

# 設計品質の工程での作り込みを可能にする設計プロセス可視化 ソフトウェアの研究

(体系的 RDC モデルを利用した設計プロセスの動的な記録と共有)

モデリングテクノロジー 中沢俊彦\*

## 1. はじめに

完成品検査のみによって製品の品質を保証することの困難さはかねてより認識されてきた。そのため、製造現場においては工程で品質を作り込むという考え方が浸透し、各工程において製造条件の管理と品質特性の確認作業が徹底して行われ、製造プロセス起因の品質問題の未然防止に寄与している。特に 5M と呼ばれる 5 つの特性を管理することが重要であると言われている。すなわち、材料・部品 (material)、設備・機械 (machine)、作業員 (man)、作業方法 (method)、検査・測定 (measurement) の変化とバラツキをデータによって可視化し適切に管理することが製造品質の工程内での作り込みの根幹を成している。

製造物の品質は製造品質と設計品質に大別することができるが、上述のように製造品質が工程内での品質特性の管理に重点が置かれているのに対し、設計品質の管理は最終成果物である図面や CAD モデル、中間生成物である数値計算モデルなどに対するレビューやテストなどの検証に依存している。これは、設計プロセスが設計者の頭の中で行われる思考プロセスであり、それゆえプロセスに内在する特性の可視化や計測が難しく、設

計品質の工程内での作り込みが実行され難い原因となっている。

## 2. 既存研究と課題

設計プロセスを可視化する方法として実際の設計現場に普及しているのが品質機能展開 (以下、QFD) である[1]。QFD は顧客要求から要求品質、さらに品質特性へと展開することによって設計プロセスの論理展開の過程を可視化していく手法である。QFD は構造がある程度定まった既存製品の改良や流用設計のプロセスを分析するには優れた方法である一方、①分析に多くの工数を要する。②新しい発想や複数の選択肢、試行錯誤を伴う新規製品の設計プロセスの分析には適用しづらい、など設計品質を工程で作り込むことを目的とした恒常的なプロセス可視化手段としては運用面における課題がある。また、他にも設計プロセスの可視化手法は複数提案されてはいるものの、実験室レベルでの適用に留まっていて広く普及していないのが現状である。一般的に言えることは、複雑な設計プロセスを捉えようとするこれらの手法は設計プロセスを構成する諸要素とそれらの厳密な関連を整理し記述することを目指しているため、データ入力の手間が大きくなりすぎて日々の設計業務内での活用が難しいという側面がある。このことが QFD を始めとする既存の設計プロセスの可視化手法の適用が必須の設計品質管理工程とならない要因になっていると考えられる。

プロセスの可視化は品質を工程で作り込むため

---

A study of software to record a design process that allows to fabricate the design quality within the process.

\*NAKAZAWA Toshihiko (Modeling Technologies)

の前提条件であるが、設計プロセスの可視化が進まない設計品質の工程での作り込みのための諸策をプロセス内で実行することができない。そこで本稿では多くの試行錯誤を伴う新製品の設計、既存製品の改良設計、流用設計を問わずに適用可能な設計プロセス可視化の新しいアプローチを提案する。

### 3. 解決策の提案

設計プロセス可視化の方法として体系的 RDC モデルがある[2]。この方法は、設計プロセスを構想設計フェーズと詳細設計フェーズに大別し、各フェーズで要求定義作業と設計定義作業、およびそれぞれの作業に対する確認作業とそれらの作業によって作成される設計情報を洗い出す設計プロセス分析手法である (図1)。

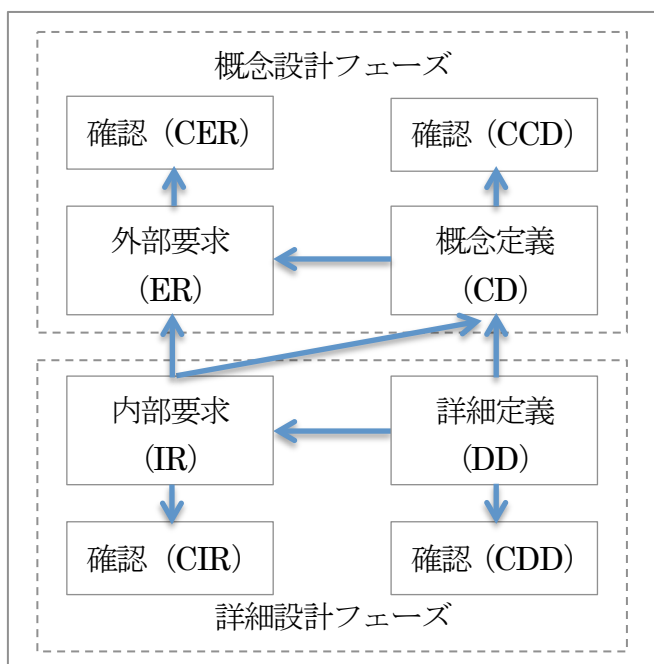


図1：体系的 RDC モデルの設計情報種別 (矢印は設計情報間の依存の方向を示す)

体系的 RDC モデルも上述の既存設計プロセス可視化手法と同様にデータ入力の手間が大きく、また過去に実施された設計プロセスを整理して記述することには有効な方法であるものの、現在進行している設計作業への動的な適用方法は提案されていなかった。体系的 RDC モデルの考え方に

基づいて、多くの作業要素と複雑な作業間関係により構成される設計プロセス内で作成される設計情報を動的に記録していくためには、そのために適切に設計されたソフトウェアが必要である。そこで本稿では、体系的 RDC モデルに基づき、設計プロセス内で作成される設計情報を動的に記録するためのソフトウェアのユーザーインターフェース (以下、UI) はどうあるべきかの検討とその検討結果を実装したソフトウェアの試作を行ったので、その状況を報告する。

UI を設計するにあたって、新規の設計プロセスが持つ以下の動的な特徴に適応し、プロセス内で作成される設計情報の記録が実現できることをソフトウェアの機能目標とした。

- ある作業が直前の作業の結果によって発生し、事前に存在が予測できないことがある。
- より良い設計解を探索するために一時的にプロセスが分岐し複数の設計解が作成されることがある。
- 設計情報の作成が時間的に離れた何回かの工程に分解されて行われることがある。
- 作業の循環 (プロセスの後戻り) があり、一旦作成された設計情報が削除または編集されることがある。

設計プロセス内を構成する作業とその作業によって作成される設計情報の項目数は比較的簡単な設計作業においても数百に及ぶ。さらにその実施順序は上流から下流へと一方向のフローチャート状に並ぶのではなく、さながら進行方向を探りながら連結していく神経細胞のごとく、分岐と後戻りを繰り返しながら増殖する複雑なネットワーク構造になる。フローチャート書式でこのプロセスを動的に記述しようとする、要素ブロックの頻繁な置き換えと関連の引き直しが必要になり、プロセスの記述作業自体に多大な作業工数が発生してしまう。この記述作業を設計者が行おうとすると、設計者は設計作業そのものと設計プロセスフローチャートの作成と編集という2つの作業を同時平行で進めることになり、設計作業そのものへ

の注力が削がれてしまう。また、作業効果が実感できないと同時平行でプロセスをフローチャート化しようというモチベーションも生まれにくい。これらのことから、以下の非機能要求の存在を示唆することができ、これらを満足するように UI の設計を行うことも重要と考えられる。

- 設計作業そのものを阻害しない。
- 設計プロセスを記録することの効果を実感しながらソフトウェアを利用できる。

以上の機能要求と非機能要求を満たすソフトウェア UI のコンセプトとして、設計作業の結果として作成される設計情報をそれらの関係を意識することなく、ひとまず発生順に時系列に記録してしまうタイムライン方式を採用することにした。この時、無分別に情報を記録するのではなく、体系的 RDC モデルの作業分類に従って記録する設計情報を分類するための選択メニューを用意することにより、記録しようとしている作業の種類を申告しながら設計情報の記録が行えるようにした (図2)。また、確認情報については確認項目の発生と確認作業の結果作成される情報の発生には時間差があるため、これらは個別に入力ができるようにした。



図2： タイムライン入力時の選択メニュー

図1に示した体系的 RDC モデルの作業要素分類によって類別された各設計情報は体系的な設計思考過程に沿った依存関係を持っている。個々の設計情報の入力時にはこの依存関係に従って、入力済みの設計情報の中から依存関係にある設計情報の候補を表示することにより、考慮すべき情報の把握を支援しつつ新たな設計情報の記録ができるようにした。例えば、概念定義作業により作成された設計情報の入力時には既に入力済みの外部

要求認識作業により得られた外部要求リストが表示され、設計者はどの要求を考慮して設計情報を作成したのかを記録できるようにした。この関連は後に新たな外部要求が記録された場合には更新することができる。この工夫の結果、タイムライン上に記録された概念定義の外部要求への依存関係が入力順序に関わらず記録される (図3)。

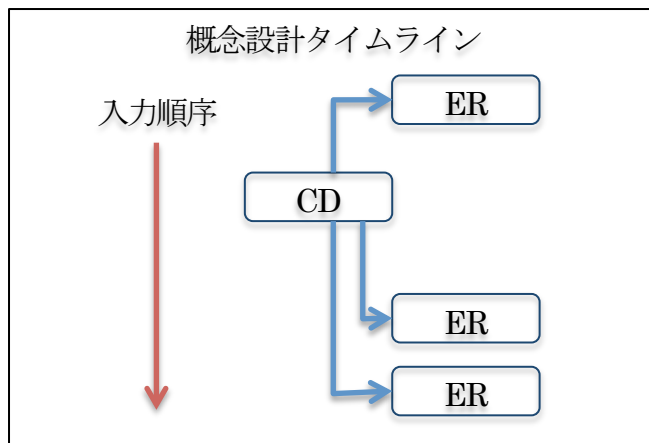


図3： 概念定義と外部要求の関連

また、確認作業は記録済みの特定の設計情報に対して実施されるため、確認対象となる設計情報を指定することにより確認情報の記録メニューが表示されるようにした。確認作業の結果作成される情報は確認対象となる設計情報への依存性が高いので、入力日時に関わらず確認対象設計情報の傍に表示されるようにした (図4)。

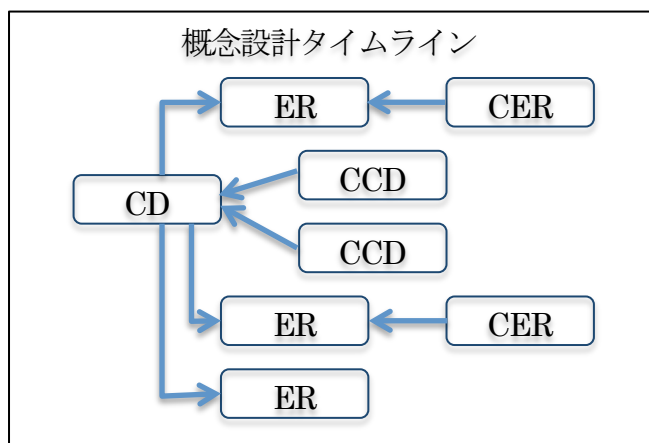


図4： 確認情報の入力

タイムラインへの記録した設計情報は編集可能であり、改訂によって情報を徐々に熟成させてい

くことができる一方、同一分類の情報をいくつでも記録できるため、設計案や作業途中の設計情報も保存しておくこともできるようにした。

詳細設計フェーズのタイムラインは概念設計タイムライン内に記録された概念定義情報より作成するようにした。すなわち一つ一つの詳細設計タイムラインは一つ概念定義に依存しており、これにより詳細設計が行われているコンセプトを容易に識別できる。複数のコンセプト案が提案されている場合には、詳細設計フェーズに進んだコンセプトを識別することにより、どのコンセプトが選択されたかを認識することができる(図5)。

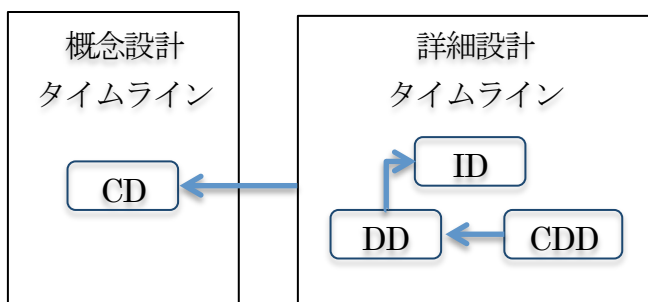


図5： 詳細設計タイムラインの依存関係

以上の工夫により、表面的には設計作業によって作成された設計情報の記録場所として利用し、利用することによって徐々に設計情報と設計情報間の関連が記録されていくソフトウェアが実現された。

#### 4. 今後

現在、本稿で説明した考え方に基づくソフトウェアの実装が進められており、今後は実際の設計現場での利用を通じた有効性の検証に移って行く。

複数の設計者によるオープンな共同設計作業を想定した利用環境も整備しており、この環境の構築により時間と場所の制約を受けない協調設計環境の実現が期待できる(<https://designearth.com>)。大規模なオープンソースソフトウェアの開発では開発リーダーが管理するマスターフロー上のソフトウェアコードに、他の開発者が管理する複数のブランチフロー上で開発されるコードを統合していくことによってサブシステムのメインシステム

への統合が行われる[3]。本稿で述べてきたソフトウェア UI においても、様々なサブシステムの設計に関する設計情報をどのように統合すれば一つのメインシステムとしての成立性を確保できるのか、またそのための UI はどうあるべきなのかは今後の重要な検討課題である。また、オープンソースソフトウェアの開発では複数の開発者が同じソフトウェアを再開発しないという思想に基づき、既に品質が検証されているソフトウェアコードの流用が奨励されるが、過去に記録された設計情報に対する検索機能は品質検証済みの既存の類似設計事例をその履歴を含めて速やかに探し出すのに役立つ。

設計品質を工程内で作り込むという点においては、筆者らは設計の確からしさを設計上の意思決定の状況を指標化することで定量化することを提案した[4]。このような設計情報の確からしさを数値化する仕組みを本ソフトウェアに組み込むことにより、設計品質の動的な把握が可能になる。

設計プロセスの動的な可視化が実現されれば、製造プロセスにおける5Sに相当するような設計品質に影響を及ぼす特性への理解が進み、それらを計測する方法の研究が進展することによって、より優れた設計品質特性の動的な管理手法が考案されるであろう。

#### 5. 参考文献

- [1] 吉澤 正・大藤 正・永井一志 (2004):『持続可能な 成長のための品質機能展開』,日本規格協会.
- [2] 中沢 俊彦・増田 宏 (2011)『設計品質管理のための設計プロセス計画と検証及び要求開発のための一手法に関する研究』, Journal of the Japanese Society for Quality Control 41(2),pp.81-93
- [3] <https://github.com>
- [4] 中沢 俊彦・増田 宏 (2008)『設計プロセスの確からしさの可視化に関する研究』, Design Symposium 2008