



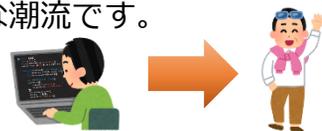
# プログラミングの民主化を促進する ローコード開発環境

知能情報学部 教授 新田 直也

## 研究の概要・特徴

### プログラミングの民主化とは？

ITの専門家が担ってきたソフトウェア開発を、一般ユーザ（**市民開発者**）でも行えるようにしようという世界的な潮流です。日本においても神戸市、東京都、大阪府など多くの自治体のアプリケーション開発において成果を上げています。



### プログラミングの民主化の背景

背景には、プログラミングをまったく、あるいはほとんどしなくてもアプリケーションを開発することができるノーコード・ローコード開発ツールの急速な発展があります。



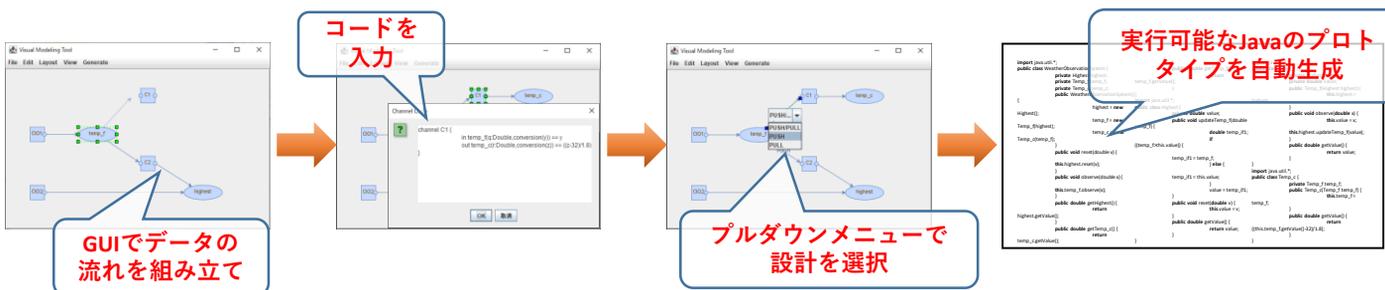
### ノーコード・ローコード開発ツールの課題

ノーコード・ローコードツールは典型的な課題には効果的ですが、一定以上の複雑な要件には対応が困難です。また、特定の分野に特化したものが多く、さらにアプリケーション開発後の保守作業が使用したツールに完全に依存してしまう（ベンダーロックイン）という問題もあります。

### 数理モデルに基づいたローコード開発環境

**新しい数理モデル**に基づいた汎用なローコード開発ツールを開発しています。GUI上で開発や設計変更ができます。

**新しい数理モデル**に基づいた汎用なローコード開発ツールを開発しています。GUI上で開発や設計変更ができます。



### 新規性・優位性

- ✓数理モデルに基づいているため**汎用**で、ある種の正しさが保証されています。
- ✓Javaのプロトタイプを自動生成できるため、**ツールに依存することなく**保守作業を行うことができます。
- ✓**GUI上で設計の選択や変更**ができます。
- ✓記述量を本質的に削減したアーキテクチャ記述言語を使った開発も可能です。

### 実用化によって期待される効果

- ・**開発段階**：数理モデルはプログラミング言語に依存していないため、他の言語（たとえばPythonやC）に対応することも可能です。
- ・**適応分野/用途**：現在は主に**Webサービスの開発**への適用を想定していますが、組み込みシステム、業務アプリ、スマホアプリなどの開発への適用も検討しています。

