

移動体端末開発における Drawrial 適用効果 について

パナソニック MSE 株式会社
移動体端末事業部 要素開発六部 開発三課

野崎 和久

1. まえがき

近年、移動体端末はユーザの様々なニーズに応える為に多様な機能の追加が行われてきた。その結果、移動体端末に占めるソフトウェアの比率も高まりソフトウェア開発量も膨大となっている。しかし 1 モデルの開発期間は変わらない為、開発に要する工数は機能の多様化、複雑化とは反比例して、削減が求められている現状にある。

移動体端末のソフトウェア開発における設計工程では、画面遷移の設計の割合は 2 割から 3 割、仕様変更などメンテナンス作業を含めれば更に多くを占める。開発全体から見ると 1/5 程度と多くの工数を要している場合が多い。この効率化を行う為には、設計工程の改善が必要不可欠である。本稿では、移動体端末開発の設計工程を対象に、支援ツールによる開発効率の向上をめざした取組内容について紹介する。

2. 移動体端末開発における設計工程効率化の課題

今回の取り組みにおいては、移動体端末開発の設計工程における以下のような画面遷移設計時の問題に注目した。

(1) 仕様決定に時間がかかる

仕様決定時の判断材料として画面遷移図等が使用されるが、画面遷移図のみでは直感的に機能の把握が困難である。この為、仕様決定までにかかる工数が増大してしまう。

(2) 通常の描画ツールで画面作成した場合の工数圧迫

従来、設計図へ移動体端末の画面を記述する場合、一般的な描画ツール(以下、描画ツールという)を使用していたが、汎用ツールである為、画面の作成時に手間がかかり工数増加の要因であった。又、画面内で共通するパーツの再利用性も低く生産性低下の原因であった。

(3) 画面仕様変更時の設計図への反映

描画ツールでは、背景などの画面パーツに画面仕様の変更が生じた場合、そのパーツが使用されるすべての画面に対して変更作業が発生する為、多くの工数を必要とする。

(4) 設計時の抜け漏れによる仕様変更の発生

移動体端末の開発では、画面仕様設計の際に画面遷移図を用いる為、必要な遷移を見落とすなど、抜けや漏れが発生しやすい。

(5) 開発工程のつながり

描画ツールで作成した場合、画面遷移図は設計書の範囲で収まってしまう。開発全体を効率化する為には、設計工程での成果物を次の工程へとスムーズに引き継ぐ事が重要な課題となってくる。

3. 画面遷移図作成支援ツール「Drawrial」による設計手法の提唱

以下に述べる機能により、設計支援ツールである「Drawrial」を開発工程に適用させることで、画面遷移設計における従来の課題を克服し、仕様の検討から設計書の作成までの効率向上が可能か検討を行った。

(1) 仕様決定の早期化

Drawrial に実装されている設計レビュー機能を使用することで、仕様を作成

した次の瞬間から動的に動作を確認することができる。これにより移動体端末のキャリアなどへ明確な仕様が提示でき、認識合わせが容易になると考えられる。その結果、開発の早期の段階で認識のずれを軽減させ、仕様決定における工数について 20% 程度の削減を図る。

(2) 専用ツールによる画面遷移図作成工数の削減

画面遷移図作成に特化したツールを使用する事で画面遷移図作成における工数を削減する。特に次の点が通常の描画ツールよりも優れていると考えられる。

- ・ 使いやすい直感的なユーザインタフェース
- ・ 作成した画面仕様のメンテナンス性

(3) 仕様変更時における部品化による反映作業の効率化

移動体端末は、複数のパーツによって構成されて一つの画面ができています。Drawrial ではパーツの再利用性を最大限まで高める為に、各画面で共通するパーツについては、全て同一の物を使用する「部品化」という概念を導入した。これによりパーツに仕様変更があった場合でも、元のパーツを変更するだけで当該パーツを使用して作成された全ての画面に対して一括で変更が可能になり、メンテナンス作業効率の大幅な向上を図る。

(4) 画面遷移図の設計における抜け漏れ防止による品質の向上

ユーザインタフェース設計時の成果物である仕様データから画面を状態毎に管理し、状態遷移表へとコンバートする仕組みを実現する事で、システム設計への連携がスムーズに行え、設計全体の大幅な効率化につなげることが可能となる。又、ZIPC を使用して状態遷移表を作成する事で静的チェック等の検証が可能になり、従来に比べ抜けや漏れの防止と、設計時の生産性、品質の向上を図る。

(5) 各工程への引継ぎ

ユーザインタフェース設計時において

Drawrial で作成した設計データから、移動体端末における資源データを自動的に生成させる事で、開発工程における作業工数について 10 - 20% 程度削減する効果を図る。

検査工程への取組としては、設計時に作成した仕様データから検仕データを自動で生成する事により、工程間の連携を強化する。また、検査工程における作業自動化を進め、納期短縮、費用削減、品質向上を図る。

以上のことから、移動体端末 1 モデルの開発において各工程での連携が強固となり、開発全体における効率、品質の向上が期待できる(図1 参照)。

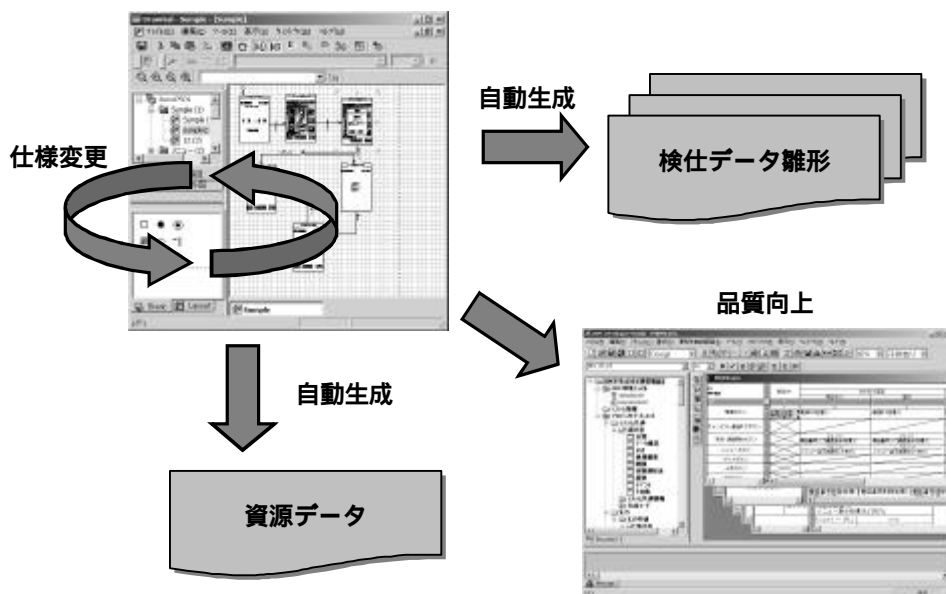


図 1 : Drawrial と ZIPC の連携概要図
(Fig.1 The cooperation outline figure of Drawrial and ZIPC)

4. 取り組みにおける効果

今回の導入目的である、工数増大の要因であった課題について以下にまとめる。

(1) 仕様決定に時間がかかる

Drawrial に実装される設計レビュー機能の導入により、ソフトウェア動作モデルの可視化が可能となり、設計当初のシステム分析やレビューには非常に有効であった。この為、開発者と顧客へ仕様を水平展開する効率が上り、およそ 1/6 の工数削減につながった。しかし、ユーザインタフェース設計工程への適用においては、上流工程の途中から既に工程が開始されている為、仕様変更の反映/更新に予想より 1.4 倍の工数がかかる結果となった。

(2) 通常の描画ツールで画面作成した場合の工数圧迫

Drawrial を導入することで、直感的な操作性と「部品化」によるパーツの再利用性の向上により画面作成時における工数を 15 ~ 25% 程度、削減できた。(表 1: 1 ファイルあたりの作成時間参照)

(3) 画面仕様変更時の設計図への反映

実際に設計工程全体での画面仕様の変更を想定した場合、設計図への反映作業における通常の描画ツールと Drawrial を比較してみた結果が表 1 (画面仕様に変更があった場合の比較) である。

Drawrial での変更工数は一つの共通部品変更あたり平均 10 画面の更新時における工数から推定した期待値である。(約 600 画面の更新時変更される部品数 = 約 60 部品)

表 1 画面仕様に変更があった場合の比較
(Table 1 Comparison when screen specification included modification)

	描画ツール	Drawrial
1 ファイル作成時間	2-3(h)	1.5-2(h)
変更数	約 600	約 60
変更工数	2.5 人月	0.25 人月以下

上記の結果から、Drawrial を使用する事で仕様変更時の設計図への変更部位反映作業を、現状の 10% 以下に削減する事が期待できる結果となった。

(4) 設計時の抜け漏れによる仕様変更の発生

システム設計工程への移行については、ユーザインタフェース設計時の成果物である設計データに対してモデリング化を行い、状態を洗い出し管理する作業に予想より大幅な工数がかかり、今回の取組では完全な連携を行う事ができない結果となった。ただし、他システムの事例においては、ZIPC を導入する事で、設計の抜けや漏れを防ぐ為のレビュー工数を 1/7 に削減できたという実績がある為、

今後の課題として連携実現手段を検討していきたい。

(5) 開発工程のつながり

Drawrial の画面レイアウトデータから直接資源データを生成する仕組みの導入により、全体の 20% 程度の資源データが自動で生成可能となった。又、設計データが実開発コードとなる為、画面ユーザインタフェースにおける仕様変更、更新を容易に行う事が可能となり大きな効果があると考えられる。

検査工程においては、雛形となる検仕データの自動生成を実現し、検仕データの作成工数をかぎりなく 0 に近づける事ができた。しかし、設計工程における画面レイアウトデータ作成工程において 1 ドット単位での正確性を求められることから、設計データの作成工数が当初

の計画に比べ大幅に工数が必要となった。この結果、検査工程全体としては、25% 程度の効率化に留まった。

5. あとがき

Drawrial は、適切な運用を行えば大変効果的なツールである。Drawrial の誕生からその進化の軌跡を眺めさせて頂いた訳だが、現状に満足せず今後も更なる飛躍を遂げていくと信じている。今後は、今回の利用資産、適用経験を生かし、Drawrial、ZIPC のポテンシャルを更に活用し、更なる生産性と品質の向上が課題と考えている。

今回は、駆け足で取組全体の概要を説明した為、各工程における詳細な内容について記述できなかったが、今後、別の機会があれば紹介させていただければと思う。