

統合環境 Vela におけるデザインレビュー支援

落水 浩一郎

静岡大学 工学部 情報知識工学科

我々は、以下の2種類の協調支援機能の実現を目標としたプロトタイプ Vela (Versatile Environment based on Logic programming and AI techniques) の開発を進めている。(1) 設計の熟練者から初心者への技術移転の促進、(2) プロジェクト構成員間の意思伝達、合意形成、情報交換の促進。本稿では、後者の機能の実現をめざして定義した交信支援のプロセスモデルとそれを応用したソフトウェアデザインレビュー支援アーキテクチャについて報告する。本稿で提案するモデルの特徴は、話題対応に永続的仮想リンクを設定する機能と、討論の内容を構造的に記録する機能にある。

Design Review Support in the Integrated Environment Vela

Koichiro OCHIMIZU

Department of Computer Science, Faculty of Engineering, Shizuoka University
3-5-1 Johoku, Hamamatsu, 432 JAPAN

Vela (Versatile Environment based on Logic programming paradigm and AI techniques) is a process-oriented software environment, especially designed for (1) Promoting technology transfer from design experts to the novice, (2) Supporting communication among project members based on a project structure. In this paper, we report a process model for communication support and an architectural design of design review support with applying the process model. The major features of our model are; establishing a persistent virtual link for each issue and structural recording of the conversation history based on types of opinions.

1. はじめに

「多大な工数を費やして開発しなければならないようなアプリケーションにおいては、すべてを理解している人は組織のどこにも存在せず、組織として理解していることの全体は多くの経験豊富な専門家一人一人に分散している [1]」ような現状があり、分散環境においてはこの事実はさらに顕在化する傾向にある。

このような問題に対する一つの接近法として、近年、プロセス・モデルに基づく統合環境の構築が試みられている。統合環境に期待されるサービス機能としては、(1) 集団活動の制御、(2) 熟練者の作業プロセスを再利用することにより、より経験の少ない人の活動をナビゲートする、(3) ソフトウェアプロジェクト構成員間の情報交換 (交信) の支援、(4) 開発対象物に対する目標と基準を設定すると共に、個々人の活動を調整する (版管理等) 機能などがあげられる。さらに、対象領域の特徴、開発の規模や分散度等をパラメータとして、ユーザ固有の作業形態に適合した開発環境を生成する機能も必要である [2]。以上の関係を図 1 にしめす。

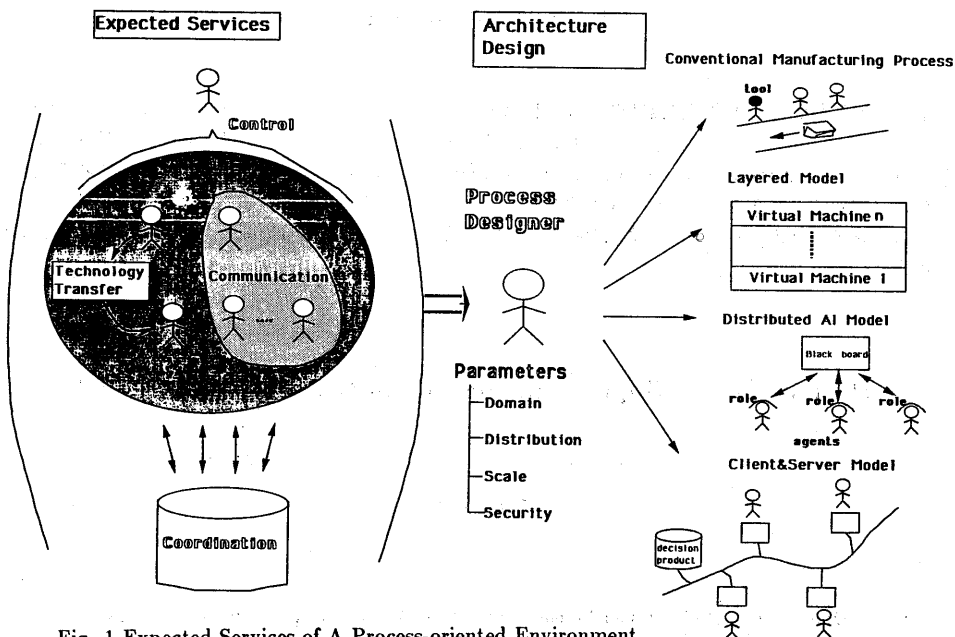


Fig. 1 Expected Services of A Process-oriented Environment

われわれはプロセスモデルに基づく統合環境の一例として V e l a (Versatile Environment based on Logic programming and AI techniques) を開発中であり、先にのべたサービス要求間の相互関係を図 2 に示すように概念モデルとして表現し、それに基づいて各部のプロセスモデルとアーキテクチャを設計開発しつつある [3]。図 2 の背景には以下のような考え方をしている [4]。「ソフトウェア開発は複数のタスクからなり、それらはプロジェクト開始時または進行時に動的に生成される。各タスクはプロジェクト管理者からみた仕事の最小単位であり、実行順序はタスク・ネットワークにより制御される。タスクには 2 種類の型があり、それぞれプロセス指向タスク、情報指向タスクと呼ぶ。プロセス指向タスクにおいては種々の中間生成物を別の中間生成物に変換する作業 (Transformation) が実施され、情報指向タスクにおいては、意志決定、合意形成、情報交換等の開発者相互間の交信 (Communication) が実施される。

両者のタスクのインスタンスは生成物間の依存関係をもとに定義されるタスクネットワーク (タイムドベトリネット) により順序づけられる (Control)。また、中間生成物や交信の記録はオブジェクト指向データベースとして構造化され、その品質基準や構成管理のルールと共に保存される (Coordination)。

このうち、Transformation 部に関しては、経験度の高い作業熟練者の変換プロセスを事例型推論によって収集し、再構成、洗練化をおこなった上でルールベースのプロセス記述に変換し、メタ推論、仮説推論により経験度のより少ない開発者に利用させる知識処理アーキテクチャとプロトタイプ開発を実施した [5、6、7、8]。本稿では Communication 支援に関するプロセスモデルを提案し、ソフトウェアデザインレビュー支援アーキテ

クチャに適用した結果を報告する。

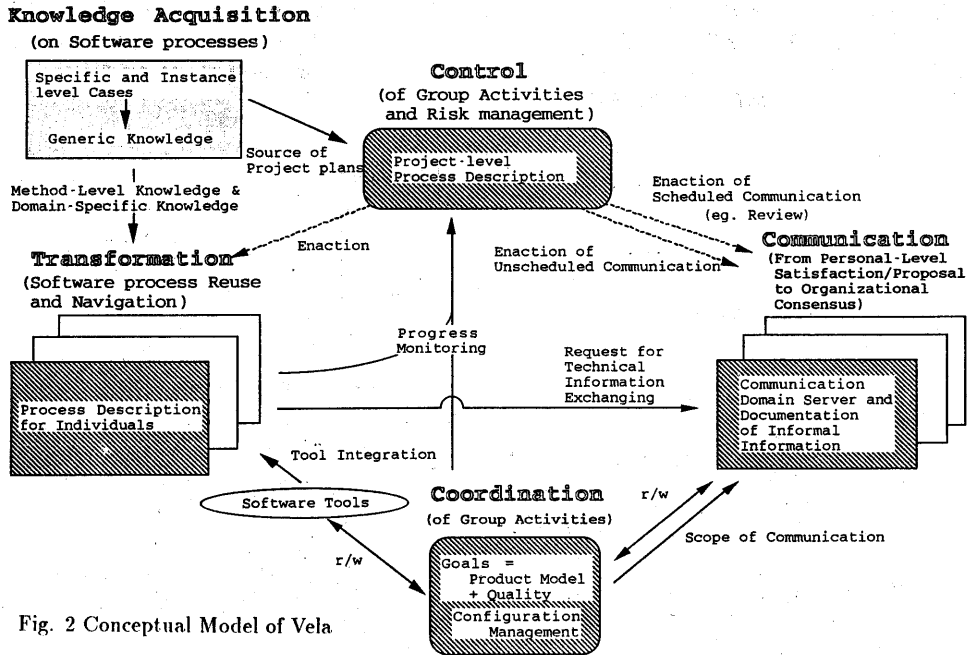


Fig. 2 Conceptual Model of Vela

2. 交信のプロセスモデルの設計

L. オスタワイルによって提案されたプロセス記述は [9]、集団の制御やプロセスの再利用には有効である。われわれは、プロセス記述を2レベル(プロジェクトレベルと個人レベル)とし、前者はタイムドベトリネット、後者はルールベース記述を基本にした接近を進めている [3、10]。

しかし交信の支援がプロセス記述によって円滑になされるという前提はうけいれがたい [2]。むしろ電子メールや電子掲示板が発展した形を想定したい。一方、単なる電子メールでは以下のような問題が生じる。すなわち、ソフトウェア開発の進行にともなって生じる様々な話題に対応して交信は発生するはずであり、そこで検討された内容を話題毎にグループ化して構造的に文書化することは意義がおおきい [3、11]。このような立場から、論点対応に永続的な仮想リンクを設定し、リンクを介した対話の進行にともない検討内容が構造的に付加されていくようなプロセスモデルを定義した。

交信プロセスのモデル化にあたっては以下の点を考察する必要がある。

- (1) 交信のきっかけ — プロジェクト計画にあらかじめ組み込まれるもの (例 レビュー) と自然発生的におこるもの (例 技術情報交換) がある
- (2) 交信の形態 — 合意形成, 指示伝達, 情報交換等
- (3) 交信のトポロジー — グループ単位, 特定の話者間
- (4) 交信参加者の役割 — 情報の送り手, 情報の受け手, モデレータ, 観測者
- (5) 伝達のメディア — 文書, 絵, 肉声 (揮発性の情報がおおい)

モデル化にあたって、さしあたっては、プロジェクト計画にあらかじめ組み込まれるもの、合意形成、グループ単位、テキスト情報を対象にした。

また、アーキテクチャ設計上考慮すべき問題点としては、

- (1) 時間 / 空間分散に対する処置 (人間は忘れやすい生き物である)
 - (2) 途中から参加する人への対応
 - (3) 発言内容間の矛盾の検出 (人間は不完全な生き物である)
 - (4) 情報の保護と情報アクセス範囲の設定 (情報の価値、必要性は人によって異なる)
- 等があげられるが、ここでは (1) のみを対象にした。図3にプロセスモデルを示す。

図3において、Domain Serverは話題対応に永続的な仮想リンクを設定し、会議の中断、再開を支援するサービスを行う。Domain Serverによってはられる仮想リンクを以後交信ドメインと呼ぶ。交信ドメインを流れる会話は発言型毎に整理され、オブジェクト指向データベースに登録される。これをDiscussion Boardと呼ぶ。それぞれの発言型が電子メール、Discussion Boardが電子掲示板に対応する概念である。CMはCommunication Managerの略記であり、交信参加者やDiscussion Boardからのメッセージを他に配布する機能を持つ。

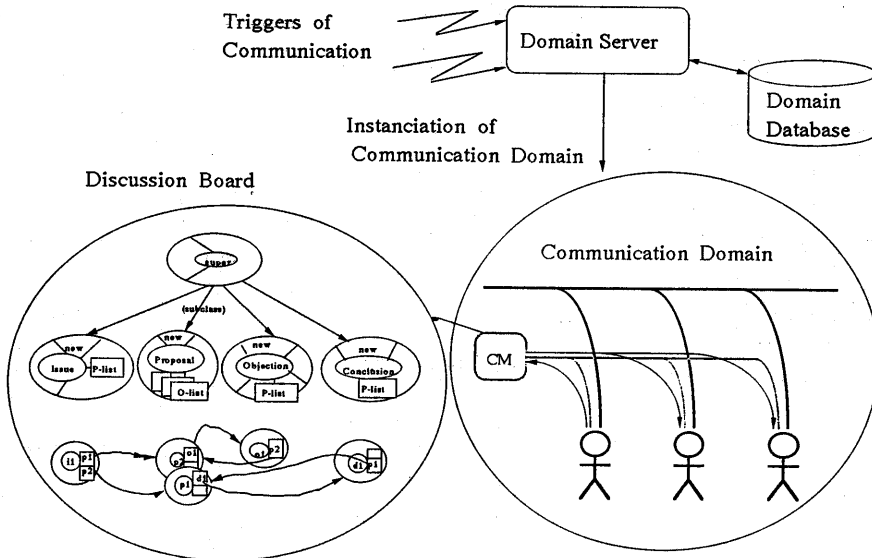


Fig. 3 Process Model for Communication Support

3. ソフトウェアデザインレビューの分析

文献[12]を以下の視点から分析した。(1)レビューのきっかけ、(2)レビューの種類、(3)レビューの対象、(4)レビューの構成員と役割、(5)レビューの目的、(6)利用するチェックリストとチェック項目、(7)レビューの実施法とその詳細、(8)レビューリーダー、書記、レビューア、管理者の心がけ。図4に分析時に作成した作業シートを示す。

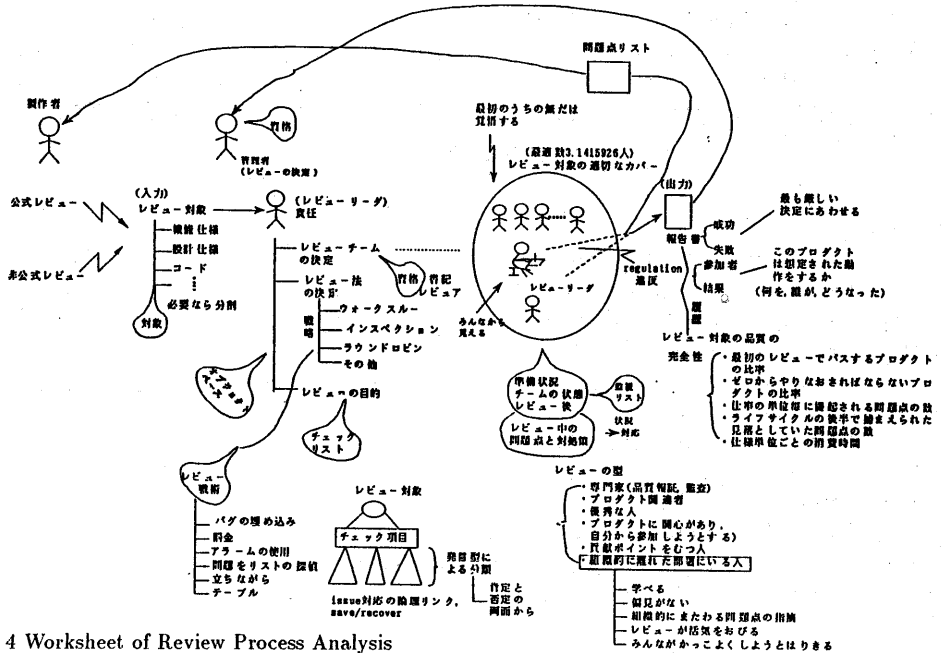


Fig. 4 Worksheet of Review Process Analysis

図から読みとれるように、製作者によって作成された中間生成物は、レビュー対象の型に応じたレビュー法、チェックリスト、レビュー参加者の発言等をもとにして情報を付加され、最後にはそれらをまとめた報告書になる。すなわち、レビュー作業とは製作者(個人レベル)の満足感を、チームレベルの合意に変換するプロセスであり、情報指向タスクの典型である。

4. ソフトウェアデザインレビュー支援アーキテクチャの設計

2節、3節における検討結果をまとめてレビュー支援のアーキテクチャを図5のように設計した。レビュー作業はレビュー計画とその実施からなる。図5において、左上隅の図は一つのタスクを実行するための制御単位である。◎はプロセス指向タスク起動の前提条件が成立したと、○はレビューの準備がととのったこと、ラベルRがついた○はレビューが終了したことを表す。レビューはレビュー対象とレビューリーダーを指定して開始される。右上隅の図はレビュー計画を示す。レビューリーダーはレビュー対象の型に応じたレビューテンプレートの中から最も適したものを選び、その内容を埋める。下側の図はレビューの実施支援を表す。内容が埋められたレビューテンプレートがDomain Serverにわたされ、レビュー参加者間に交信ドメインが設定される。交信ドメイン上での検討内容はレビュー履歴としてDiscussion Board中に意見のクラス階層から実体化されて記録される。報告書生成機能が問合わせ機能を利用して報告書を生成する。

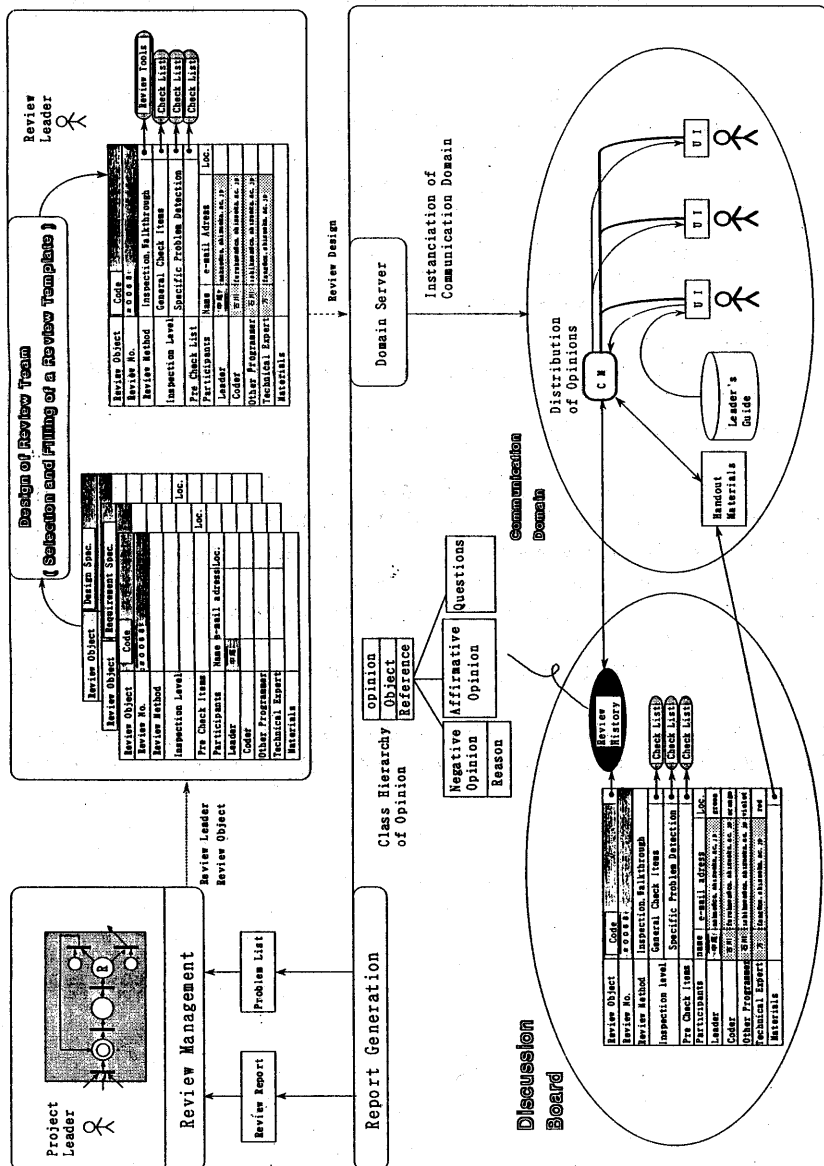


Fig. 5 An Architecture for Design Review Support

5. 討論と今後の予定

本稿では、分散環境上でのソフトウェア開発を前提として、プロジェクト構成員間の交信を支援する一つの方式を提案した。提案したモデルやアーキテクチャの特徴は以下の点にある。

- (1) 交信ドメインの決定支援…中間生成物を入力として、交信に必要な環境を設定し、話題に関係する人々の間に情報交換のための通信チャンネルを自動設定してくれる機能
- (2) ネットワーク上を長期間にわたって進行する討論の内容を、話題毎に構造化して記録し、合意形成、情報交換の結果の文書化を支援する情報構造

このような研究ははじまったばかりであり事例が少ない[13, 14]。プロトタイプ開発と実験の積み重ねによる問題点の認識、改善のアプローチが有効な解を見つけだすための最善の方法であろう。本稿でのべたVelaに関しては、レビューテンプレートの設計、Domain ServerとCommunication ManagerのプロトタイプがTCP/IPプロトコルの上に開発されたばかりである。今後実験を積みかさねていく予定である。また、ワークスルー用のツールとして、プロセス指向タスクの実行時に利用されたルール群を利用して作成過程の再現を補助するような機構の実現も計画している。

謝辞 本研究はSDAコンソーシアムの補助金と、科研費重点領域研究(1)(課題番号02249109)の援助のもとにおこなわれた。記して謝意を表す。また、本稿3節における、本学学生、方嬢、石川、古川両君の資料分析と討論に感謝します。

参考文献

- [1] S. シュレイアー、S.J. メラー著、本位田、山口訳、「オブジェクト指向システム分析」、啓学出版、1990.
- [2] K.Ochimizu, "Current Progress on Vela Project", 文部省科研費国際研究交流、第1回研究交流会、at UC Irvine, 1990.
- [3] 落水, "ソフトウェアプロセスモデルに基づくソフトウェア開発支援環境Vela", 日本ソフトウェア科学会第7回大会論文集, pp.205-208, 1990.
- [4] K.Ochimizu, X.Fang, T.Ohta, "Vela - A Software Design Support Environment based on Logic Programming Paradigm", 静岡大学大学院電子科学研究科研究報告、第11号、pp.75-88, 1990.
- [5] 落水、山口, "ソフトウェア開発における協調支援環境Vela - (1) ねらいとアーキテクチャー", 第41回情報処理学会全国大会講演予稿集、1990.
- [6] 山口、落水, "ソフトウェア開発における協調支援環境Vela - (2) プロセス知識の獲得と利用", 第41回情報処理学会全国大会講演予稿集、1990.
- [7] K.Ochimizu, "Knowledge Acquisition in Software Design Processes", International Software Process Symposium, Tokyo, 1990.
- [8] 山口、落水, "ソフトウェアプロセスモデル構築における知識獲得と利用方式", 人工知能学会研究会資料、1991.
- [9] L.Osterweil, "Software Processes are Software Too", 9th ICSE, pp.2-13, 1987.
- [10] K.Ochimizu and T.Yamaguchi, "A Process Oriented Architecture with Knowledge Acquisition and Refinement Mechanisms on Software Processes", 6th ISPW, 1990 (in press).
- [11] 落水, "CASEにおけるデータベースの機能と役割", 平成元年電気情報関連学会連合大会講演論文集、分冊5、1989.
- [12] D. フリードマン、G. ワインバーグ著、岡田監訳、「ソフトウェア技術レビューハンドブック」、TBS出版会、1987.
- [13] C.Potts and G.Bruns, "Recording the Reasons for Design Decisions", 10th ICSE, pp.418-427, 1988.
- [14] 鈴木、村田、鎌田, "電子メールをベースにしたグループウェア記述言語「LAMB」", 第32回プログラミングシンポジウム報告集、pp.87-96, 1991.