

ドメインユーザにもやさしいオブジェクト指向 自然日本語記述言語 simpleOONJ とその記述環境

島山正行^{†1} 池田武徳^{†2}
生井沢和也^{†3} 松本賢人^{†4}

要約: 本論文は OO 分析記述言語 OONJ の記述規則の数が多く複雑で, 理解し難さを解決するために考案されたドメインユーザ (以降, DU と略) にも使いこなせる簡潔な OO 技術, すなわち simpleOO 技術を開発した報告である. それを適用して OONJ の記述規則を四段階に分けた simpleOONJ (oonj1, 2, 3, 4 と呼ぶ) に再構成設計し, 記述を支援するエディタを開発した. 記述実験や記述例を検討した結果, oonj1,2 共に狙い通り DU に十分に簡潔に書けること, 記述エディタで更にやさしく使えることが検証され, その有効性が検証された. simpleOONJ と記述エディタは DU が実際に OO 技術を使いこなす際に有用であること, simpleOO 技術が有効であることが立証できた.

Object-oriented Natural Japanese Description Language simpleOONJ and Its Description Environments that are simple for Domain Users

MASAYUKI HATAKEYAMA,^{†1} TAKENORI IKEDA,^{†2}
KAZUYA NAMAIZAWA^{†3} and YOSHITO MATSUMOTO^{†4}

Abstract: We have already developed a Natural Japanese(NJ)-based, Object-oriented (OO) description language OONJ for the Domain Users (DUs) who are the experts of a certain professional domain in science/technology. The DUs have pointed out that the description rules of OONJ are too many and complex. To solve this problem, we have newly developed the "simpleOO" technology, and then designed a new simpleOONJ as the four stage language series oonj1, 2, 3, 4 with its description editors. Some description experiments/examples have shown that the oonj1 and 2 are simple enough for DUs, and that the description editor has provided far simpler environments than that without it. As for the conclusion, the simpleOONJ based on the simpleOO technology have successfully been realized.

1. はじめに

著者らは既出の論文¹⁾において自然日本語 (以降, NJ と略) 記述を用いてオブジェクト指向 (以降, OO と略) 記述を行う方法として OOSF (Object-oriented Structured Frame) を提案した. またこの OOSF を核とする OO 分析記述言語 OONJ^{2),3)} を設計/開発した. この OONJ は利用想定ユーザである科学技術分野の専門分野 (ドメイン) を持ち, 計算機を利用して開発

や計算, 研究を行うドメインユーザ (以降, DU と略) から見ると, 確かに記述力の“強さ”や“詳細さ”は十分に良好であるが, 記述規則が多くて利用方法も複雑であるとの指摘があった. そこで OONJ を DU にも使いこなせる仕組みへの改良が必要となった.

もう一つの動機は現代の複雑化した OO 分析技術を DU でも利用できる形で再構築したいと考えたことである. 現状の OO 分析技術は主として OO ソフトウェア工学に多く存在しているが, その技術体系は DU には複雑かつ煩雑なものでしかない.

そこで DU 向けの簡潔な OO 分析技術を提案する. それを仮に simpleOO 技術と呼ぶことにする. そしてその一つの具体化適用例として, 理解性と記述性を大きく向上させ, より簡潔に記述できる記述言語 simpleOONJ とその記述エディタを構築/提案する.

^{†1} 茨城大学工学部情報工学科
Department of Computer & Information Sciences, Faculty of Engineering, Ibaraki University
^{†2} (株) 東日本技術研究所 e-JAPAN IT Co., Ltd
^{†3} (株) ユー・エス・イー
^{†4} 茨城大学大学院理工学研究科情報工学専攻
Graduate School of Science & Engineering, Ibaraki University

2. simpleOO という発想/原理とその技術

—DU にも簡潔に使いこなせる OO 技術の構想—

simpleOO 技術は次の三段階からなる設計とした。

- (1) 「DU にとっての OO 構築原理」の適用
- (2) 簡潔な DU-OO 技術へと改良/組み替え/再構築/再設計など
- (3) 記述環境の設計/構築

上記 (1) の「OO 構築原理」は前論文¹⁾で提案した。

原理 (1)：離散化

原理 (2)：構造化

原理 (3)：明示形式性

この原理は現在著者達が知る最もシンプルな原理である。本論文ではこれを OO「分析」に適用する。

第一段階として (1) を用いて OO 技術を構築する。第二段階としてこれを改良/組み替えて技術の十分な練り直し DU 向けの方法を作り上げる。第三段階は使いやすさ向上のために記述 (支援) 環境を構築する。

3. simpleOONJ の提案とその設計

表 1 OOSF と NJ のレベル分けに依る simpleOONJ の分類

oonj	OOSF 構造記述規則	NJ 構文規則
oonj1	基本則 37 則の内 15 則. (相互関連図, 相互作用図)	文法的に正しい NJ 記述なら任意の記述
oonj2	基本則 37 則の内 28 則. サブスロット則. 付置属性記述. 複合文様式の一部. 相互関連図. 相互作用図.	NJ 単位文の並びに変換/再記述. mp と内部振舞い文の書き分けが必要.
oonj3	OOSF の全基本則 37 則. 任意深度サブスロット規則. 多重構造定義. 複合文様式. 対象世界状況定義.	
oonj4	全構造記述規則 (106 則)	全構文規則 (73 則).
OONJ	規則は同じだが, 完成度の高さを要求	式や記号も使った定量 (数量) 的な記述.

原理 (2) の構造化のために記述言語 OONJ の構造記述規則と NJ 文法を四つの段階に区分けした。

oonj1 はフレーム 1/スロットレベルまで

oonj2 はサブスロットレベルまで

oonj3 はサブスロットの複合文様式レベルまで

oonj4 = OONJ

表 2 構成要素種類表

fn1	オブジェクト名
fn2	属性
fn2.1	振舞い参照属性, fn2.5 相互作用 (伝達付置) 属性
fn3	振舞い
fn3.1	内部振舞い, fn3.2 相互作用伝達 (mp)
fn3.3	相互作用振舞い
fn4	相互関係
fn4.1	相互作用伝達 (mp), fn4.2 汎化/特化
fn4.3	集約/被集約, fn4.7 詳細化/簡易化

(注) 印は oonj1 用. 表全体は oonj2 用.

oonj1, 2, 3, 4 とは各段階毎の記述言語名である。これらを総称して simpleOONJ と呼ぶ。oonj1 ~ 4 の概要を表 1 に纏めて示した。oonj1 と 2 の記述要素種類名とファセット番号を表 2 に、構造記述規則を表 3 に示した。oonj3 と 4 は省略した。

oonj1

- (1) 離散単位としての記述単位であるフレーム
- (2) フレーム内部の離散要素としてのスロット
- (3) 離散単位間の構造化の仕組みである相互関係

これで必須最小限構成であり、OO 構築原理 (1) と (2) を満たしている。NJ 記述については、文法的に正しければ制約無しとした。

oonj2 では、スロット/サブスロットの詳細化とその構造化機能を追加する。

1. NJ 記述の“ NJ 単位文 ”化
2. 付置属性を記述
(fn2.1 振舞い参照属性, fn2.5 mp 付置属性)
3. スロット内部での集約構造化記述法の導入
4. よく用いられる複合文様式の典型である反復/条件分岐の各複合文を取り入れた。

4. oonj1 及び 2 専用記述エディタの構築

共通の基本方針としては、四つの主要要素 (フレーム, スロット, サブスロット, 相互関係) を個々にウィザードを用いて対話的に GUI 編集/入出力可能にする。特に入力促進画面では入力方法や簡単な事例を同時に提示する様にする。ウィザードはユーザの入力を待ちながら様々な情報をユーザに伝えて入力対象の説明や例などを充分にし、ユーザが入力に躓いてしまうことを防ぐことを狙う。特に重要な機能としては、

1. 各フレーム/スロット間の相互関連表示機能。
2. 編集/入出力ダイアログの指示/アドバイス機能
3. 記述経過や既記述例の任意時点での閲覧機能。
4. 各ダイアログにはその場の状況の説明を表示。
5. サブスロットに対する上記同様のダイアログ

表 3 simpleOONJ (oonj1, 2) の構造記述規則

1	<対象世界>:=:<フレーム> (< lf > + ³ <フレーム>)+
2	<フレーム>:=:<フレームヘッダスロット> (< lf > <スロット>)+
3	<フレームヘッダスロット>:=: フレーム番号 < sp > "fn1" < sp > <フレーム名> [< rt > <相互関係>]
4	<フレーム名>:=: オブジェクト名
5	<相互関係>:=:<相互関係記号> <相互関係相手名>
6	<相互関係記号>:=:(" " <相互関係名> " ") (" " <相互関係名> (" " " "))
7	<相互関係名>:=:"mp" "汎化" "特化" "集約" "被集約" "詳細化" その他
8	<相互関係相手名>:=: フレーム番号 <フレーム名> [" スロット番号 "]
9	<スロット>:=: 短冊形のスロット実線外枠線 (<スロット総称文> <分岐/反復文> <注釈>)(< lf > (<サブスロット> <分岐/反復文> <注釈>))*
10	<スロット総称文>:=: スロット番号 < sp > <ファセット記号> < sp > NJ 単位文 [< rt > <相互関係>]
11	<サブスロット>:=: < sp > < sp > <ファセット記号> <記述階層表現> NJ 単位文 [< rt > <相互関係>]
12	<ファセット記号>:=:"fn1" "fn2.1" "fn2.5" "fn3.1" "fn3.2" "fn3.3" "fn4.1" "fn4.2" "fn4.3" "fn4.7"
13	<記述階層表現>:=: < mr > [<インデント>] <階層表記線>
14	<階層表記線>:=: " "(集約)
15	< mr >:=: 直上の行に<階層表記線>があればそこまで右シフト, そうでなければそのまま. move right の省略形
16	<インデント>:=: 下の記述階層であることを示す各 DU 定義の任意一定幅空白.default は全角一文字分
17	< rt >:=: 右詰め
18	< lf >:=: 次の行またはスロットに移り, その左先頭に戻ることを示す記号
19	< sp >:=: 任意一定幅空白
20	<分岐/反復文>:=:<分岐/反復総称文> < lf > (< if 文> < while 文>)
21	<分岐/反復総称文>:=: スロット番号 < sp > <ファセット記号> < sp > ["(反復)" "(分岐)"] NJ 単位文 [< rt > <相互関係>]
22	< while 文>:=: < sp > < sp > <ファセット記号> <記述階層表現> "while" < sp > 反復条件判定 NJ 文 (< lf > <"while 文">反復)+ < lf > < mr > "while" ここまで
23	<"while 文">反復:=: < sp > < sp > <ファセット記号> <記述階層表現> <特定条件下振舞い文> [< lf > (<サブスロット記述> <分岐反復文記述>)]
24	< if 文>:=: < sp > < sp > <ファセット記号> <記述階層表現> "if" < sp > 分岐判定 NJ 文 < lf > <"if 文">分岐>
25	<"if 文">分岐:=: < then 振舞い文> [< lf > < else 振舞い文>]
26	< then 振舞い文>:=: < sp > < sp > <ファセット記号> <記述階層表現> "then" < sp > (<特定条件下振舞い文> < if 文>)[< lf > (<サブスロット記述> <分岐反復文記述>)]
27	< else 振舞い文>:=: < sp > < sp > <ファセット記号> <記述階層表現> "else" < sp > (<特定条件下振舞い文> < if 文>)[< lf > (<サブスロット記述> <分岐反復文記述>)]
28	<特定条件下振舞い文>:=:(NJ 振舞い文 [< rt > <相互関係>]) <分岐反復文記述>
29	<注釈>:=:[スロット番号]" (注釈:)" NJ 記述

(注) : oonj1 と共通の規則 : oonj1 との共通の規則を拡張 残りは oonj2 用規則

編集機能 (特に構造化機能) の実現 .

5. 記述例からの評価と考察

記述例として取り上げた自動販売機を図 1 と図 2 に示す「mp」はメッセージパッシングを意味し「|」は上の行の記述に集約されていることを示している。OO の知識があれば説明無しでも大半の内容が「何となく分かる」というのが初めてみたユーザの感想であるほどにやさしく書けているのが特徴である。記述エディタの使用実験から、OONJ を用いた記述よりは記述性と使いやすさが大幅に向上したと認められた。

6. simpleOO 技術の評価と考察

simpleOO 技術は第 2 章で述べた (1)(2)(3) の三段階

を OONJ に適用した場合の検証と評価/考察を行う。

(1) については、表 3 の左辺の要素を分類すると、<離散単位> (= < NJ 単位文 (記述) >), <構造記述子>, <要素特定子> の三種類に分類可能である。この分類から OOSF が原理 (1)(2)(3) の実現であると分かった。

(2) については、表 1 ~ 表 3 の四段階分けの特徴は、対象世界の離散単位、その離散単位内部での記述の詳細さの段階に従って区分されており、対象世界の集約階層構造に合わせた区分でもある。それは DU のモデリング/記述に必ず適応した区分設計でも分かった。

(3) についても、oonj1, 2 の記述実験、記述エディタの利用実験、記述例 (図 1 や図 2) を通して DU の使いやすさと利便性の向上が確認された。

以上から simpleOO 技術が DU にとって適切な技

術であること, simpleOONJ が OO 構築原理を充分に実現した適切な記述言語及びその記述エディタであること, が分かった.

7. ま と め

記述言語 simpleOONJ(oonj1, 2) とその記述エディタは DU 向けの簡潔な OO 記述を実現し, その有効性を検証できた. この結果は simpleOO 技術の構築の方法が妥当であること, 一つの典型例ができたことで同時に simpleOO 技術の基礎の一部も確立できた.

今後の課題としては, simpleOO 技術の設計/実装段階への適用, oonj1~4 の記述規則とその記述環境の改良, 記述例の作成と記述実験 (普及) 活動がある.

参 考 文 献

- 1) 畠山正行, オブジェクト指向分析自然日本語構造化フレーム OOSF の設計と表現技法, シミュレーション学会誌, 22-4, 50/67, Dec., (2004).
- 2) 畠山正行, オブジェクト指向自然日本語記述言語 OONJ の設計とその記述例, 第 145 回ソフトウェア工学研究会報告, 2004-SE-145, 53/60, (2004).
- 3) 畠山正行, 松本賢人, オブジェクト指向自然日本語記述言語 OONJ の設計とその応用記述例, 第 102 回 HPC 研究発表会研究報告, (2005).

1 fn1	自動販売機	提供する	??:人
1 fn4		集約	??:ボタン
2 fn4		集約	2:コントローラ
3 fn4		集約	??:硬貨投入口
4 fn4		集約	??:硬貨箱
5 fn4		集約	??:おつり口
6 fn4		集約	??:おつり返却バー
7 fn4		集約	??:商品格納庫
8 fn4		集約	??:商品口
9 fn4		集約	??:金額表示 LED

2 fn1	コントローラ	集約	1:自動販売機
1 fn3	ボタンが押されたことと, どのボタンが押されたのかを受信する.	mp	??:ボタン [??]
fn3	もし, 購入可能状態でないなら, 何もしない.		
fn3	そうでないなら, 押されたボタンから商品名, 金額を判定する.		
fn3	商品名の商品を出すように指示する.	mp	??:商品格納庫 [??]
fn3	もし 合計金額 > 商品金額 であるなら, おつりを求める. (おつり金額 = 合計金額 - 商品金額)		
fn3	おつり金額を送信する.	mp	??:硬貨箱 [??]
fn3	ボタンにランプを消灯させるように伝える.	mp	??:ボタン [??]
fn3	お金の投入待機状態になる.		
fn3	合計金額をゼロにする.		

3 fn1	ユーザ (人)	利用する	1:自動販売機
1 fn3	金額が十分になるまで硬貨を投入する.	mp	??:硬貨投入口 [??]
fn3	もし購入しないなら, おつり返却バーを回す.	mp	??:おつり返却バー [??]
fn3	欲しい商品のボタンを押す.	mp	??:ボタン [??]
2 fn3	商品を受け取り, 飲む.	mp	??:商品口 [??]
3 fn3	おつりを受け取り, 財布へしまう.	mp	??:おつり口 [??]

図 1 oonj1 で記述された自動販売機の世界

3 fn1	コントローラ	集約	1:自動販売機
1 fn3.3	ボタンが押されたことを受信し商品を出す様指示する.		
fn3.2	ボタンが押されたことを受信する.	mp	2:ボタン [3]
fn2.5	ボタン種別		
fn3.1	各ボタンに対する商品の商品種類とする.		
fn2.1	(ボタン種別, 商品種類)		
fn3.1	(分岐) 商品購入待機状態であるか.		
fn3.1	if 商品購入待機状態である.		
fn3.2	then 商品を出すように, 商品格納庫に指示する.	mp	8:商品格納庫 [1]
fn2.5	商品種類		
fn3.1	(分岐) おつりを計算する.		
fn3.1	if 合計金額 > 商品金額.		
fn2.1	(合計金額, 商品金額)		
fn3.1	then おつりを (合計金額 - 商品金額) で求める.		
fn2.1	(おつり金額, 合計金額, 商品金額)		
fn3.2	おつり金額を送信する.	mp	5:硬貨箱 [2]
fn2.5	おつり金額		
fn3.2	ボタンにランプを消灯させるように伝える.	mp	2:ボタン [2]
fn2.5	ランプ消灯信号		
fn3.1	合計金額をゼロにする.		
fn2.1	合計金額		

11 fn1	ユーザ (人)	利用	1:自動販売機
1 fn3.3	商品を購入する.		
fn3.1	(反復) 金額が十分になるまで硬貨を投入する.		
fn3.1	while 投入金額 < 商品金額		
fn3.2	硬貨を投入する.	mp	4:硬貨投入口 [1]
fn2.5	硬貨		
fn3.1	(分岐) 購入しない場合		
fn3.1	if 購入しない		
fn3.2	then おつり返却バーを回す.	mp	7:おつり返却バー [1]
fn2.5	返却レバーの回転力		
fn3.2	欲しい商品のボタンを押す.	mp	2:ボタン [3]
2 fn3.3	商品を受け取り, 飲む.		
fn3.2	商品を受け取る.	mp	9:商品口 [1]
fn2.5	商品		
fn3.1	商品を飲む.		
3 fn3.3	おつりを受け取り, 財布に入れる.		
fn3.2	おつりを受け取る.	mp	6:おつり口 [1]
fn2.5	おつり		
fn3.1	おつりを財布に入れる.		

図 2 oonj2 で記述された自動販売機のコントローラとユーザ