

## 大規模高信頼性プラットフォームの構築 (2)

2S-2

### - GUIアプリケーション構築のプラットフォーム

佐藤 明良\* 友部 実\* 山之内 徹\* 渡辺 正信\* 土方 雅之†

\*NEC C&C 研究所

†NEC 流通・サービス業システム開発本部

#### 1 はじめに

同じようなアプリケーションを大量に高品質に開発する必要がある時の開発プロジェクトの形式を提案する。ここでは、そのような開発対象の例として、(1)管理者不在、(2)初心者も利用、(3)大規模展開、という条件をもつシステム [1] 上に構築した、複雑で多くの画面を持つ GUI アプリケーション (GUI-AP) 構築のプラットフォームについて述べる。GUI-AP 構築の観点から高信頼性を実現することとは、「不用意な GUI 操作によってもシステムダウンや誤動作、致命的動作を出来るだけ発生しないようにすること」である。つまり、GUI-AP の品質がキーポイントとなる。

#### 2 高信頼性を実現するための要件

高信頼性を実現するためには、以下の2つの視点からの要件を考慮する必要がある:

(1) **エンドユーザの GUI 操作** — エンドユーザに不用意な GUI 操作をさせないためには、初心者にもわかりやすい GUI 操作だけに限定すればよい。つまり、アプリケーション専用に GUI 操作ボタンや GUI 部品を抽象化することが要件となる。

(2) **システム開発** — アプリケーション専用に限定された GUI 操作ボタンや GUI 部品だけを利用してシステム開発を行なうことが要件となる。また、システムにバグがあると、エンドユーザが正しく GUI 操作を行なっても誤動作やシステムダウンを引き起こすので、高品質なシステムを開発することも重要な要件となる。

#### 3 高信頼性を実現するプラットフォーム

前節で述べた要件を満足するために以下の3要素を有するプラットフォームを実現した:

(1) **専用 GUI ガイドライン** — 業務要件からアプリケーション専用の GUI 操作ボタンや GUI 部品に関する規約を設定した。この規約とは、例えば、GUI ボタ

ンの形状や色、表スクロール方法、画面クローズ方法などを含む。

(2) **専用 GUI ライブラリ** — 専用 GUI ガイドラインに則った GUI を実現するための、ボタンや表・グラフ表示部品などの GUI 部品群を実現した。

(3) **専用 GUI ビルダ** — 専用 GUI ガイドラインに則って、専用 GUI ライブラリを利用して、画面設計図から画面操作プログラムを自動合成する特定アプリケーション専用の GUI ビルダを実現した。

#### 4 GUIアプリケーション構築のプロセス

上述のプラットフォームを用いた GUI アプリケーション構築プロセスは次のようになる (図 1):

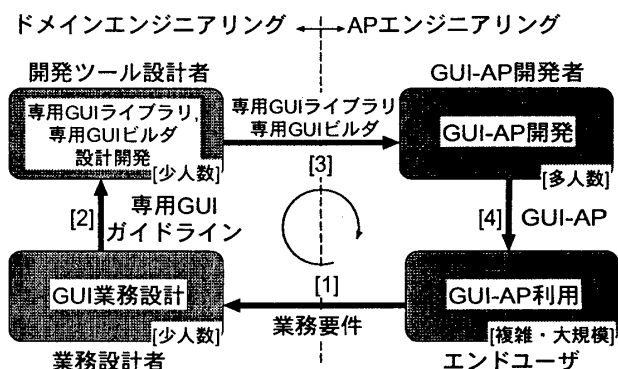


図 1: GUIアプリケーション構築のプロセス

[1] エンドユーザからの複雑な業務要件を分析し、業務設計者 (少数人数) は専用 GUI ガイドラインを作成。

[2] 専用 GUI ガイドラインを基に、開発ツール設計者 (少数人数) は専用 GUI ライブラリとビルダを実現。

[3] GUI-AP 開発者 (多数人数) は、専用 GUI ライブラリとビルダを利用してアプリケーションを開発。

[4] GUI-AP 利用者は、GUI-AP を利用評価し、必要に応じて、業務要件の修正変更を行なう。

図 1 に示すプロセスでは、複雑な業務要件による多数の GUI-AP 開発者が保証する品質を、少数人数の業務設計者および開発ツール設計者が保証するという形式になっている。つまり、「AP エンジニアリング領域に複雑に多数分散していた開発や品質保証作業を、ドメインエンジニアリング領域の少数精鋭によって集

A Platform for Highly-reliable Large-scale Systems (2) - A GUI Application Development Platform

Akiyoshi Sato, Minoru Tomobe, Toru Yamanouchi, Masanobu Watanabe and Masashi Hijikata  
NEC Corporation

中して行なうことにより、高信頼性を実現するアプローチ」である。このアプローチと GUI-AP/ プロジェクト性質との関係を図 2 に示す。次節でこのアプローチの、特に GUI ビルダへの、適用例について述べる。

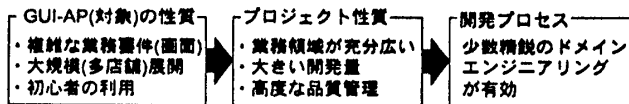


図 2: AP/ プロジェクト性質と開発プロセスの関係

### 5 専用 GUI ビルダ構築

図 1 に示す GUI アプリケーション構築のプロセスに従って、実際に、特定の業務情報システム開発に特化した、専用 GUI ビルダ構築を行なった (図 3) [2]。図 3 に示す専用 GUI ビルダは、GUI プログラム生成 (G2) だけでなく、仕様チェック (G1)、イベントディスパッチャ生成 (G3)、コールバック関数仕様ドキュメント生成 (G4) やテストデータ生成 (G5) も行なう。

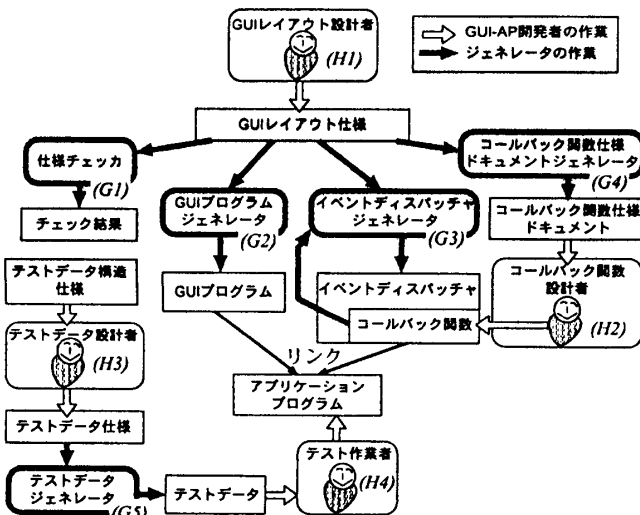


図 3: 専用 GUI ビルダの構成

### 6 評価

表 1 に、上述した専用 GUI ビルダを適用した場合の、設計開発者に関する、従来方式との比較を示す。表 1 からわかる通り、提案する GUI-AP 構築プラットフォームに則って専用開発のために、従来 GUI-AP 開発者が手動で行なっていた作業 (仕様チェック、イベントディスパッチャ開発、コールバック関数ドキュメント作成、テストデータ開発) は、少数精鋭の開発ツール設計者が開発ツールを設計開発することによって、自動化された [2]。

この事例を通して、専用 GUI ビルダによる開発自動化は、自動化による開発生産性向上だけを効果とするのではなく、「(従来の)GUI-AP 開発作業を、GUI-AP 開発者から開発ツール設計者へシフトさせる効果」もあることが実証された。実際、開発された GUI-AP

は、専用 GUI ビルダに起因するバグ数はゼロ件であり、従来と比較して品質は向上した。これは、上述した「シフト」の効果であると考えられる。

表 1: 対象ソフトウェア開発者の比較

対象ソフトウェア	従来開発方式	提案方式
(仕様チェック)	GUI-AP 開発者	開発ツール設計者 (G1)
GUI プログラム	開発ツール設計者 (G2)	開発ツール設計者 (G2)
イベント ディスパッチャ	GUI-AP 開発者	開発ツール設計者 (G3)
コールバック 関数仕様帳票	GUI-AP 開発者	開発ツール設計者 (G4)
テストデータ	GUI-AP 開発者	開発ツール設計者 (G5)
コールバック関数	GUI-AP 開発者	GUI-AP 開発者

### 7 大規模性からの要件 - 提案する GUI-AP 構築プラットフォームの適用範囲

[システム開発コスト] システムが大規模であるほど、高信頼性が要求されるが、同時に高信頼性を実現するためにかけられるコストも大きくなる。従って、専用の GUI ガイドラインを作成する専用チームを結成することや、専用 GUI ライブラリやビルダを作成するコストをかけても、システム全体ではペイするのである。

[ドメインエンジニアリング] 図 1 に示すプロセスは、AP エンジニアリングが複雑で人数数だからこそ、分離された少数精鋭のドメインエンジニアが意味を持つのである。もし、システム開発規模が単純で小規模であれば、分離せずとも、少数の精鋭だけで全てを開発可能である。

### 8 おわりに

高信頼性を実現するために、従来では多数のアプリケーションエンジニアに分散していた開発や品質保証作業を、少数精鋭の業務設計者および開発ツール設計者で集中して行なう GUI-AP 構築プラットフォームとその適応事例について述べた。今後、より幅広く本プラットフォームと構築プロセスを適用し、対象/プロジェクト性質、および、ドメインエンジニアの持つべき資質について検討・評価していく予定である。

### 参考文献

[1] 内田, 佐藤, 中川, 中嶋, 横田: 大規模高信頼性プラットフォームの構築 (1) - 高信頼性プラットフォームのアーキテクチャ, 情報処理学会第 52 回全国大会, 2S-1, 1996.  
 [2] Sato, A., Tomobe, M., Yamanouchi, T., Watanabe, M. and Hijikata, M.: Domain-Oriented Software Process Re-engineering with Software Synthesis Shell SOFTEX/S, Proc. of the 10th Knowledge-Based Software Engineering Conference, pp. 97-104, 1995.