

UI 設計ツール NINA による 組み込み機器 UI ソフトウェアの再利用性向上

豊岡 明[†]小中 裕喜[‡]三菱電機株式会社 先端技術総合研究所[†]三菱電機株式会社 先端技術総合研究所[‡]

1. はじめに

カーナビや情報家電などの組み込み機器では、機器の高機能化による UI (User Interface) の複雑化・肥大化が進んでいる。その一方で、組み込み機器の UI は機種ごとに画面デザインや操作デバイスが異なることから、UI ソフトウェアの再利用や共通化が難しいという特徴を持つ。このような理由から UI ソフトウェアの開発工数は増大を続けている。

我々はこれらの問題を解決するために、UI 設計ツール NINA^[1]を開発し、機器開発に適用している。本稿では、UI 設計ツール NINA の適用による、組み込み機器 UI ソフトウェアの再利用性の向上について述べる。

2. NINA の概要

NINA は組み込み機器向けの UI 設計ツールである。PC 上で動作し、UI の設計とシミュレータによる動作確認、実機用ソースコードの生成、ドキュメントの生成などを行える。

組み込み機器の UI は PC 上のものと異なり、一つの画面で行える操作を限定し、複数の画面を切り替えることで多くの操作を可能とするのが一般的である。NINA では、このような画面の切り替えを伴う UI に適した、SCO (State Chart Object) というモデルを使って UI を設計する。

SCO は、複数の状態(状態)と、イベントによる状態間の遷移を定義可能な UI 部品である。各状態は 1 つの画面レイアウトを持ち、ある時点での SCO の状態に対応する画面レイアウトが、その時点での SCO の外観になる。各画面レイアウトにはボタンやラベルといった基本 UI 部品のほかに、他の SCO を UI 部品として配置できる。これにより SCO の階層化が可能になる。SCO の階層化により、機器特有の UI 仕様に対応するカスタム UI 部品から、機器アプリケーションの UI に至るまで、すべて同一のモデルで設計できる。

SCO には、機器のボタン押下などのイベントに対応して実行されるイベントハンドラを定義でき

る。イベントハンドラは、JavaTM言語のサブセットをベースとするシンプルなスクリプト言語で記述する。スクリプト言語は、簡単な文字列操作や四則演算、条件分岐、繰り返し、ロジック(ユーザ定義関数)の呼び出しなどを記述できる。ユーザ型の定義やスレッドの生成などの高度な機能はもたない。

設計した SCO からは、NINA のコード生成機能により実機用の C++言語のソースコードを生成する。生成されたコードは、NINA ランタイムとともにコンパイル・リンクすることで実機の UI として動作する。NINA ランタイムは、生成コードの実行に必要なイベントの管理や、UI 画面の描画などを行うソフトウェアモジュールである。図 1 に、NINA を適用した機器のソフトウェア構成を示す。

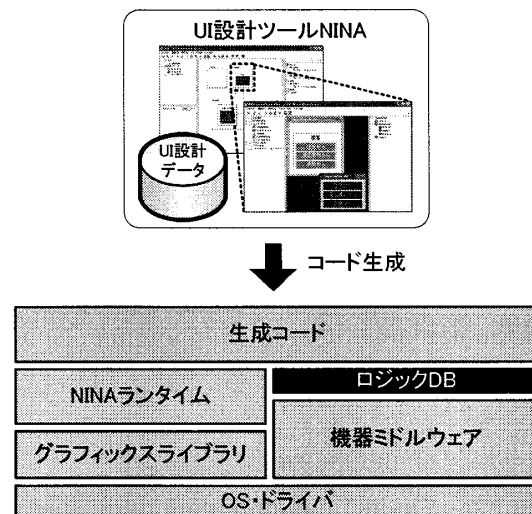


図 1 NINA 適用機器のソフトウェア構成

グラフィックスライブラリは、NINA ランタイムが UI 画面を描画するための 2D グラフィックス描画機能を提供する。機器ミドルウェアは、機器特有の機能を実現するソフトウェアモジュール群である。

ロジック DB (DataBase)は、生成コードから機器ミドルウェアの機能呼び出すためのユーザ定義のインタフェース関数群である。NINA では各インタフェース関数をロジックと呼ぶ。なお、NINA 上ではロジックのプロトタイプのみを定義し、ロジ

Reusability improvement of embedded UI software using UI design tool NINA.

[†] Akira Toyooka, Advanced Technology R&D Center, Mitsubishi Electric Corporation.

[‡] Hiroki Konaka, Advanced Technology R&D Center, Mitsubishi Electric Corporation.

ックの本体は C++言語の関数として実装する。関数内で機器ミドルウェアの API 関数を呼び出すことで、UI 画面表示に必要な情報の取得や、機器ミドルウェアへの操作要求を行う。

2.1 ロジックによる UI・非 UI の分離

従来の UI ソフトウェアでは、UI 仕様に依存した処理と、UI 仕様から比較的独立した処理が、一つのソフトウェアモジュール内に混在しており、これが UI ソフトウェアの再利用を困難にする要因となっていた。NINA では UI 部と非 UI 部を分離し、再利用性の向上を図っている。ここで UI 部は画面レイアウトや入力デバイスなどの UI 仕様に強く依存する処理を実装する部分である。UI 部は UI 仕様変更時の流用が難しいため、それ自体の開発を効率化するとともに、再利用性の高い処理を含めないようにする必要がある。非 UI 部は、UI ソフトウェアの中でも UI 仕様への依存度が低く再利用性の高い部分である。

NINA では、UI 部と非 UI 部の境界としてロジック DB を設け、UI 部を SCO でモデル化する部分、非 UI 部をロジック DB の実装部分としている。その上で、UI 部については、NINA により開発効率を改善するとともに、スクリプトの記述能力を抑えて、再利用性の高い複雑な処理がイベントハンドラに記述されることを防いでいる。また、ロジック DB という明確なインタフェースを設けることで、ロジックの UI 仕様への依存度を下げ、再利用性の向上を促す。

3. 評価

NINA 適用の効果を評価するために、NINA を適用したデジタル TV の派生機種開発における、UI ソフトウェアの再利用率を測定した。また、比較対象として、NINA 導入前に採用していた、UI ツールキットによる UI 開発手法についても、元機種のソースコードを用いて、開発経験者による再利用率の見積もりを実施した。

再利用率の測定対象には、デジタル TV において比較的複雑な UI である電子番組表を選択した。表 1 に、測定対象とした電子番組表の概要を示す。元機種と派生機種では、画面のレイアウトや表示内容、表示番組数などが異なる。その一方、基本的な画面パターンは同一である。またどちらもリモコンでの操作であるが、操作に使用するボタンの数が異なる。

表 2 に、従来方式と NINA の開発工数の比較を示す。NINA については、実際のソースコードから再利用されたコードのステップ数を計測した。従来方式については、画面レイアウトやボタン操作に依存するコードを再利用不可、それ以外のコードを再利用可とみなし、再利用可能なコードのステップ数を見積もった。

表 1 電子番組表の概要

項目	元機種	派生機種
機能	番組名, 開始時刻, 予約状況を 4 番組分表示	番組名, 開始日時を 6 番組分表示
操作デバイス	ボタン 8 個 (上下左右, 決定, 戻る, 特殊機能ボタン×2)	ボタン 6 個 (上下左右, 決定, 戻る)
画面パターン	5 画面(通常 1 画面, エラー 4 画面)	5 画面(通常 1 画面, エラー 4 画面)

表 2 再利用率の比較

		元 ステップ	再利用 ステップ	再利用 率
従来方式		2809	1641	41%
NINA	生成コード	2882	1792	62%
	ロジック DB	1851	1710	92%
	計	4733	3502	73%

表 2 から、電子番組表の UI においては、従来 4 割程度だった再利用率が NINA 適用によって 7 割以上に改善されることが確認できた。特に、機種間での再利用を目的とするロジック DB については 9 割超の再利用率を達成した。また生成コードについても、基本的な画面パターンが同一であったため、スクリプトの再利用率が高くなり、結果として 6 割の再利用率となった。

ステップ数については、NINA 適用による設計モデルの変更や、コードの自動生成、UI・非 UI の分離により、UI 部・非 UI 部全体で 1.7 倍程度に増加した。しかし、UI 部のコードについては、NINA により従来比 25%~50%程度の工数で開発可能²⁾なため、全体の開発工数は従来比 50%程度に削減できると思われる。

4. おわりに

UI 設計ツール NINA を適用した組み込み機器 UI ソフトウェアについて、派生機種開発時のソフトウェア再利用率を測定した。また、従来の UI ツールキットを用いて開発した UI ソフトウェアと比較して、再利用率が向上することを確認した。

今後、デジタル TV 以外の機器の UI についても測定を行い、評価の精度を高める予定である。

参考文献

- [1] 小中裕喜, ほか: 階層的部品定義に基づく組込用 UI 設計ツール, 組み込みソフトウェア工学シンポジウム 2002, 情報処理学会研究報告, 2002-SE-139, 7~8 (2002).
- [2] 豊岡明, ほか: 車載デジタル TV への UI 設計ツール “NINA” の適用, 三菱電機技報, Vol. 81 (No. 9), 71~74 (2007).