

RAD/SVにおけるシナリオ入力法

3M-8 高橋 久* 真下祐一** 柳町美帆** 湯浦克彦** 大成宣之*** 亀田達也***

* 日立超LSIエンジニアリング（株）
 **（株）日立製作所 システム開発研究所
 ***（株）日立製作所 情報システム事業部

1. はじめに

オブジェクト指向型のモデル構築においては、用意された部品によれば容易にモデル構築が可能である¹⁾。ところがこの部品の作成に関しては一般にシステムエンジニアが行っていたが、プログラム言語や組み込み方法を知っている必要があり複雑で時間がかかっていた。RAD/SV²⁾では、部品の機能をシナリオ（具体的なオブジェクトの動作）という形で入力することで、脱プログラミングを試みた。

2. RAD/SV構成

RAD/SVはオブジェクト指向言語CLOS（Common Lisp Object System）上に図形入出力言語³⁾、適用型図形エディタ⁴⁾およびそれを利用した各種エディタを設けた構造となっている（図1）。図形入出力言語は、図形オブジェクト・アニメーションメソッドの記述を可能にする。適用型図形エディタは、図形オブジェクトから機能のオブジェクトを操作する汎用インタフェースを提供する。シナリオエディタは要求仕様をシナリオとして入力し、シナリオを統合することで機能のオブジェクト・メソッドを生成する。アニメエディタはレビュー時の表示方法を指定すると、図形オブジェクト・アニメーションメソッ

ドを生成する。シミュレーションエディタはシナリオエディタとアニメエディタにより生成された部品を利用して、システム構成の作成、シミュレーション実行および実行制御を行う。

3. シナリオ

3.1 シナリオ概要

要求仕様の検討段階においては、エンドユーザからの断片的な仕様、即ちシステム動作の具体例を対話的に入力・変更し、エンドユーザに提示することを繰り返すことが有効である。このシステム動作の具体例のことをいくつかのオブジェクト指向方法論ではシナリオと呼び、トレース図等で表現している⁵⁾。RAD/SVではこの概念を発展させ、オブジェクトおよびオブジェクト間のメッセージの流れをより詳細に、具体的に表現する。シナリオエディタ上ではオブジェクトをアイコンで、オブジェクト間のメッセージの連鎖をアイコン間の結線で表現する。そして1つのシナリオでは、メッセージの流れが一筆書きで書ける範囲を記述する。これは例えばループとループ脱出後の処理は、別々のシナリオとして記述することを意味する。このように分割して記述したシナリオ間の連鎖はシナリオ統合することで、一連のメソッド及びオブジェクトを生成する。これによりエンドユーザからの仕様変更は、当該のシナリオを修正または追加し、統合することで迅速に対応できる。

3.2 シナリオ入力

シナリオの入力を、物品管理システムの見積・受注サービスの例で説明する（図2）。まずエディタ（ID=1）上で顧客、営業担当、見積伝票の各アイコンを、オブジェクト定義アイコン（図2(1)(a)）にオブジェクト名称、属性名、メッセージ名を入力して定義する。次にメッセージの連鎖は図2(1)(b)の様に営業担当の「見積処理」と顧客の「検討」とを結線して定義して行く。

このようなメッセージの連鎖の他にシナリオの構文として下記のを規定している。

(1) ループ 既に呼び出されているメッセージを他から呼び出すことで表現する（図2(1)(c)）。

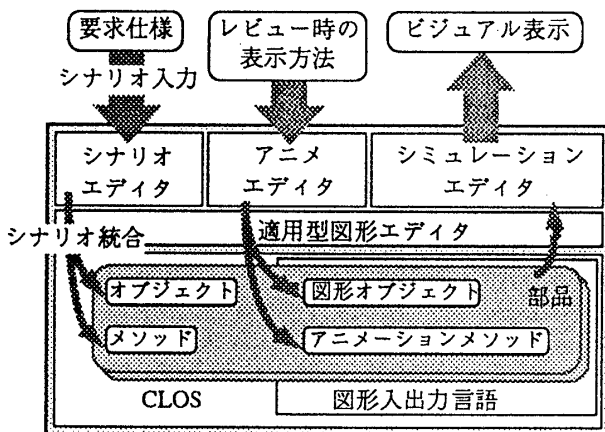


図1 RAD/SVシステム構成

Scenario input method for RAD/SV

Hisashi TAKAHASHI*, Yuuichi MASHITA**,
 Miho YANAGIMACHI**, Katsuhiko YUURA**,
 Noriyuki Onari***, Tatsuya KAMEDA***

* Hitachi ULSI Engineering Corp.
 ** Systems Development Laboratory, Hitachi, Ltd.
 *** Hitachi, Ltd. Information Systems Division

(2) 引数 メッセージ連鎖の線とオブジェクトとを結線することで引数を表現する。図2(1)(d)は「検討」メッセージに見積伝票を引数に渡すことを表す。

(3) オブジェクト属性値の更新 メッセージ連鎖定義時に、関連するオブジェクトに対し属性および具体値を入力することで、属性値の更新を表現する。図2(1)(e)は「検討」メッセージの中で見積伝票の「顧客名」、「見積価格」属性の各値をそれぞれ「日立」、「10万」に設定していることを表す。

(4) 分岐 分岐させたいメッセージを別エディタ上に複製し、別シナリオとすることで表現する。

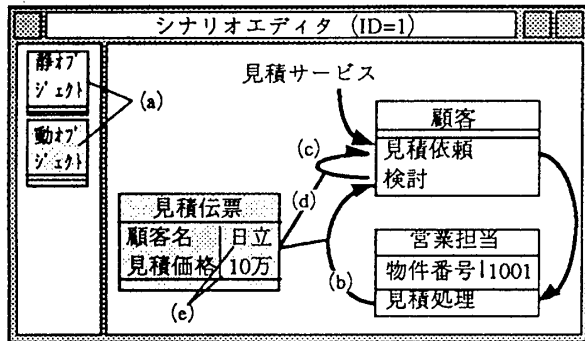
例ではまずエディタ(ID=1)上各アイコンをエディタ(ID=2)上に複製しておき、次に「検討」メッセージ(図2(1)(c))をエディタ(ID=2)上に複製(図2(2)(f))することで分岐を表す。

(5) 分岐条件 分岐として複製したメッセージに対して次の様な分岐条件を表現することができる。

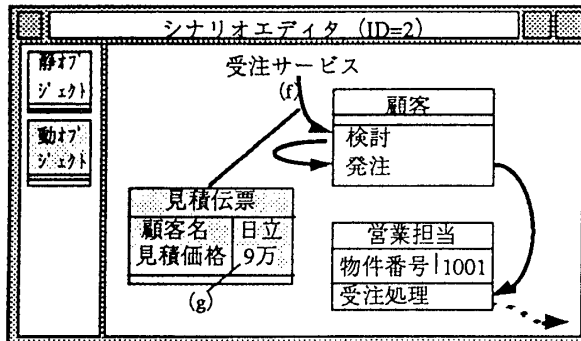
(a) メッセージに関連するオブジェクトのある属性値と具体値が一致するかどうか。

(b) メッセージ引数の型・数が合致するかどうか。

(c) コメントをダイアログ表示し、メッセージを選択するかどうか。



(1) 見積サービス



(2) 受注サービス

凡例

オブジェクト名	
属性名	値
メッセージ名	

図2 シナリオ入力

3.3 シナリオ統合

統合は各シナリオエディタ上のシナリオをシミュレータエディタ上にドラッグ&ペーストすることで行う。この時エディタ上の各メッセージは同名のヘッダメソッドと、エディタ毎のIDを持つボディメソッドに分割定義する(図3)。ヘッダメソッドはボディメソッド群を保持している。ボディメソッドはシナリオIDの他、分岐条件や次のメソッドへの連鎖を保持している。

シミュレーション実行時には、ヘッダメソッドにおいてボディメソッドのセレクションを行うことで分岐等を制御している。

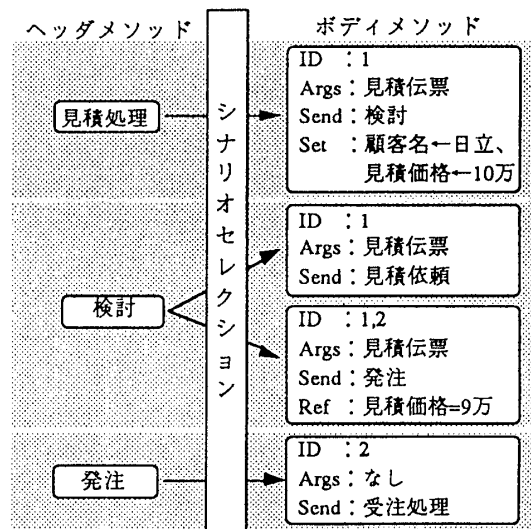


図3 シナリオ統合

4. おわりに

具体例によるシナリオ入力とシナリオ統合によるプログラム生成により、脱プログラミングによる要求仕様定義の見通しを得た。

参考文献

- [1] 湯浦,外: ODETTE:オブジェクト指向CLOSをベースとした設計支援構築環境,情報処理学会オブジェクト指向ソフトウェア技術シンポジウム論文集,1-11(1991).
- [2] 真下,外: オブジェクト指向による要求仕様視覚化ツールRAD/SV,情報処理学会第49回全国大会論文集(1994).
- [3] 高橋,外: ODETTEにおけるオブジェクトの図形入出力言語,情報処理学会第44回全国大会論文集,5-41(1992).
- [4] 久保,外: CLOSを利用したODETTEのウィンドウインタフェースの実現,情報処理学会第44回全国大会論文集,5-39(1992).
- [5] J.ランボー,外著,羽生田栄一監訳: オブジェクト指向方法論OMTモデル化と設計,株式会社トッパン