

演習コースⅡ 形式手法と仕様記述

「SBVR を活用した仕様の形式知化」

2014/02/28

(株)メタテクノ 宮本 陽子

目次

- 背景と課題
- 主語/述語/目的語の整理, ドメイン語彙の定義
- 標準語彙とは
- 論理演算子 Logical Operations
- 標準語彙を網羅するSBVR
- SBVR を活用した仕様記述文の作り方
- 定量化 Quantification
- まとめ
- 参考文献

背景と課題

私の現場では、**機能仕様書に記述されていない暗黙の仕様(行間)**に困っている。

「厳密に書きましょう」と言うだけでは、人によって、とらえ方が異なる。

【課題】

- **機能仕様書に記述されていない仕様, 考えられていない情報を考えるには？**

「担当者コードを入力する。」という仕様を例に考える。

これは厳密といえるか？

いいえ. いつ, 誰が, どこに, が不明

主語/述語/目的語, ドメイン語彙の定義(1/2)

① 担当者コード を 入力する.

目的語

述語

自由に記述すると、自分は知っているのに、無意識に省略してしまうことがある。

	述語	Xが(人)	Yに(入力先)	Zを(内容)
①	入力する	?	?	担当者コードを

述語ごとに、必要とする構成要素(主語/目的語)を意識するよう、表に書きだす。無意識に省略せず、明示を徹底しやすくする

② ユーザ が 担当者コード欄 に 担当者コード を 入力する

主語

間接目的語

直接目的語

述語

ユーザとは？

担当者とは？ユーザと同じ？

このように1つ1つの用語の意味を考え、定義する。

自分(達)のドメインの語彙が定まる

主語/述語/目的語, ドメイン語彙の定義(2/2)

そうは言っても, わかりきった事をいちいち書くのは大変.

ユーザが 担当者コード欄に 担当者コードを入力する.

ユーザが 担当者要件を検索する場合には, 検索キー欄に 氏名を入力する.

「ユーザが」「システムが」と常に主語や, わかりきったことを書く必要はない. しかし, 同じ主語を持つ述語を整理できたり, 構造化できたりする.

例)



1. ユーザが行うこと

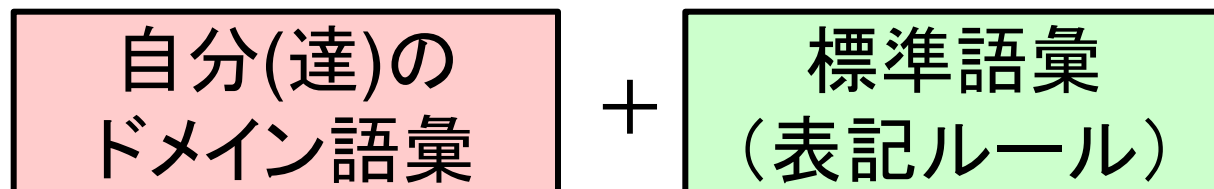
1-1. 担当者コードを入力する(担当者コード欄)

1-2. 担当要件を検索する場合, 氏名を入力する(検索キー欄)

1-3. ...

標準語彙とは

ここまでは、自分(達)のドメインに特化した語彙についてであった。



いくら定義したとはいえ、自分(達)だけの語彙を、独自のルールで好き勝手に並べた場合、理解しにくい内容になってしまうだろう。

そのため、ある決まった表記ルール、定型の言い回しに沿って、自分(達)の語彙を並べる必要がある。

この、決められた表記ルール、定型の言い回しを、ここでは「標準語彙」と呼ぶ。「標準語彙」は、例えば、C言語 や Java言語などのプログラミング言語で用いる「論理演算子」“or”, “||” のような、一意に定義された記法であり、意味を共有できるもの。

なお、SBVR という英語の形式言語には、次のような「論理演算子」がまとめられている。

論理演算子 LOGICAL OPERATIONS

p でない : it is not the case that p logical negation 論理否定 \neg

p かつ q : p and q conjunction 論理積 \wedge

p または q : p or q disjunction 論理和 \vee

p または q (のうち p かつ q をのぞく):

p or q but not both exclusive disjunction 排他的論理和 $\underline{\vee}$

p ならば q : if p then q implication 包含 \rightarrow

q もし p なら : q if p implication

p を言い換えると q : p if and only if q equivalence 同値 \Leftrightarrow

p かつ q の両方でない : not both p and q nand formulation

p でも q でもどちらでもない : neither p nor q nor formulation

これらの論理演算子は、文の関係を表す接続詞として、一意に定義された記法があり、意味が定義されたものである。つまり「標準語彙」(もし, and や or が通じなかったら困る)

標準語彙を網羅するSBVR

なぜ、「標準語彙」だけは共通で認識できるのか？

それは「標準語彙」が「論理学」に基づいたものであり、「論理」は人々の文化や背景によらず、書かれていることに対する認識を一致させやすいから。

SBVR は「論理学」に基づく、英語の形式言語である。OMGがまとめたもので、ビジネスルールを構成する用語とその定義を、すべての表現法とともに集めたもの。仕様記述の典型で、「標準語彙」の網羅版

概念まで整然と
記述されていて驚き！

SBVR: ● ○ ○

Semantics of Business Vocabulary and Business Rules

- OMG(Object Management Group) がまとめたもの
- V1.0 2008年1月公開
- ビジネス語彙, ビジネス上の事実(Fact)、ビジネスルールの意味を定義
- ソフトウェアツール間でビジネス語彙とビジネスルールを交換可能にする

SBVR を活用した仕様記述文の作り方(1/2)

1. 主語や目的語を一通りそろえた文で始める

ユーザ は 担当者コード欄に担当者コードを

rental *has* driver

【7パターンの質問】様相を表記する?

(義務)必ずすべき? : it is obligatory that

禁止する? : it is prohibited that

必要である? : it is necessary that

不可能である? : it is impossible that

可能である? : it is possible that

許可する? : it is permitted that

2. 様相の必要性を判断し, 追加する

様相論理(様相の必要性, 義務, 禁止, 必要, 不可能, 可能, 許可)の7パターンの質問を行い, 1で作った文型の後ろに置く.

ユーザ は担当者コード欄に担当者コードをすることが必須である

it is obligatory that rental *has* driver

SBVR を活用した仕様記述文の作り方 (2/2)

3. 定量化する

ユーザ は 担当者コード欄 にちょうど1つの 担当者コード を入力することが必須である。

it is obligatory that each rental *has* at least one driver

it is obligatory that each rental *has* no more than 4 drivers

【定量化のパターン】

すべての, いくつかの, 少なくとも 1つ, 少なくとも n 個, 多くとも 1つ, 多くとも n 個, ちょうど 1つ, ちょうど n 個, 少なくとも n 個かつ多くとも m 個, 1より多い

4. 条件を追加し, 限定する

初期画面を表示する場合には, ユーザ はちょうど1つの 担当者コード を 担当者コード欄 に入力することが必須である。

if a rental return *is* more than 4 hours *late* ...

定量化 QUANTIFICATION

each	<u>universal quantification</u>	すべての \forall
some	<u>existential quantification</u>	いくつかの \exists
at least one	<u>existential quantification</u>	少なくとも1つ \exists
at least n	<u>at-least-n quantification</u>	少なくともn個
at most one	<u>at-most-one quantification</u>	多くとも1つ
at most n	<u>at-most-n quantification</u>	多くともn個
exactly one	<u>exactly-one quantification</u>	ちょうど1つ
exactly n	<u>exactly-n quantification</u>	ちょうどn個
at least n and at most m ともn個かつ多くともm個	<u>numeric range quantification</u>	少なく
more than one	<u>at-least-n quantification</u> with $n = 2$	1より多い

SBVR では、定量化に用いる
10パターンの表現がまとまっている。

一階述語論理

ちなみに、形式仕様記述言語 VDM++ には、すべての(forall), 少なくとも1つ(exists), ちょうど1つ(exists1)という限量子がある。

まとめ

○ 機能仕様を考え、書くためには？

主語/述語/目的語を整理すること、ドメイン語彙を定義すること、および、SBVR を活用した仕様記述文の作り方の4ステップをふみ、パターンについて質問形式で考えることが、機能仕様を考え、書くためのヒントになる。

ぜひ、使ってみてください！ 例えば
直接的な利用：ご自分で文を書く際に、
間接的な利用：質問パターンをレビュー、テストの観点として

副次的効果

- 主語/述語/目的語を整理することと、SBVR のパターンや論理学の知識は、仕様記述だけでなく、日本語の作文にも役立つ
- 形式言語と論理学との関係について、考えることができる
- SBVR を読むだけでも、英語と論理学の勉強になる

参考文献

- Semantics of Business Vocabulary and Business Rules(SBVR), v1.0, OMG
OMG Document Number : formal/2008-01-02
- John Hall, Model Systems, Semantics of Business Vocabulary and Business Rules(SBVR), Presented at the BPM Think Tank, Arlington, VA 23 May 2006
- 三浦順治, 英語流の説得力を持つ日本語文章の書き方, 創拓社出版, 2009
- 三浦俊彦, フシギなくらい見えてくる! 本当にわかる論理学, 日本実業出版社, 2010
- 新井紀子, 数学は言葉, 東京図書, 2009