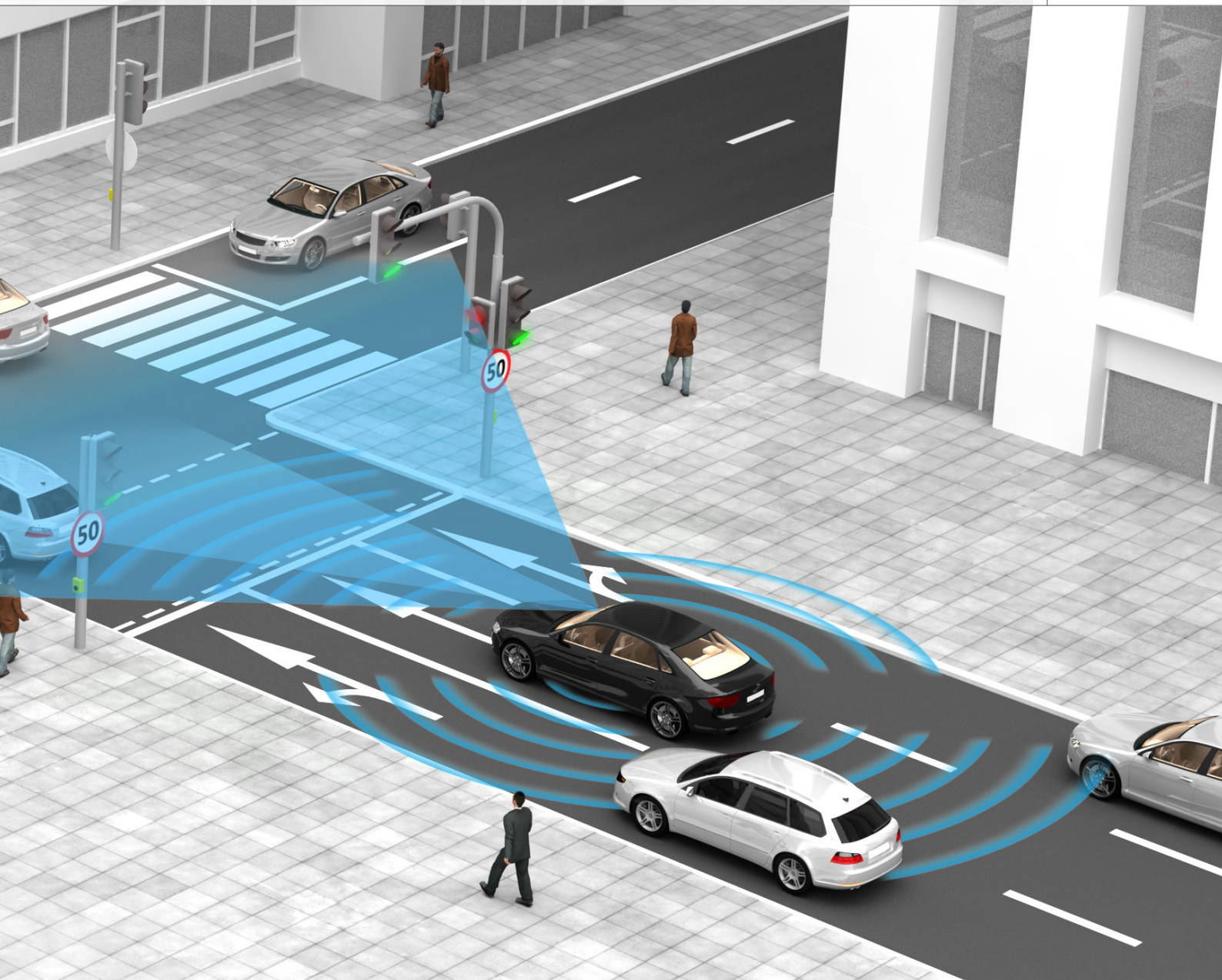


運転支援機能の一貫したモデルベース開発により、  
さまざまなプラットフォームに対応

# Developing Intelligent Assistants

Automotive Safety Technologies GmbH では、センサベースの運転支援システムを開発するプロジェクトにおいて、複雑なデータを確実に処理し分析することを目標にしています。量産コード生成ツール TargetLink を使用すれば、シームレスで効率的なワークフローを実現することができます。



センサベースの運転支援システムでは、さまざまなセンサから提供される多数のデータが融合することにより、データ量が膨大になるだけでなく、アルゴリズムやデータ構造が複雑化します。横断歩道や交差点に対応した運転支援システムの開発プロジェクトで必須となるのは、次のようなデータ処理を実現することです。

- カメラ、レーダー、レーザーといった多数の異なるセンサからのデータの融合
- 生のレーザーデータに基づくオブジェクトのビルド

- 車両や歩行者といったオブジェクトに対するカメラベースのマルチハイポセシスなトラッキング

複雑な機能の診断をモデルベースの開発環境で実行するには、大量の機能内情報（直接アクセスできない情報）を妥当性確認やテストの段階で利用できるようにする必要があります。これらの詳細な情報を処理するには、さらに多くのメモリと実行時間が必要となります。ただし、必要な機能の開発が完了すると、この機能内情報は不要になり、今度はメモリ要件の最小化や実行時間の最適化が重要になっていく

ことを考えると、問題は開発フェーズで解決しておく以外ありません。そのため、非機能範囲についてもスケラブルに作成できるソフトウェアコンポーネント（SWC）が必要になります。また、このSWCは特定のアプリケーションに合わせて柔軟にコンパイルできる必要もあります。開発フェーズでは、多くの場合、適切なターゲットプラットフォームをまだ入手できません。そのため、特に先行開発プロジェクトでは、特定のプラットフォームに依存することなくプロトタイプ機能を迅速に実装することが重要になります。開発環境であるADTF（Automotive Data and

&gt;&gt;



制御モデルから生成されたソフトウェアコンポーネント (SWC) の統合を示す概略図：SWC は、ECU ではバイナリファイル (.bin) として組み込まれるのに対し、ADTF では実行可能なライブラリ形式 (Windows の場合、.dll) で含まれています。

Time-Triggered Framework) は、この目的に適したツールであり、PC 上で機能を実行したり、機能を他のコンポーネントに接続することができます。

#### センサデータに対応した開発環境

ADTF は、車両のセンサデータをオンラインで同期的に記録し、関連する機能を提供するための開発環境です。ADTF では、記録データをオフラインで再生し、記録時間に関係なく機能を実行することもできます。また、コンポーネントを自動的にプログラミングしたり、既存のコンポーネントを ECU 上で制御することも可能です。ただし、Windows®/Linux ライブラリを作成する場合に、原則として ECU コードを生成するためのモデルと同じものを使用します。また、前述の通り、関連する大量の

機能内情報に開発段階でアクセスすることが必要になります。これは、特に機能が正しく動作しない場合に、機能を解析し、理解するためにも重要です。この場合、アクセスのしやすさを確保するために、この内部データを事前にモデルとして提供する必要があります。データをモデルとして提供すれば、開発環境内でコンパイルされたモデルをコンポーネントから呼び出す際に、内部データを解析してビジュアル表示したり、他のコンポーネントに転送することができます。

#### 実装の技術的要件

実行可能な ECU ソフトウェアの機能を実装するには、dSPACE の量産コード生成ツールである TargetLink® が役立ちます。TargetLink を使用すると、SWC 用のコー

ドを制御モデルから生成し、コントローラと開発環境の両方に統合することができます。実行した SWC の動作は、この 2 つのプラットフォーム上でトレースし、解析結果をビジュアル表示できることが必要です。このような状況は、ECU と開発環境の両ランタイム環境で発生しますが、メカニズムはそれぞれ異なります。ECU では、SWC は計測および適合プロトコル (XCP) を使用して、あらかじめ定義されたランタイム変数に外部ツールからアクセスできるようにします。一方、開発環境では、あらゆる SWC 変数は他のプログラムに転送され、必要に応じてビジュアル表示されます。

ECU とは異なり、PC ベースの環境では SWC からメモリを供給する必要がないため、SWC はデバッグ変数を何個でも追加して拡張し、デバッグ機能を十分に活用することができます。ここでの課題は、ECU コードをほとんど変更しないでおきながら、SWC を統合するための要件も考慮に入れなければならない (デバッグのためにできるだけ多くの変数を作成して使用可能にするなど) という点にあります。また、各ターゲットプラットフォーム (Linux または Windows) で統合を行うための自動化レベルを高くし、ワークフローにおける手作業での介入レベルを低く抑えることも必要です。これを解決するには、SWC モデルや SWC データベース、関連するスクリプトに変更を施すことが必要となります。

#### TargetLink による自動実装

AST 社では、この自動化を実現し、開発プロセスにおける TargetLink の統合をサポートするために、TargetLink のコールバックメカニズムを採用しました。

デバッグ変数に対応したコード生成により、目的に応じて柔軟に対応できる TargetLink Data Dictionary の仕様。

| プロパティ                 | 値                               |
|-----------------------|---------------------------------|
| Description           | "Variable class"                |
| Storage               | default                         |
| Scope                 | global                          |
| ArgClass              | <>                              |
| Volatile              | off                             |
| Const                 | off                             |
| Macro                 | off                             |
| Alias                 | off                             |
| InitAtDefinition      | off                             |
| RestartFunctionName   | "InitPredictionVariables" 初期化関数 |
| SectionName           | "                               |
| TypePrefix            | "UAS_API" コンパイラスイッチ             |
| DeclarationStatements | {}                              |
| UseName               | on                              |

「当社では、TargetLink を利用して、生成コードを開発プラットフォーム向け・ターゲットプロセッサ向け双方に切り替え可能なシームレスな自動実行ワークフローを構築しています。これにより、高性能な運転支援機能を効率的に開発できるようになりました」

Matthias Issbruecker 氏、Automotive Safety Technologies GmbH

TargetLink 固有のフック機能を使用すると、コンパイルプロセスでのユーザ固有の変更をサポートすることができます。フック機能が呼び出されると、ユーザ定義の命令が自動的に実行されます。そのため、追加の手作業による調整は必要ありません。

AST 社では、TargetLink を使用することで、追加のデバッグ変数を配列として実装することができました。この配列は、コンパイラスイッチにより、作成するか（開発環境の場合）、省略するか（ECU の場合）をコンパイルプロセスにおいて選択できます。コンパイルバリエーションを使用すれば、異なる開発プラットフォーム向けに同一のコードを開発し、コンパイルプロセスの時点でコードを目的に合った形式に変換することができます。これらのデバッグ変数配列は、さまざまな幅とデータタイプを持つことができます。AST 社では、再利用可能な汎用的なソリューションを作成し、ライブラリ形式で実装することができました。さらに、プリプロセス命令およびプレフィックスの自動挿入を使用することにより、同一の C コードを生成しながら、コンパイラスイッチによって ECU または Windows/Linux 向けの異なるランタイム環境に対応することができました。 ■

Matthias Issbruecker 氏、Mohinder Pandey 氏、Automotive Safety Technologies GmbH

## まとめ

Automotive Safety Technologies 社では、量産コード生成ツール TargetLink を使用して、開発プラットフォームとターゲットプロセッサの両方に対応したコードを生成できるようにすることで、効率的な開発環境を実現しています。同社では、コンパイラスイッチを使用して、制御モデルから自動的にプラットフォーム固有のコードを生成できます。コードは、デバッグ変数を追加して拡張することも、ECU 上で実行することも可能です。このアプローチの利点は、完全に同一な制御モデルから開発環境 ADF が供給されるため、容易に運転支援機能を開発し、ECU 向けのコードを生成し、実際のランタイム条件でコードをテストすることができる点です。

### Matthias Issbruecker 氏

交差点アシスタント機能の開発担当、Automotive Safety Technologies GmbH (ドイツ、ガイマースハイム)



### Mohinder Pandey 氏

交差点アシスタント機能の先行開発担当、Automotive Safety Technologies GmbH (ドイツ、ガイマースハイム)

