



# e-コクピット時代の HMI 開発環境

～ HMIの振舞い設計は状態遷移/画面遷移が重要 ～

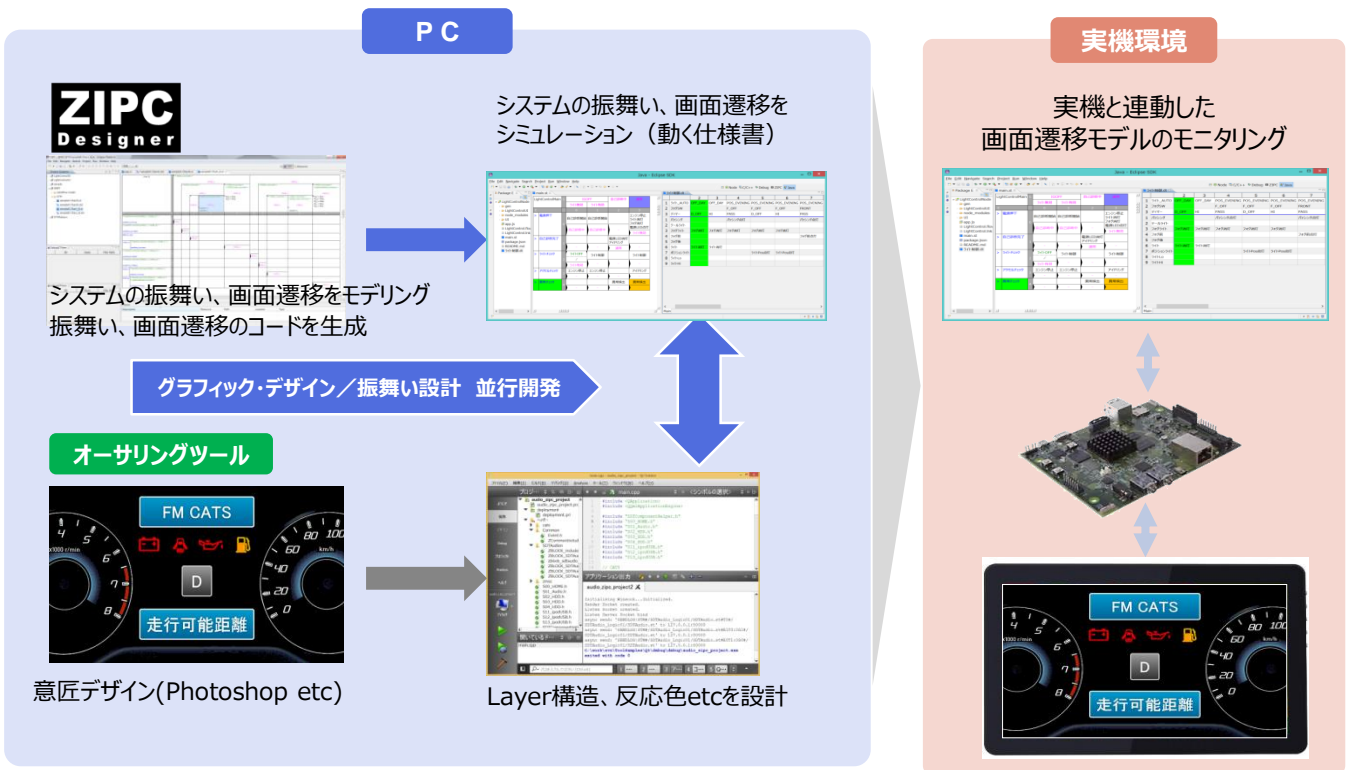
## こんな状況はありませんか？

- × デザイナー作成のグラフィックが制約される
- × 振舞いの仕様が確定しない
- × 実装後の手戻りが多く工数、コストが増大する
- × より複雑な条件でのグラフィック表示の制御が難しい

## 本 HMI 開発環境で期待される効果

- 実装開発を並行で進められ、リッチな画面を実現できます
- 動く仕様書で明確な振舞いを早期に定義、確定できます
- 振舞いのモデルからソースコードを生成し手戻りが減ります
- 状態遷移/画面遷移モデル、決定表で複雑な制御も可能です

## グラフィック・デザインと画面遷移設計を並行して開発できる



## 本環境での HMI 開発の流れ

1

「グラフィック・デザイン」と HMI の「振舞い、画面遷移」のモデリングを並行して実施できる。

2

グラフィック構造と制御モデルのコードをコンパイルし、PC で実装前に「動く仕様書」を動作確認できる。

3

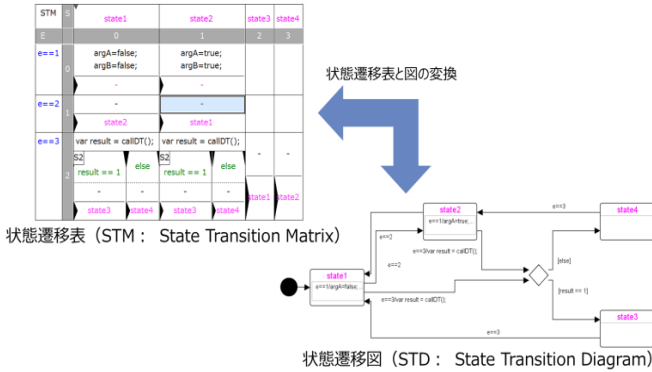
動作が正しいと確認されたコードを実装し実機確認するので手戻りが無くなり、生産性が向上する。



# e-コックピット時代の HMI 開発環境

～ HMI の振舞い設計は ZIPC Designer で！ ～

## 特長① 状態（画面）遷移設計



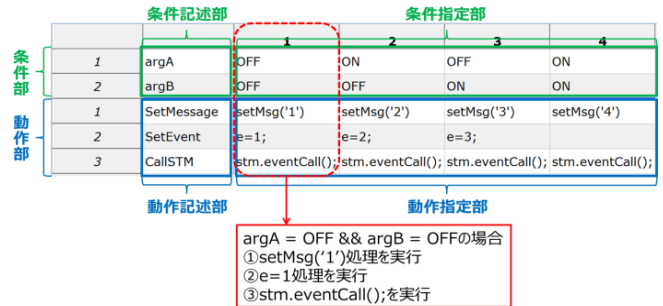
- あいまいな仕様の定義を明確に表現できます。
- 複雑な振舞いも状態と事象の組合せで網羅します。
- 状態モデルがそのまま設計書として資産化できます。



## 特長② 決定表



- ISO準拠の決定表エディタです
- 複雑な条件の組合せを整理、定義できます



## 特長③ 様々なオーサリングツールとの連携



- 主要なオーサリングツールと連携可能です

## これらの特長により HMI 開発の生産性、品質が飛躍的に向上します

