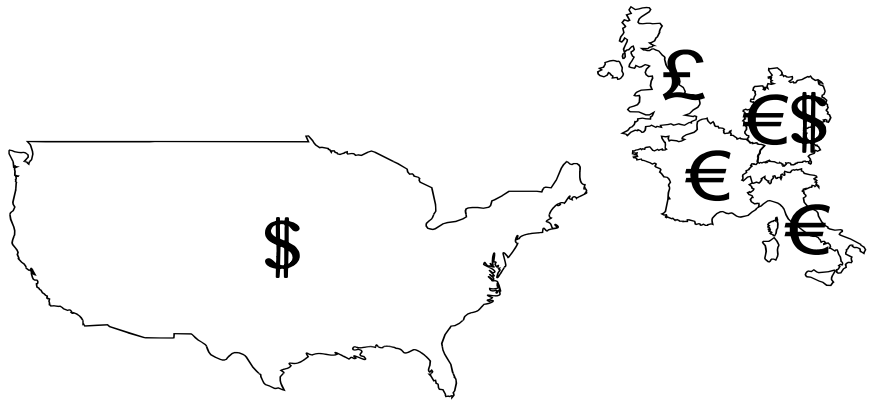

Figure M.4 - Concrete not permitted business example

The business domain expert will clearly say "No." It is not allowed for an operating country to use two or more different business currencies as specified in line 7 of EU-Rent Use case 1.1. This answer is shown diagrammatically in the fact type diagram in Figure M.5.

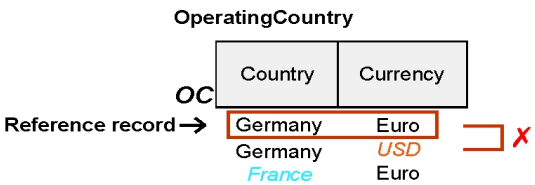

Figure M.5 - Matrix method for uniqueness: forbidden combination of records

Based on this particular answer from the business domain expert it is possible to conclude that the name of an operating country can only appear once in the fact population below the role Country of fact type diagram OperatingCountry. This

results in the situation that the name of a country is unique within this fact population. In a fact type diagram in NIAM2007 a uniqueness constraint is indicated by a solid line with an arrow at both ends. In Figure M.6, the uniqueness constraint, arbitrarily named pk23 as an indication of primary key, is added to the fact type diagram OperatingCountry.

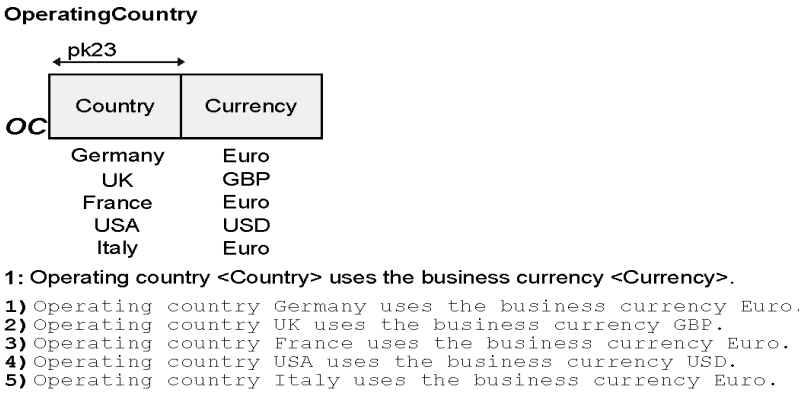

Figure M.6 - Fact type diagram OperatingCountry, after addition of uniqueness constraint pk23

What are the operational semantics of a uniqueness constraint? Every uniqueness constraint arrow means: below me in the fact population no duplicate values or signifiers can occur.

To know if a uniqueness constraint holds for the second role named “Currency,” one has to ask the business domain expert whether or not the following two facts can appear at the same time in the fact population:

Operating country Germany uses the business currency Euro.

and

Operating country France uses the business currency Euro.

The business domain expert will say “Yes, this was already clear in Figure M.2.” It is indeed possible that France as well as Germany use the same business currency; please note that this was represented in the data use case of Figure M.2. So the use of a specific business currency is not unique in this fact population. This implies that the values under the role “Currency” are not unique in this fact population and therefore no uniqueness constraint applies to this particular role of the fact type.

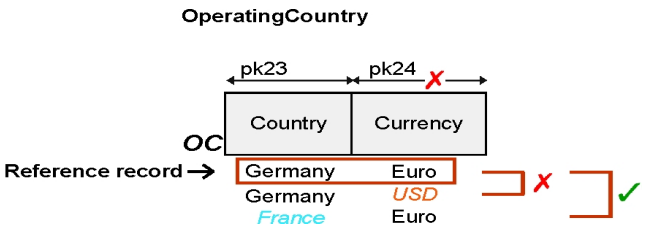

Figure M.7 - Intermediate result: no uniqueness constraint on Currency role

In Table M.3 an overview diagram of the procedure mentioned above is given.

Table M.3 - The register of answers given by the domain-specific expert

Country	Currency	Business domain expert's answer regarding the simultaneous existence with the first record	Explanation
Germany	Euro		Record 1 (Reference record)
Germany	***USD***	No	It is not allowed that an operating country uses more than one different business currency.
France	Euro	Yes	It is possible that France as well as Germany both use the same business currency.

The use of easily recognizable symbols, like traffic-signs, makes communication about a conceptual schema more productive. In addition to the uniqueness constraint symbol introduced above, Figure M.8 introduces the following symbols:

- value rule symbol (curly brackets {} in a circle),
- data type symbol (a character or character combination in the left upper corner of a role's rectangle), and
- the non-empty rule symbol (a black rectangle in the right lower corner of a role's rectangle).

A value rule limits the values that can be used to fill in a particular role to a given list of possible values (listed at the bottom of the fact type diagram for representational purposes). In our use case we know that there is a limited number of operating countries for EU-Rent: Canada, USA, France, UK, Ireland, Germany, Italy, and Switzerland. Others do not exist, thus should not be recorded, which is prescribed by value rule val5003. A data type limits the possible values of a particular role to values of a specified type. In the OperatingCountry fact type in Figure M.8, all values are allowed by the data type 'c' which is an abbreviation for 'character'. A non-empty rule forbids a role to be left empty in a record. In the fact type OperatingCountry this is true for both roles, as it is not allowed to record an operating country without recording the corresponding business currency and vice versa. The latter is implied by the uniqueness constraint pk23 and the former is indicated by the non-empty rule symbol in the Currency role.

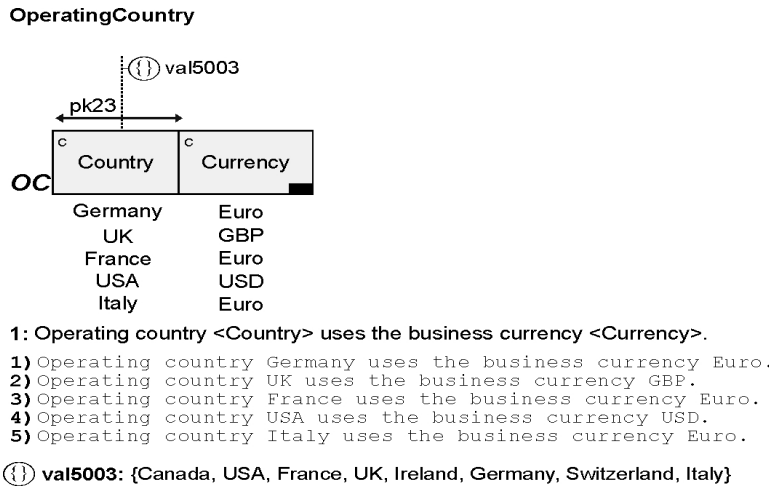

Figure M.8 - Complete fact type diagram OperatingCountry

M.4 LA Route to Time Invariant Knowledge

The previous sub clauses illustrated that we are able to verbalize the facts depicted in Figure M.2. From these verbalizations it was possible to derive fact type readings, placeholders, fact types, roles, and some constraints (based on the questions systematically posed to the business domain expert). The business domain expert is proficient in answering questions in his own language or business vocabulary ("language that is readily understood by the business domain experts" sub clause 24.2.1.2) in terms of permitted or not permitted concrete examples, a familiar world to the user. In other words, this was a trip from:

1. the level of ground facts (middle part of level I in Figure M.15), using verbalization to arrive at
2. the textual representation of the facts (right part of level I), then applying generalization to arrive at
3. the level II (middle part) of the domain-specific component of the conceptual schema in 24.2.1.2.

The next interesting question is: is it possible to verbalize these resulting diagrams of the conceptual schema with the aim to arrive at the next level? Let us apply verbalization to the fact type reading. Hence we treat the fact type reading in the middle part of level II in the same way we treated the middle part of level I. For a fruitful discussion we first provide a concept definition of 'position.'

Position

A {Position} in a fact type reading may either consist of a constant piece of text until the first placeholder, or a constant piece of text between two placeholders, or a constant piece of text at the end of the fact type reading behind the last placeholder, or a position in a fact type reading is taken by an individual placeholder.

If we use this concept definition to analyze fact type reading 1, we get four positions, which are indicated below in the fact type reading by representing every character in the consecutive positions by the corresponding position number within the fact type reading:

```
1: Operating country    <Country>   uses the business currency   <Currency>
   11111111111111111    222222222   33333333333333333333333333   4444444444
```

Based on this we are able to give the verbalizations of a part (in this case a fact type reading) of the diagrammatical representation of the domain-specific component of the conceptual schema, seeTable M.4.

Table M.4 - Verbalization of a part of the domain-specific conceptual schema

Fact type reading	1 has in position 1 A constant with contents Operating country
Fact type reading	1 has in position 2 A variable with contents Country
Fact type reading	1 has in position 3 A constant with contents uses the business currency
Fact type reading	1 has in position 4 A variable with contents Currency

When we replace the constant sentence parts in every sentence after the first sentence in Table M.4 by quotation marks, the result is as shown in Table M.5:

Table M.5 - The result of assigning constant and variable parts

Fact type reading	1	has	in	position	1	a	constant	with contents	Operating country
″ ″ ″	1	″	″	″	2	″	variable	″ ″	Country
″ ″ ″	1	″	″	″	3	″	constant	″ ″	uses the business currency
″ ″ ″	1	″	″	″	4	″	variable	″ ″	Currency

From the example above it is straightforward to derive the fact type reading. It is decided to assign number 1000 to this new fact type reading:

1000: Fact type reading <FactTypeReading> has in position <Position> a <ConstOrVar> with contents <Contents>.

The resulting fact type diagram is given in Figure M.9.

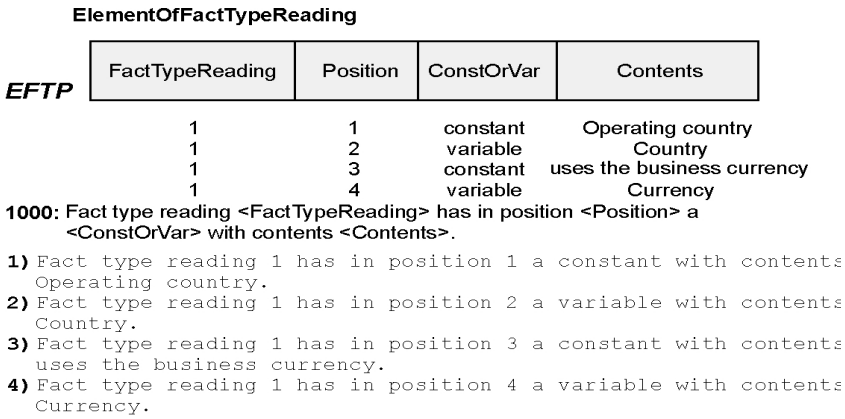

Figure M.9 - Meta fact type with population of domain-specific schema elements

Fact type reading 1000 is a rule that applies to all fact type readings, irrespective of the domain specifics. In other words, we have now arrived at the next level, the generic component of the conceptual schema. In NIAM2007 this is called the conceptual meta schema or Metasemantic Grammar (see Figure M.1). This component is topic independent.

Regarding the structural understanding of the world of these kinds of examples, at least the following concept definitions have to be provided:

ConstOrVar

{ConstOrVar} indicates whether or not an element in a [Fact type] or [Fact type reading] is variable or constant. In other words, whether or not an element in a [Fact type] or [Fact type reading] remains fixed or can be used to indicate a [Role] or [Placeholder] respectively.

Fact (Facts)

A {Fact} is a proposition that is taken as true.

Examples of facts are:

Within the class of all Member States of the United Nations the name	Austria	identifies a specific member state.
Within the class of all Member States of the United Nations the name	Belgium	identifies a specific member state.
Within the class of all Member States of the United Nations the name	Canada	identifies a specific member state.
Within the class of all Member States of the United Nations the name	The Netherlands	identifies a specific member state.
Within the class of all Member States of the United Nations the name	United States of America	identifies a specific member state.

These are five examples of unary, existential facts.

Existential facts need not be unary.

An example of a ternary existential fact is:
Within the class of all telephones in the EU the combination of country code 31, area code 45 and local number 5600222 identifies a specific telephone.

An example of an existential {Fact} as used in sub clause 24.2.1.2 is as follows:
There is a country that has the country code 'US'

Based upon a long experience in industrial fact oriented modeling we recommend to use the formulation:
Within the class of all <ConceptPlural> the <Signifier> identifies a specific <ConceptSingular>

An example of a unary non-existential fact is:
Bill Clinton smokes

Five examples of binary facts are:

Amsterdam	is the capital of	The Netherlands
Brussels	is the capital of	Belgium
Ottawa	is the capital of	Canada
Washington	is the capital of	United States of America
Vienna	is the capital of	Austria

Five examples of ternary facts are:

The Netherlands	entered the	EU	in	1957
The Netherlands	entered the	NATO	in	1949
United States of America	entered the	NATO	in	1949
United States of America	entered the	NAFTA	in	1989
Canada	entered the	NAFTA	in	1989

Fact type (Fact types)

A {Fact type} is a structure that enables recording of [Variable elements] of [Facts] that can be verbalized within a subject.

Fact type reading

A {Fact type reading} is a mould that belongs to a [Fact type] that consists of constant parts of a [Fact] and [Placeholders], with which the [Population] of a [Fact type] can be displayed in understandable sentences.

Placeholder (Placeholders)

A {Placeholder} is a part of a [Fact type]; each {Placeholder} in a [Fact type reading] has a corresponding [Role] in a [Fact type].

Population

A {Population} is a set of all [Variable elements] of a [Fact] that are being recorded in the [Role] of a [Fact type].

Position

Position is the place of a [Variable element] or constant in a [Fact type] or [Fact type reading].

Role (Roles)

A {Role} is part of a [Fact type]. It facilitates recording of one specific [Variable element] of those [Facts], for which all [Variable elements] are being recorded by means of this [Fact type]. [Fact types] always contain one or more {Roles}.

Variable element (Variable elements)

A {Role} is part of

{Variable elements} are the varying parts within a set of distinct [Facts], where these [Facts] must have the same kind of meaning and use the same kind of phrasing.

Above, the required concept definitions are given. In Figure M.9 we have defined an intermediate fact type diagram. We will now proceed to derive the uniqueness constraint for the fact type.

The following questions have to be asked:

a. Is it possible that the following two sentences can exist at the same time in the fact population?

   Fact type reading 1 has in position 1 a constant with contents ***Operating country*** and
   Fact type reading 1 has in position 1 a constant with contents **Operating company**

b. Is it possible that the following two sentences can exist at the same time in the fact population?

   Fact type reading 1 has in position 1 a ***constant*** *with contents Operating country* and
   Fact type reading 1 has in position 1 a **variable** with contents Operating country

c. Is it possible that the following two sentences can exist at the same time in the fact population?

*Fact type reading 1 has in position **1** a constant with contents Operating country* and
Fact type reading 1 has in position **2** a constant with contents Operating country

d. Is it possible that the following two sentences can exist at the same time in the fact population?

*Fact type reading **1** has in position 1 a constant with contents Operating country* and
Fact type reading **2** has in position 1 a constant with contents Operating country

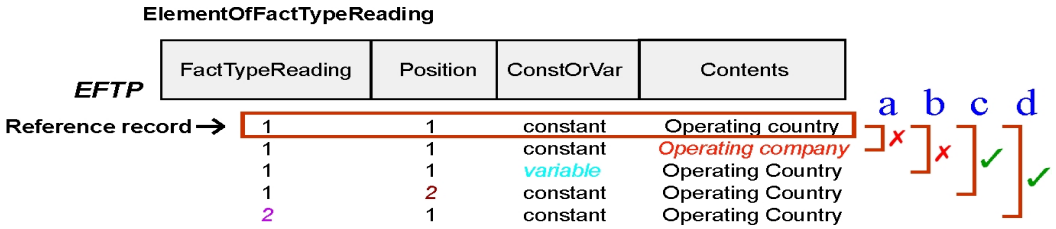

Figure M.10 - The register of answers given by the generic expert

Based on the above mentioned procedure, the following results are recorded.

Table M.6 - The register of answers and explanations given by the generic expert

Fact Type Reading	Position	ConstOrVar	Contents	Answer regarding the simultaneous existence with the first record	Explanation and notes
1	1	constant	operating country		Record 1 (Reference record)
1	1	constant	operating company	No	There can be only one kind of content in any given position.
1	1	variable	operating country	No	Every position in a fact type reading is either a constant or a variable. Additionally, we have to mention that we deal with a value rule for the role “ConstOrVar” which prescribes that only the values “constant” or “variable” can be used, see Figure M.12.
1	2	constant	operating country	Yes	An additional constraint is: two adjacent positions cannot both be of type constant.
2	1	constant	operating country	Yes	Of course

Based on these answers the analyst is able to derive the uniqueness constraint and adds it to the fact type diagram as shown in Figure M.11.

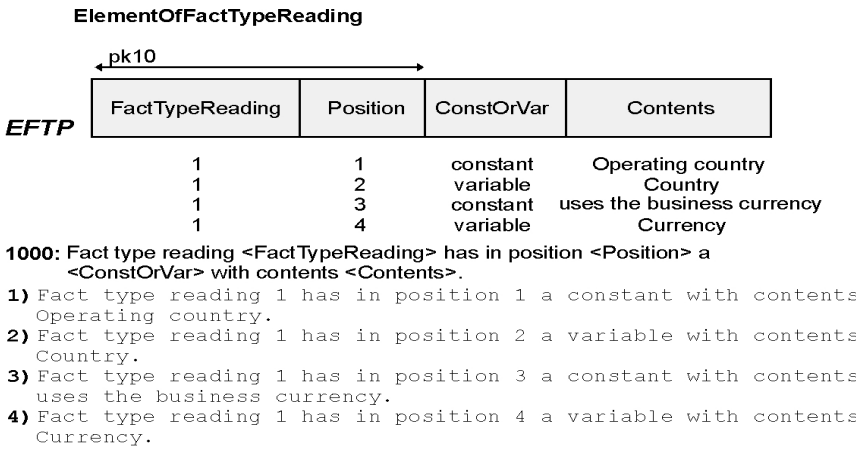

Figure M.11 - Meta fact type with population of domain-specific conceptual schema elements

Since we know that the role "ConstOrVar" can only contain the values "constant" or "variable," a value constraint is applied to limit the possible values to the ones mentioned, as can be seen in Figure M.12. In addition, the necessary mandatory roles and data types are added ('n' is an abbreviation for 'numeric').

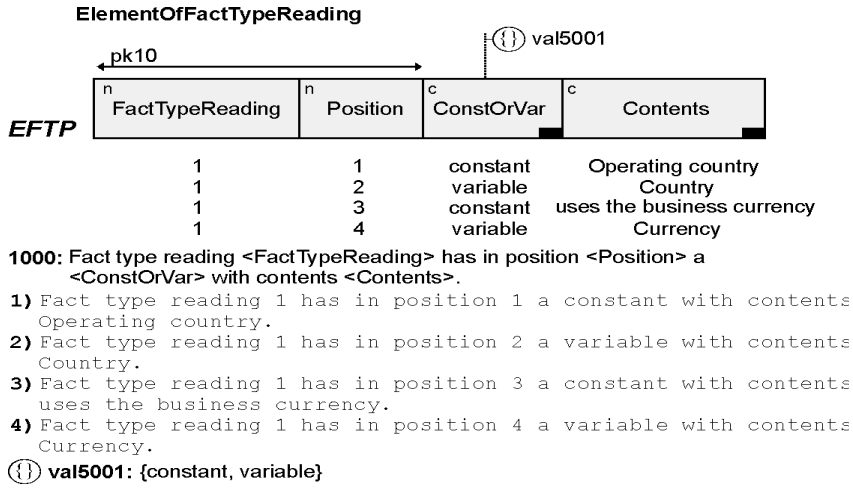

Figure M.12 - Meta fact type with population of domain-specific conceptual schema elements, mandatory role, value rule and data types added

Since we are now at the middle part of the generic component level in the knowledge triangle (i.e., level III), the question can be asked whether it is possible to verbalize (a part of) this fact type diagram. Using the methodology we used at levels I and II, we come to the result as shown in Table M.7, when we verbalize the fact type reading of fact type diagram "ElementOfFactTypeReading."

Table M.7 - Verbalization of a part of the generic conceptual schema

Fact type reading	1000	has in position	1	a	constant	with contents	Fact type reading
Fact type reading	1000	has in position	2	a	variable	with contents	FactTypeReading
Fact type reading	1000	has in position	3	a	constant	with contents	has in position
Fact type reading	1000	has in position	4	a	variable	with contents	Position
Fact type reading	1000	has in position	5	a	constant	with contents	a
Fact type reading	1000	has in position	6	a	variable	with contents	ConstOrVar
Fact type reading	1000	has in position	7	a	constant	with contents	with contents
Fact type reading	1000	has in position	8	a	variable	with contents	Contents

When the constant sentence parts are replaced by quotation marks, we come to the result shown in Table M.8.

Table M.8 - The result of assigning constant and variable parts

Fact type reading	1000	has in position	1	a	constant	with contents	Fact type reading
″ ″ ″	1000	″ ″ ″	2	″	variable	″ ″	FactTypeReading
″ ″ ″	1000	″ ″ ″	3	″	constant	″ ″	has in position
″ ″ ″	1000	″ ″ ″	4	″	variable	″ ″	Position
″ ″ ″	1000	″ ″ ″	5	″	constant	″ ″	a
″ ″ ″	1000	″ ″ ″	6	″	variable	″ ″	ConstOrVar
″ ″ ″	1000	″ ″ ″	7	″	constant	″ ″	with contents
″ ″ ″	1000	″ ″ ″	8	″	variable	″ ″	Contents

From this it is evident that there is no higher level than the generic component level and that this level actually is self-describing. Since we previously concluded that the fact type diagram ElementOfFactTypeReading is a grammar rule for all fact type readings, as a consequence, it is possible to add the information from fact type reading 1000 to this diagram as well. Hence the rules of the generic component can be presented as fact population of the generic component itself, see Figure M.13.

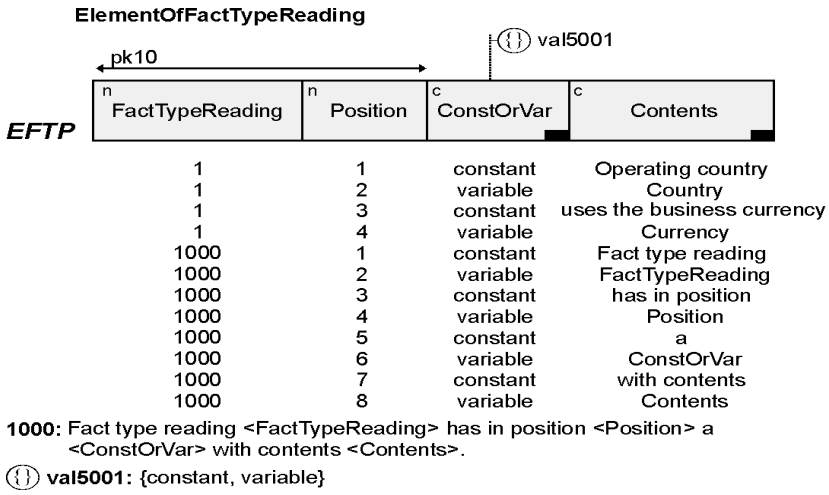

Figure M.13 - Meta fact type with population of both domain-specific and generic conceptual schema elements

We have now followed certain paths of the methodology in the knowledge triangle. In Figure M.14 a more elaborate knowledge triangle enhanced with processes is given. In this figure, the arrows (1b), (4b) and (7b) differ from the similar ones (1a), (4a) and (7a), because their function is not to use verbalization in order to get to a higher level, but to verbalize in a language business people are familiar with. Thus, no arrows of the kind of 2, 5 and 8 appear on the left side in the knowledge triangle.

M.5 Overview of structured and structuring knowledge

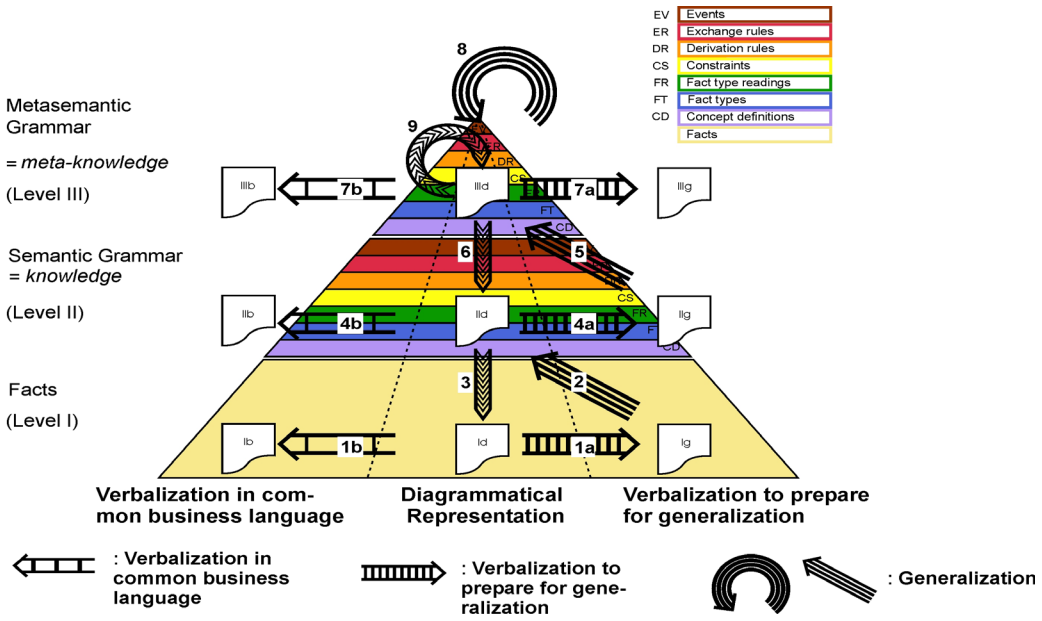

Figure M.14 - Knowledge triangle with process aspects

In conclusion, we started from the level of ground facts (level I in Figure M.14). At this level the facts have no grammatical function. By applying verbalization (1a and 1b) to the diagrammatic representation at the ground fact level (Id, i.e. level I diagrammatic representation), the results are the facts in a textual format (Ig and Ib). By applying generalization (2) to textual representation Ig at the ground facts level, the core of the domain-specific component of the conceptual schema was obtained in a diagrammatic format (IId).

This diagrammatic format of the domain-specific component of the conceptual schema (IId) - a semantic grammar - describes the meaning of terms at the ground fact level (Id) and it specifies the rules for fact populations (Id), fact population transitions (Id) and it contains the fact generation rules (IId). Hence IId determines (3) Id and describes its meaning.

Next, by applying verbalization with the aim to arrive at a higher level (4a) to the diagrammatic representation at the level of the domain-specific component of the conceptual schema (IId), we obtain a textual format of the domain-specific component (IIg).

Continuing this process, by using generalization (5) at level II, the result is a diagrammatic representation of a core part of the generic component of the conceptual schema (IIId). This diagrammatic format of the generic component of the conceptual schema - the metasemantic grammar - stipulates (6) the semantics and rules for the domain-specific component of the conceptual schema (IId). Again, by applying verbalization with the aim to arrive at the next level (7a) to the diagrammatic format of the generic component (IIId), we obtain a textual representation of a core part of the generic component of the conceptual schema (IIIg).

As was illustrated previously, by applying generalization (8) at level III, the result was the identical representation of the metasemantic grammar, i.e. there is no higher conceptual level than level III.

The beauty of (IIId), the generic component of the conceptual schema, is that in effect it stipulates itself (9)! The route we followed regarding the creation of time invariant knowledge is also illustrated in Figure M.15, with concrete examples.

The result of (4b) and (7b) could be SBVR Structured English. The aim of (IIb) and (IIIb) is to be understandable to persons who do not know the diagrammatical representation, but do of course know well-expressed English sentences.

M.6 Summary and recommendation

SBVR is a major step forward for the business and education community. The era of sufficient attention to semantics has begun in earnest. SBVR covers many aspects which cannot all be discussed in one annex as the annex would become a textbook. Various useful concepts of SBVR have not been discussed in this annex as there was a limit to the number of concepts to be illustrated in this annex. For example, the concepts of necessity, obligation, permissibility, and possibility have intentionally not been discussed in this annex. It aims to be a useful add-on to the other annexes and has therefore concentrated on

a. a diagrammatic overview of some core concepts
b. concept definitions
c. a diagrammatic representation of fact types with the longest experience in industry
d. verbalization of fact instances, to be distinguished from rule verbalization as illustrated in Annexes C, F and I and
e. a small part of a long standing methodology which shows the power of SBVR.

My expectation is that SBVR 101, SBVR 102, and SBVR 103 will start in 2007. Sooner or later it will be taught in nearly all business oriented faculties.

Our recommendation to experienced fact oriented experts is to promote widespread use of SBVR.

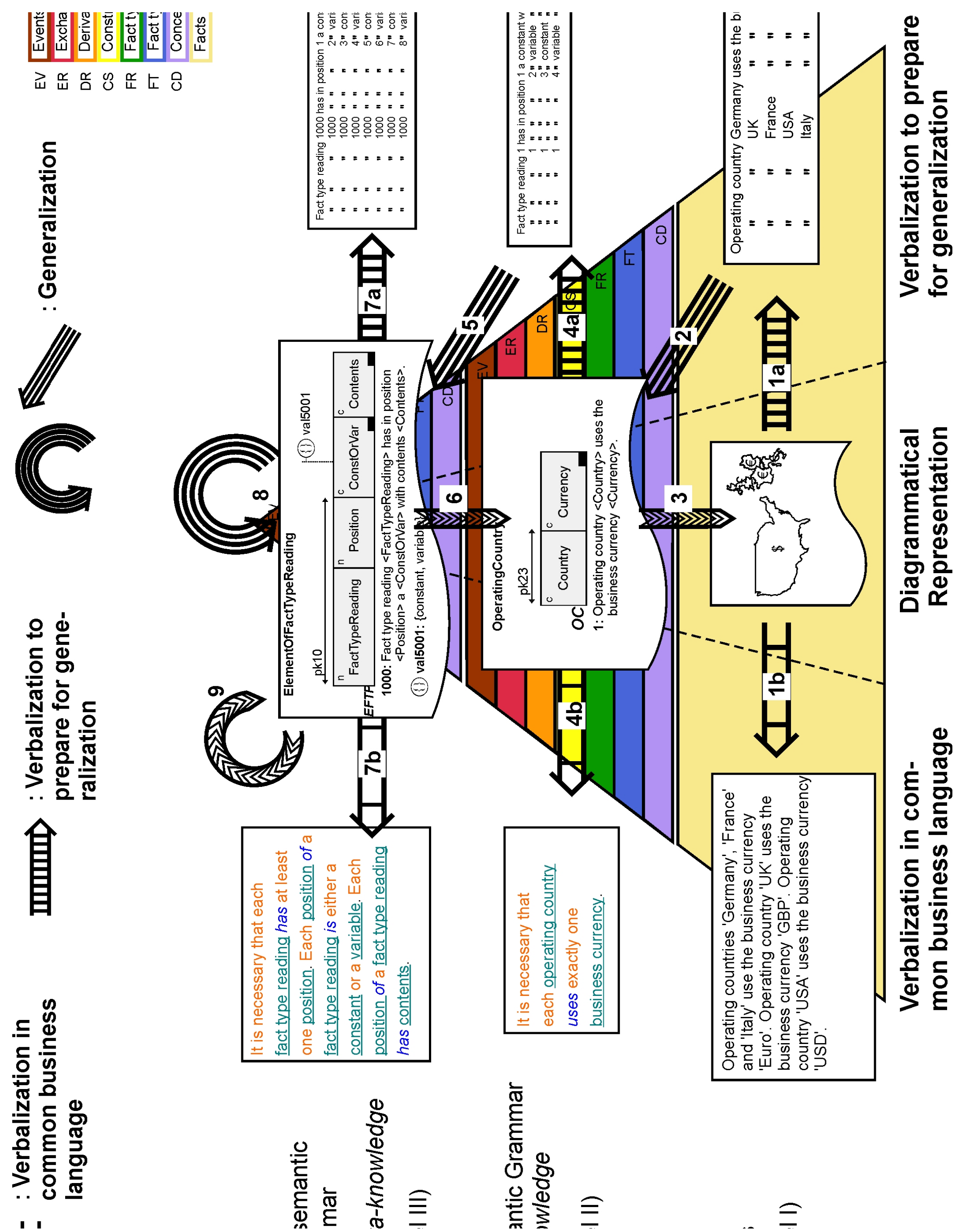

Figure M.15 - Knowledge triangle, with concrete input and output of processes