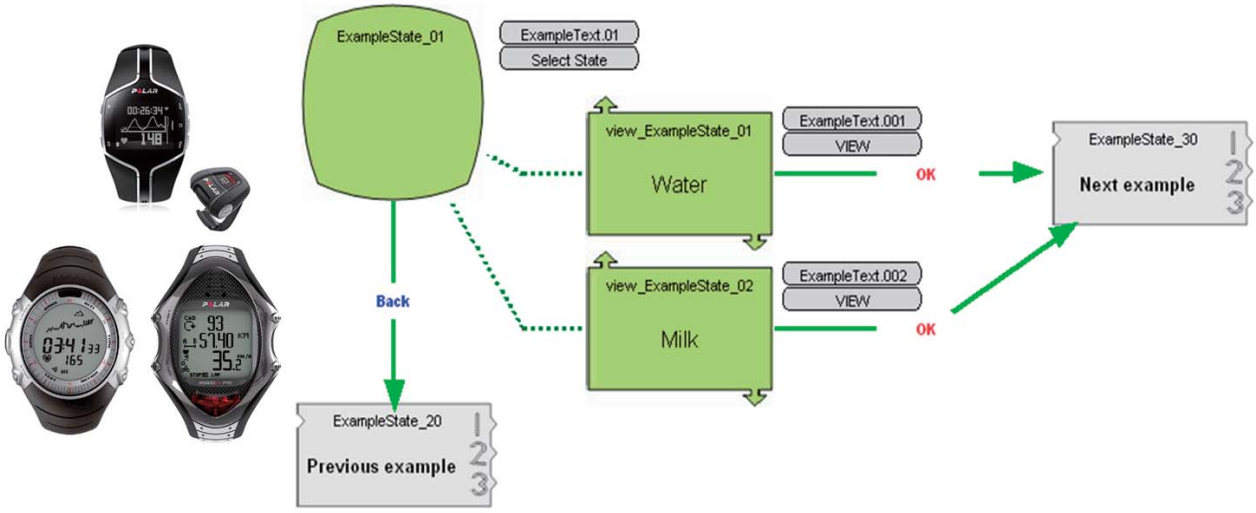
MetaEdit+

ドメインスペシフィックモデリングと自動生成



ドメインスペシフィックモデリング（Domain-Specific Modeling, DSM）は、ドメイン固有のコンセプトを用いることで、コード実装に比較して開発の抽象レベルを引き上げます。そして高い抽象レベルでモデル化された仕様をコードやドキュメントに自動変換して、開発効率や品質を飛躍的に改善します。例えばPoral社のDSM（上図）は、スポーツウォッチのコンセプトでわかりやすいモデルから、ジェネレータで完全なコードを自動生成して、開発の生産性を10倍以上改善しました。生成されるコードはマニュアル実装より高品質であることや、モデルが全ての利害関係者に理解されること、新しい開発メンバーが即戦力になることも成果として報告されています。

ライブラリ、フレームワークを活用する究極の手段 = DSM

殆どの開発者にとってフレームワークの正しいコンポーネントを探すことは簡単ではありません。考慮すべき相互依存性、排他関係、同一機能を実現する複数の選択肢などの存在は、フレームワークを正しく使いこなす為の学習に多くの時間を費やすことになります。

このような問題は、DSMで開発の抽象度を上げながらフレームワークの詳細を隠蔽することで解決します。DSMではフレームワークは利用可能なオブジェクトとなり、高いレベルのコンセプトでモデル化されたアプリケーションから、フレームワークを巧みに活かした実用的なコードが生成されます。

この自動化は、単一ドメインの要件だけを満たすDSM言語とジェネレータによって得ることができます。DSM言語に設計のルールやコンポーネント使用上の制約を持たせることで、決まりきった作業を最小限に抑え、ありがちなエラーを未然に防ぎ、フレームワークが正しく使用されることを保証すると同時に製品開発を加速します。これは熟練者の知識が全ての開発者で共有できるようになるということです。

ジェネレータは、コードとビルド用のスクリプトを同時生成させることに加えて、テストケース、各種ドキュメント、ハードウエアの割当てやデバイスの構成、Simulink等の各種ツールとの相互変換など、様々な用途に活用されます。

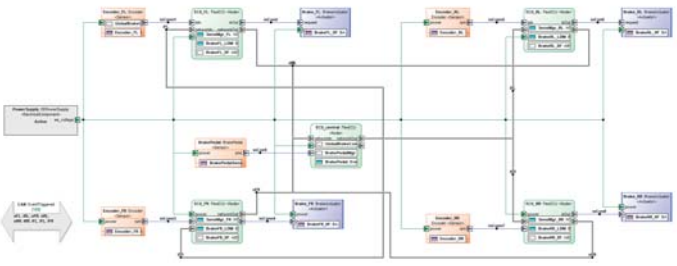
SysML、EAST-ADLなどアーキテクチャ記述言語 にも

システムズエンジニアリングへの関心が高まるなか、その取組みを手間や負担にしないためには、開発効率の改善に役立てる工夫が求められます。

例えばVolvo、Fiat、Daimler、Continental、Boschにより、MetaEdit+を用いて開発されたEAST-ADLは、AUTOSARを高い抽象レベルで補完するDSM言語で、車両レベルの機能、機能構造、要件、ソフト/ハードウエア部品、通信等の側面をカバーします。振舞いの表現にSimulinkを活用することや、プロダクトライン開発のフィーチャモデルのサポート、ISO 26262の認証作業を支援するデペンダビリティモデル、エラーモデルなど安全設計（ハザードや障害の影響範囲の分析）にも対応します。

これら要件を満たすには、DSM言語とジェネレータを開発するための専用ツールによる以下のような点での洗練が必要です。

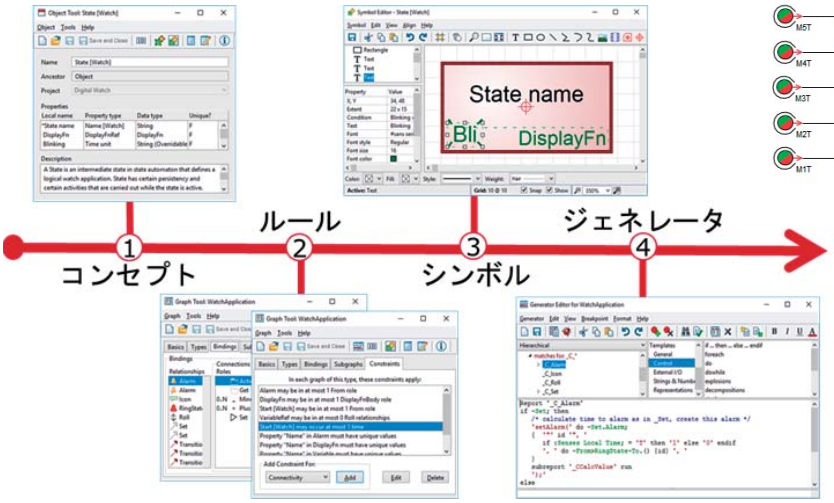
- ドメイン固有のルールや制約で間違いや曖昧さを排除
- ドメイン固有のコンセプト、表記、見た目を持たせて、モデリング作業や意思疎通を円滑化
- コード生成、モデル相互変換等でモデルを最大限に活用
- 外部ツールとの正確な連携を支えるセマンチックなルール
- ドメインの変化（仕様や業界規格）への柔軟な対応
- コラボレーション開発、大規模モデルを支える拡張性

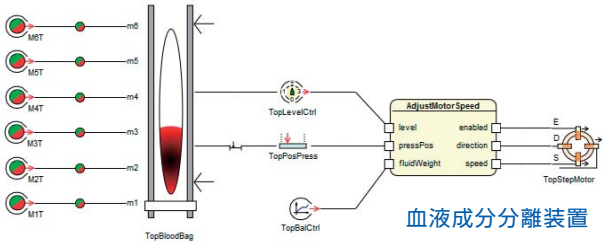


このEAST-ADLの成果は、MetaEdit+　のプロジェクトとして入手して利用することができます。

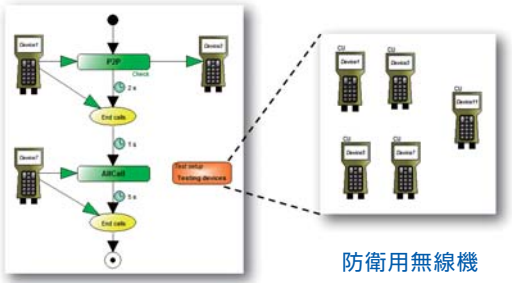
MetaCase

DSMの持続的な進化を支援できる唯一のツール





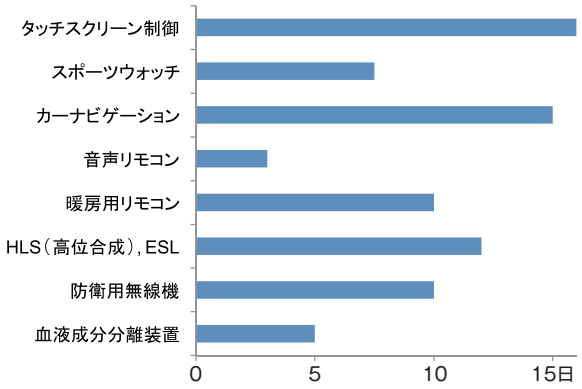
血液成分分離装置



防衛用無線機

DSMを導入するコツは、まず限定的に小さな範囲でDSM言語とジェネレータを試して、ユーザの意見を取り入れて改善を繰り返すことです。そしてリスクを最小限に抑えながら、短期間の成果で組織の賛同を得て、範囲を段階的に拡げます。また製品の持続的な変化に応じて、DSM言語とジェネレータを柔軟に変更して進化させることも重要です。

MetaEdit+ではDSM言語とジェネレータの定義が一つに統合されるので、このようなイテレーティブかつインクリメンタルな開発を支援し、一部分への変更が全ての定義と既存モデルに自動反映されます。MetaEdit+でなら通常1～2週間で開発されますが、専用ツールを用いない場合、DSMの開発に数年・数億円を要して、継続的な保守が成り立たないことが多く報告されています。



①コンセプトを定義

DSM言語の開発は、コンセプトを定めることから始まります。これは対象となるドメインに応じて、製品の部品、アーキテクチャ、製品系列の特性（プロダクトラインのバリエーション）、モデルから自動生成される成果物、などから得られます。一般に主要なコンセプトは、モデリング言語のオブジェクトとして定義され、それ以外のコンセプトはオブジェクトのプロパティ、コネクション、サブモデル、他のモデル言語上のモデルへのリンクなどです。

②ドインのルールや制約/テンプレートを用いて定義

DSM言語にドメインのルールや制約を持たせることで、モデリングの労力を軽減して、早期段階で間違いを未然に防ぐことができます。また自動生成やモデル変換のためのジェネレータの定義にも、ルールや制約が活用されます。

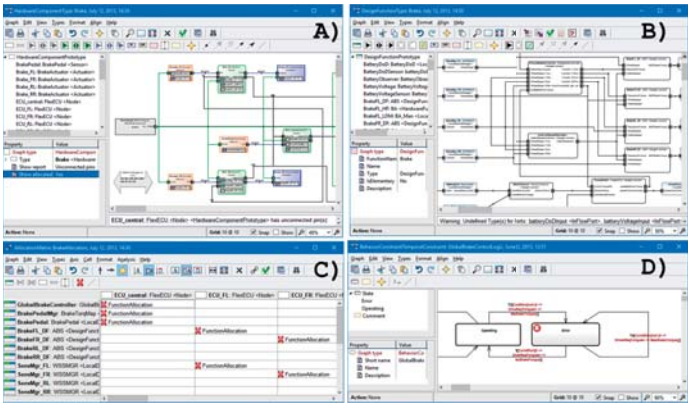
③モデルの表現や見た目を定義

モデルのコンセプト、コンセプト間の関連、ルールなどを記号化します。良い表記はドメインの描写を真似ることや、開発者間で既に馴染みのある約束事から得ることができます。それによりモデリング作業、モデルへの理解、メンテナンスが容易になります。

④ジェネレータを定義

DSMの究極の目的は、モデルを変換してソースコードやドキュメントなどの様々な成果物を自動生成することです。そしてジェネレータの開発は、モデル内のコンセプトをソースコード等の成果物にどのようにマップするかということにつきます。単純なケースでは、モデリング時に各シンボルに設定された引数を伴った、特定の決まったコードを生成します。より一般的には、シンボル間のリレーションシップや他のモデリング情報なども考慮に入れます。

【システムレベルでコラボレーション開発】



A）HW担当者2名が同じモデル図上の異なるHW部品を編集。同時にB）でシステムの論理コンポーネントをアーキテクトが定義。C）ではコード生成のために、A）とB）で編集されているHWと論理アーキテクチャ間の割当てを定義。D）では機能安全担当がB）で定義される論理コンポーネントのエラーモデルを定義。

FUJI SETSUBI

富士設備工業株式会社 電子機器事業部 www.fuji-setsu.co.jp
〒591-8025 大阪府堺市北区長曽根町1928-1 TEL.072-252-2128