

並行設計における調整コミュニケーション支援システム

田淵 篤、吉府 研治、垂水 浩幸

{tabuchi,yoshifu,tarumi}@obp.cl.nec.co.jp

NEC

関西 C & C 研究所

並行設計においては、製品のライフサイクルに関わる様々な工程から出された設計への変更要求を、作業者間のグループレビューによって調整する必要がある。本稿では、そのような調整のコミュニケーションを支援するシステム“CE-Notifier”について提案する。CE-Notifierは、統合化された設計データモデルをリポジトリ上で共有し、モデルに対する変更要求を開発者に通知する順序を最適化する機能と、特定の変更要求に関するメッセージを選択的に収集・表示する機能を提供する。これによって、調整のコミュニケーションの過程を分かりやすく整理し、グループでの意思決定を円滑化することが出来る。

Communication Support System for Negotiation in Concurrent Engineering

Atsushi Tabuchi, Kenji Yoshifu, Hiroyuki Tarumi

NEC

Kansai C&C Research Lab.

In a concurrent engineering process, requests for a design change must be negotiated by the different engineers. This paper proposes an e-mail based communication support system which clarifies the negotiation process. This system dynamically creates a review path that may reduce the communication loss, whenever distributing a design change request from the data repository. The results of a review for the same request are automatically gathered and the shared viewer shows these clusters to the reviewers. These functions also help engineers' decision making to reflect the review result.

1 はじめに

「並行設計 (Concurrent Engineering; 以下 CE と略す)」の概念は、ソフトウェア/ハードウェアの設計・製造プロセスを革新するコンセプトとして、“BPR”と並んで近年注目されてきている。

CEの大目標は、開発者が製品ライフサイクル(構想から廃棄まで)の全ての要素を総合的に考慮して設計できるようにすることである[1]。CEは、様々な工程に関わる開発者によるチーム設計であるから、設計データの共有・共同レビュー・プロジェクト管理などの面で、グループウェアシステムによる支援が不可欠と考えられる。

筆者らは特に、CEにおいて共同レビューを行なう際のコミュニケーション支援に着目している。CE的な開発業務では、共通の設計対象に対して異なる工程から様々な設計への要求が加えられることになり、それらの要求をレビューによって調整することが頻繁に起こる。しかも複数のレビューが非同期的に進行するので、見落としや混乱を招きやすい。本稿では、このような複数のレビューを分かりやすく整理し、CE特有のコミュニケーションを円滑化するシステムとして“CE-Notifier”を提案する。

2 並行設計における調整

CEの一般的な実現イメージは、図1のようなものである。即ち、

- 設計対象は、仕様書等のドキュメントを含めてネットワーク内の共有データベース(リポジトリ)上で一元的に管理される。
- 開発者は、企画・設計・検査など異なる工程に属し、それぞれの立場から、リポジトリ内の設計対象を参照する。
- 更に、権限があれば、データの作成・更新やコメントの付加を行なう。

このように、各工程の作業経過/結果を共有し、必要に応じていつでも参照を可能にすることで、先に述べた「製品ライフサイクルの総合化」を図ることができる。例えば、上流工程の設計がある程度固まった時点で、下流工程での設計を先行させたり、保守や営業などの部門からの意見を上流の設計に反映させたりできる、というメリットが生まれる。

一方、各工程での設計内容は、最終的に製品全体として一貫性を持つ必要がある。従って、各工

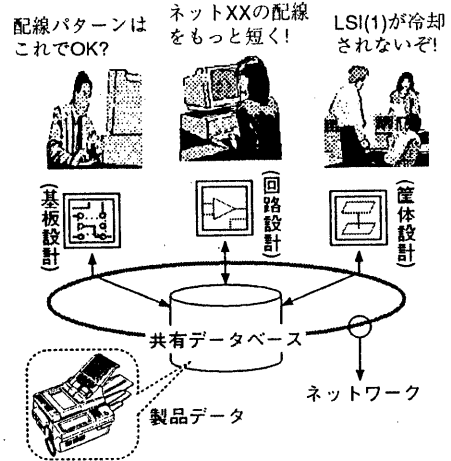


図1: CEの一般的な実現イメージ

程からの設計対象に対する様々な要求(仕様を含む)を、開発者間のレビューを通じて調整しなければならない。例えば、図1では、計算機の基板・回路・筐体の各設計に関して、各々の要求を調整している例を示している。

従来の並行ではない設計プロセスでもレビューによる調整は必要であったが、特にCEにおいて調整のためのコミュニケーションを円滑化するには、少なくとも以下のような固有の課題を解決しなければならない。

1) 非定型性: CEのレビューでは、事務系のワークフローのように定型なルーティングや作業役割を予め与えることが出来ない。例えば、常に開発者全員でレビューするのはCEの場合現実的ではないので、要求された変更ごとに、影響を受ける設計対象の作成担当者でレビューするのが妥当である。当然、それら担当者間でのレビューの進め方も実際に変更が要求されるまで確定しない。

2) 非同期性: CEの場合、各工程の開発者が設計に対する要求を随時出せるので、それらに対するレビューも非同期に発生/収束し、並行して続けられる。その結果、開発者が個別にレビュー過程を追跡することが難しくなるので、例えば自分の関与すべきレビューを他のレビューに紛れて見落とししたり忘れたりする可能性が高くなる。

3) 集約可能性: 一つの変更要求に対して寄せられたレビュー結果は、最終的には取りまとめてリ

ポジトリの内容に反映される。従って、レビューの履歴は、どの工程から見ても、対応する設計対象と密接に結び付いていなければならない。また、レビューの経緯を開発者に対して分かりやすく整理することも重要である。

4) 開放性： 各工程で設計対象を参照するツールは均一ではない。例えば同じ設計対象について、製造部門では部品形状をCADツールで、経理部門では部品価格を表計算ソフトなどで、各々参照するのが常である。従って、レビューの間にも、設計変更などの要求を多様なツールや工程の関心事に適合させて提示する必要がある。

3 CE-Notifier のアプローチ

本稿で提案する“CE-Notifier”は、前節で述べた4つの課題に対して次の3つの点からアプローチする。

レビュー経路の絞り込み： ある設計への要求を調整すべき開発者の集合が与えられた時、そのレビューにおいて各開発者が持つ意思決定の影響の大きさは均等ではない。例えば図1の基板設計の例で、各LSIの配置変更をレビューする際、最も配線数の多いLSIの配置が、全体の配置の決定に最も大きく影響すると考えられる。

この影響度を無視して、例えば各開発者のメッセージを全員に一斉配布するような方式(ex. メーリングリスト、在席TV会議)によるレビューでは、かえってコミュニケーションの無駄な後戻りを招く。CE-Notifierでは、影響度を考慮したレビュー経路を動的に絞り込むことで、元来非定型なレビューを効率的に進めることを実現する。

レビュー履歴の自動構造化： CEにおけるレビューが非同期・並行に進むのは必然であるから、開発者の見落としや忘却を防ぐには、交換されるメッセージ群を積極的に収集して、進行中のレビューの経緯を随時参照できるのがよいと考えられる。

その際、収集したメッセージのうち開発者の関心のある部分だけをピックアップして見せることが、最終的な意思決定の負担を軽くする上でも重要である。CE-Notifierでは、設計対象に要求を加えた箇所への参照をレビューメッセージに記載して、特定の対象(すなわちレビューの論点)ごとにレビュー履歴を構造化することで、担当者が経緯を把握しやすいUIを実現する。

統合的な設計モデルの利用： ツール毎に異なる設計データ表現を保ったままでは、互いのレビューが設計対象のどの部分に対応するかを交換することが出来ない。また、ツール相互でデータ表現交換仕様を定めるとしても、ツールの組ごとに異なる変換仕様が必要となり、効率が悪い。

CE-Notifierでは、多様な設計データ表現を同時に包含できるような統合的な設計モデル表現(具体的にはオブジェクト指向によるモデリング)を採用する。そして、設計対象と、それに対する開発者の要求を全てこの設計モデルの上で表現し、任意の設計ツールとの間でレビュー対象をやりとりすることを実現する。

4 システム概要

図2にCE-Notifierのシステム構成を示す。全体としては、各種の作業ツールを用いる開発者が一つのリポジトリを共有する形式になっている。

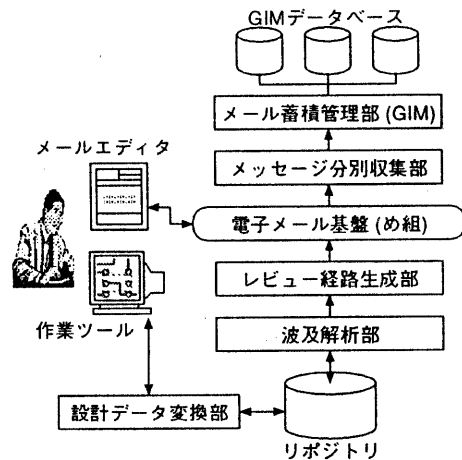


図2: CE-Notifierのシステム構成

4.1 リポジトリ

前章で述べた統合的な設計モデルを管理するために、オブジェクト指向データベースを利用する。すなわち、設計モデルの構成要素(ex. 機械設計の部品、ソフトウェアの機能モジュール)をリポジトリ内のオブジェクトで表現し、構成要素間の関係/構造をオブジェクト属性間の参照リンクや制約で与える。設計モデルの内容は、設計データ変換部において変換規則を適用することにより、各開発者の作業ツール固有の設計データと相互に

交換される。設計分野/対象ごとの設計モデルや交換規則の構成については5.2節で考察する。

リポジトリは、基本的な版管理やアクセス管理の機能の他に、オブジェクト間の参照や制約を追跡することで、あるオブジェクト属性に対する変更から、その変更による影響が波及するオブジェクトの集合を同定する機能を提供するものとする。このような機能は、例えばアクティブデータベース [3] などの技術で実現可能である。

開発者が設計に対する要求を発行する場合、各々の作業ツール上で要求箇所を指定する。要求箇所は、作業ツールごとの設計データに対してではなく、その内容を交換してできた設計モデルのオブジェクト属性への指定として表現する。更に、開発者が設計の変更を要求する場合、作業ツール上で要求箇所を直接変更し、設計モデル表現の変更箇所とその差分とをリポジトリ内に蓄積する(図3)。変更を伴わないコメントはメールエディタを用いてメール内に書き込む。

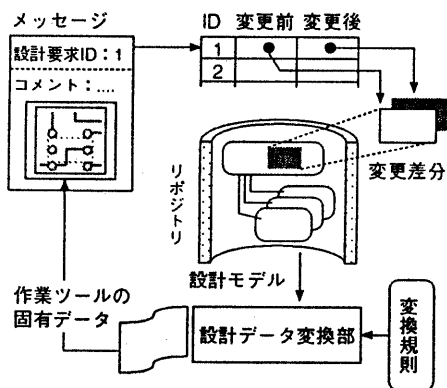


図3: リポジトリにおける設計モデル/要求の表現

4.2 コミュニケーション手段

開発者が設計要求を発行した後、その要求箇所の変更によって影響が波及するオブジェクトの集合を波及解析部で同定して、各々の作成担当者に電子メールによって通知する。

その際、通知用メールを受信した任意の開発者が元の要求を参照できるように、先の設計モデル表現内の要求箇所と、受信者がレビューすべきオブジェクト (=影響が波及するオブジェクトのうち受信者が作成したもの) とを参照する情報を電子メール内に記載する。例えば図3の例では、リポジトリ内に生成される設計要求ごとに一意に割り

当てられるIDを、発信する電子メールのヘッダ行に割り当てている。

CE-Notifierが要求通知を含めてレビューに用いる電子メールには、当研究所で開発したワークフロー向け電子メール「め組」[6]を利用する。め組は、各作業者に与えられたワークフロー制御規則によって、メールの受発信などのイベント時にメールの内容を操作できるので、次節に述べるレビュー経路の生成に伴って、各メールの配送先を動的に与えることができる。

4.3 レビュー経路の動的生成・評価

レビュー経路生成部は、前章の要求を通知する前に、影響が波及するオブジェクトのレビュー経路(一部同報を含んでも良い)を全て数え上げ、それらをオブジェクト属性間の制約などに基づいて評価する。

評価のポリシーは「各レビューでの意思決定が後戻りする可能性を軽減する」ことである。具体的には以下の2点を挙げる。

1. 設計対象全体におけるオブジェクトの変更コストを各々に与え、コストの高いものから先にレビューして確定していく。
2. 影響を受けるオブジェクト間で一方の属性値が他方に依存して決まる時、依存される側を先にレビューする。

そして、各経路がこのポリシーに違反した場合のペナルティを重みづけし、最もペナルティの少ない経路を選択する。

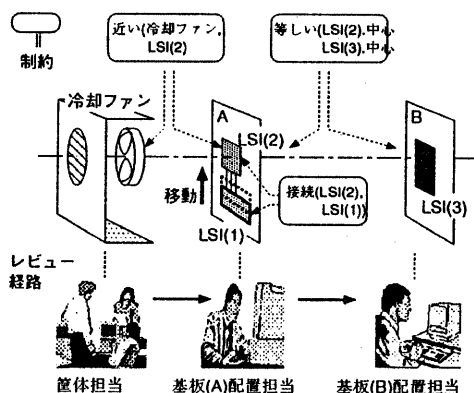


図4: レビュー経路の評価の例(部品配置について)

例えば図4の基板/筐体設計における部品配置の変更で、筐体の冷却ファンの配置変更コスト(筐

体を成形し直した場合のシミュレーション等による)がLSI(2)、(3)の配置変更コストより高い場合は、先にレビューを促す。また、LSI(2)の位置が冷却ファンの位置に依存して決まるとすると、基板(A)→基板(B)の順で配置をレビューすることになる。

選択されたレビュー経路は、最終的に各々のオブジェクトの作成者に対するめ組メールのワークフロー制御規則に変換される。例えば上の例で、筐体設計の担当者には、

```
if (LSI(1)の配置変更要求のメールを受信)
  then (次の発信先を(基板(A)及び(B)の
        配置担当者)に書き換え)
```

という規則が与えられる。

4.4 レビュー履歴の自動分別

レビューに用いられる電子メールには、4.2節で述べたように、CE-Notifierが最初に要求通知を出す際に与えた要求箇所の特定情報が記載されている。これを用いて、特定の要求箇所に関するメールを分別収集するために、当研究部で開発中のめ組メールの蓄積管理ツールである「GIM」[7]を用いる。

GIMは、め組で構成されるワークフローメッセージの蓄積ツールであり、特定のメールを収集・管理する共有データベース領域を利用者が自由に設定できるのが特徴である。CE-Notifierで利用する場合、リポジトリ内に新たな設計要求(特定のためのIDをID000とする)が追加された時点で、メッセージ分別収集部がGIMのデータベース領域(領域名を同じくID000とする)を一つ生成し、

```
if (ID000を含むレビュー用メールが着信)
  then (GIMの領域ID000に格納)
```

というめ組のワークフロー制御規則を各開発者に与える。このようにして、開発者が受信したメールは特定の設計要求ごとに履歴化される(図5)。

GIMの表示画面は、電子掲示板のように蓄積されたメールのリストや内容を各開発者に表示する。分別した結果は、例えばフォルダのアイコンなどでまとめて表示され、開発者は自分に関心のあるフォルダに格納されたレビュー履歴を参照することができる。更に、メール内の設計要求を参照して、リポジトリからレビュー対象となった設計データを各自の作業ツールで表示したり、同じ設計要求箇所に対する複数の設計案のメールから

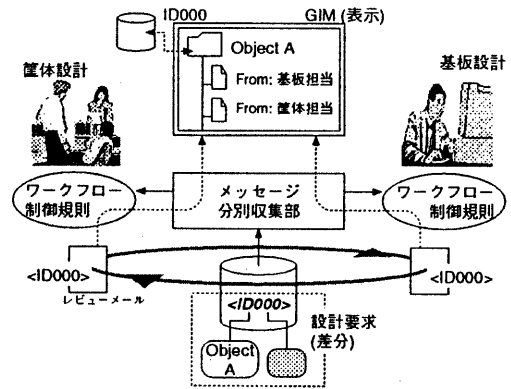


図5: レビューの分別収集の例

一つを選択して、リポジトリ内の設計モデルに反映(更新)させる等の操作を行なう。

5 議論

5.1 従来システムとの比較

これまで提案されてきた共有データベースを中心とする共同開発支援ツールのうち、例えば各種の製品データ管理(PDM)システム[2]では、多様な設計ツールからのデータを一元的に管理してはいるが、コミュニケーション手段と連動していないので、調整を始めるタイミングやレビューの履歴などを全て開発者自身が管理しなければならない。

またアクティブデータベース[3]では、リポジトリの内容更新に伴って起動される規則の副作用として、例えば関連する開発者に電子メールなどで変更通知を送ることができる。しかし、通知を送った後のレビューの実施順序や、それらを収束させる経緯にまでは関与していない。

更にハイパーメディアの応用として、設計データ(ドキュメント)の特定の箇所に変更要求やコメントをリンクし、同時に、リンクを付加したことを通知するものも存在する[4][5]。しかし、設計データ全体に渡るリンクの付加状況を大局的に把握できないので、レビューの見落としや発散が起きるという問題がある。

5.2 開放性の保証

4.1節で、オブジェクト指向の設計モデル表現について述べたが、特定のCEプロジェクトごとに、モデルを構成するオブジェクトのクラスや各

作業ツールとの表現変換規則をアドホックに作るのは容易ではない。一つの解決方法は、従来異種設計ツール間の設計データ交換標準であるSTEP[8]やCDIF[9]等における中間データを設計モデル表現に流用することである。

例えばSTEPは、オブジェクト指向の仕様記述言語“EXPRESS”によって、3次元機械系などの設計モデルに対する標準的なオブジェクトクラスを厳密に定義している。また、表現変換規則も同じEXPRESS言語を用いて定義できるが、これは各作業ツールをSTEP準拠とするためのツール提供者側の作業となる。従って、これら標準に準拠した任意の作業ツールをCE-Notifierの作業環境の中で用いれば良いことになる。

5.3 レビュー経路評価

4.3節で述べたレビュー経路の評価方式は、設計対象の内部構造に基づいていた。しかし評価の目的を「レビューを実施することによる開発者の業務負荷を減らす」という観点で考えると、例えば他の事務的業務も含めた各開発者のワークロードを総合的に評価すること[10]も有効であると考えられる。

一方、こうした経路評価に関係なく調整の過程を共有できるというのがCE本来の利点であるので、あまり厳密にレビュー経路を守らせるのも却ってその利点を潰すことになり兼ねない。例えば、経路外の開発者に自主的にメールを転送して参考意見を得るようなあいまいさを許しつつ管理するのが必要と考えられる。

5.4 レビュー履歴の構造化

4.4節で、特定の設計要求ごとにレビューメッセージを分別収集すると述べた。しかし、分別したレビュー履歴単位でも、例えば継続中のレビューと収束したものの区別を適切に管理しないと、結局どの履歴を参照すべきか画面上で分かりにくくなってしまう。これについては、最終的に一つのレビューに関する意思決定を下した時点で、対応する履歴のアイコン表示を変えるなどのUIが必要である。

また、蓄積したレビュー履歴内部で、各メッセージ間の関連を視覚的に構造化するツール(例えばgIBIS[11]など)を適用することで、意思決定を更に支援することも考えられるが、これは今後の課題とする。

6 まとめ・今後の課題

並行設計(CE)において各工程からの設計要求を調整する過程を支援する“CE-Notifier”の概要を述べた。主な特徴は、1) 要求が発生する毎に関連する開発者間で適切なレビュー経路を生成すること 2) レビュー対象箇所ごとにメッセージを分別収集し、分かりやすく表示すること 3) 統合的な設計モデルとのデータ交換により、異なる作業ツールからレビューに参加することの3点である。

今後は、試作システムを作成して、提案したアプローチの有効性を評価すると共に、具体的な並行設計の現場を想定して適用例を検討する。

謝辞

適切な助言・指導を頂いた宮井均、宮下敏昭両氏を始めとするNEC 関西C&C研究所の諸氏に感謝します。

参考文献

- [1] Institute for Defence: Analysis Report R-338, 1986
- [2] 設計環境に革命をもたらすPDM, 日経CG 1992年10月号, pp.70-85, 1992
- [3] 石川: アクティブデータベース, 情報処理 Vol.35 No.2, pp.120-129, 1994
- [4] 村水ほか: ハイパーメディアに基づく共同文書作成環境 MuHyme, 情処論文誌 Vol.34 No.6, pp.1395-1405, 1993
- [5] M.D.P.Leland et al.: Collaborative Document Production Using Quilt, Proc. of CSCW '88, pp.206-215, 1988
- [6] 垂水ほか: “GG”におけるワークフロー設計支援方式, 情処研報 93-GW-4-7, 1993
- [7] 吉府ほか: ワークフローとデータベースの相互連携システム, 情処研報 95-GW-9-23, 1995
- [8] Industrial Automation Systems — Product Data Representation and Exchange — Part I: Overview and Fundamental Principles, ISO DIS 10303-1, 1992
- [9] 篠木ほか: CDIF — CASEデータ交換形式, コンピュータソフトウェア Vol.10 No.2, pp.13-25, 1993
- [10] 垂水ほか: ワークフローの組織的最適化方式の提案, 情処研報 95-GW-9-22, 1995
- [11] Conklin, J. et al.: gIBIS: A Hypertext Tool for Exploratory Policy Discussion, Proc. of CSCW '88, pp.140-152, 1988