



PREvision 将来の車両イノベーションに対応する E/E 開発

～統合されたモデルベースのアプローチ～

現代の電気／電子 (E/E) アーキテクチャーの開発はかつてないほど難しいものになっています。膨大な数の開発基準を考慮しなければならないうえ、従来の車両ドメインを、運転支援や自動運転といった領域の新たな機能にリンクさせる必要もあります。そこでは車両の枠を越え、車外の「IT バックエンド」からもサービスとして提供される、まったく新しい機能が登場しています。サービス指向アーキテクチャーと高機能なドメインコンピューター、オンボード通信用 Ethernet、Car2X 通信ゲートウェイなどの導入や、安全とセキュリティに対する要求の拡大は、開発に携わるあらゆる組織にとり、広範かつ抜本的な変化を意味します。このような複雑な開発タスクを 1 つのチームとして円滑に進めていくには、多様な形で実装できる開発プラットフォームと E/E データベースが欠かせません。

現代の車両の E/E システムが提供する多くの機能は、これまでと同様に、パワートレイン、シャーシ、ボディー、マルチメディアといった従来のドメインに関連付けることができます (図 1)。ただし、車両内外での新技術の使用により、運転支援、自動運転、接続性 (Car2X/V2x 通信) などのテーマの重要性が格段に高まっています。

機能と ECU からなるこのような包括的なネットワークは、コンピューター支援の開発環境と一元的な E/E データベースを利用してこそ効率的に管理できます。この開発環境がなければ、内部的な調整と、整合性を失いがちな冗長なデータの管理に要する負担が増加します。

その結果、開発費が上昇して必要な作業時間が伸びるばかりでなく、市場投入後のエラー発覚という最悪の事態を招きかねません。

機能と ECU のネットワークの管理

E/E システムが提供する車両機能は、そのほとんどが開ループ／閉ループ制御、モニタリング、診断の各機能に関するものです。これらと車両の機械部品とのやり取りはセンサーとアクチュエーターを介して行われますが、それらは車内のさまざまな場所に設置されるため、E/E システムも当然、ジオメトリー上分散したシステムとなります。多くの機能は 1 つのドメイン内で動作するだけでなく、ドメインの境界を越えた、1 つの機能ネットワーク上で連動します。特に運転支援機能はパワートレイン、ステアリング、ブレーキシステムの機能と密接に連携しています。これは車両機能の実装に必要なセンサー、ECU 機能、アクチュエーターからなる制御チェーンを使用して表すことができます (図 2)。

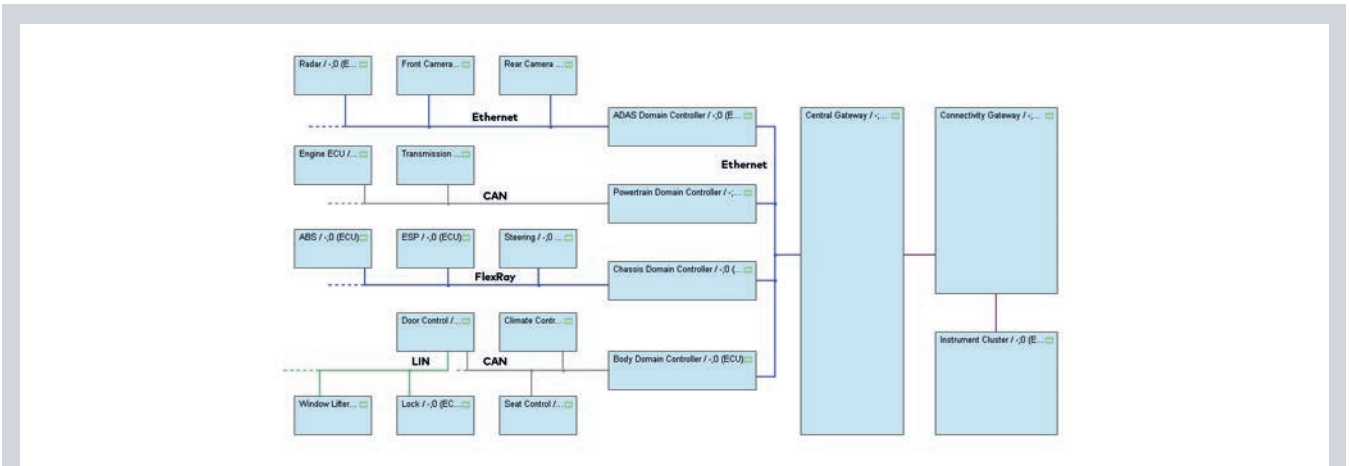


図 1: ECU のネットワーク

機能および ECU のネットワークの定義と ECU ネットワークへの機能の分散は設計の自由度が高いため、アーキテクチャー上の重要な決定事項となっています。総コスト、重量、設置スペース、バス負荷、安全性、セキュリティなど、アーキテクチャーのほぼすべての評価基準がこの影響を受けるといってもよいでしょう。このことは、量産開発のプロセスにおいて、これら 2 つのレイヤー、すなわち機能ネットワークと ECU ネットワークの両者を一貫して区別しなければならないことを意味します。さらに、これらのレイヤーはある特定の設計バリエーション専用には開発されるものではありません。

開発は常に、同じプラットフォームを基盤として設計される一連の車両を、より広くカバーできるように行われ、何人もの担当者が、(通常 20 以上の) 複数のバリエーションと一緒に開発するのが普通です。そのため、タスクの複雑さは相当なものになります。組

織がこれに対処するには、開発におけるすべての設計要素とそれらの相互関係を、設計要素の共通ストレージへの保存、可視化、コンピューター支援による評価と設計の自動化などを通じて俯瞰的に把握できるようにしておくしかありません。

E/E システムの分割と統合におけるプロセス制御

機能ネットワークと ECU ネットワークのこのような区別は、E/E システムの分割と統合、そして自動車メーカーとサプライヤーの間の協力を円滑に進めるための決め手ともいえる前提条件です。機能および ECU ネットワークを分割/統合するタスク、役割、責任は、これを土台としたうえで詳細に定義、計画、追跡できます。図 3 に、E/E 開発のプロセス全体の概要を示します [1]。

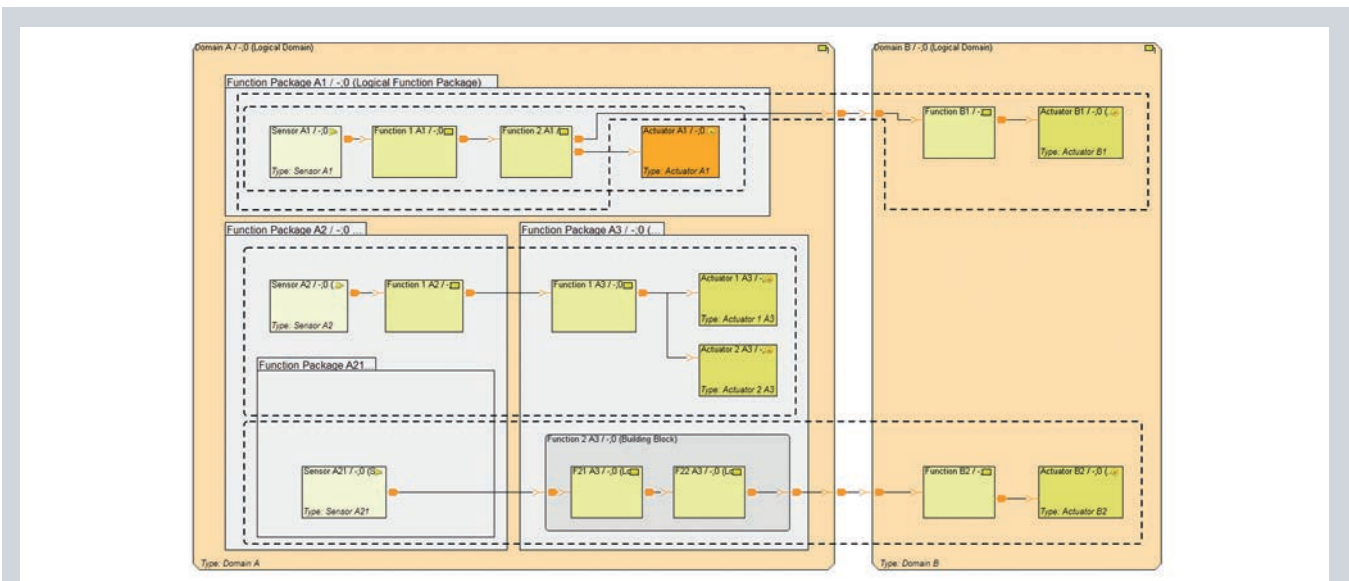


図 2: 機能のネットワーク

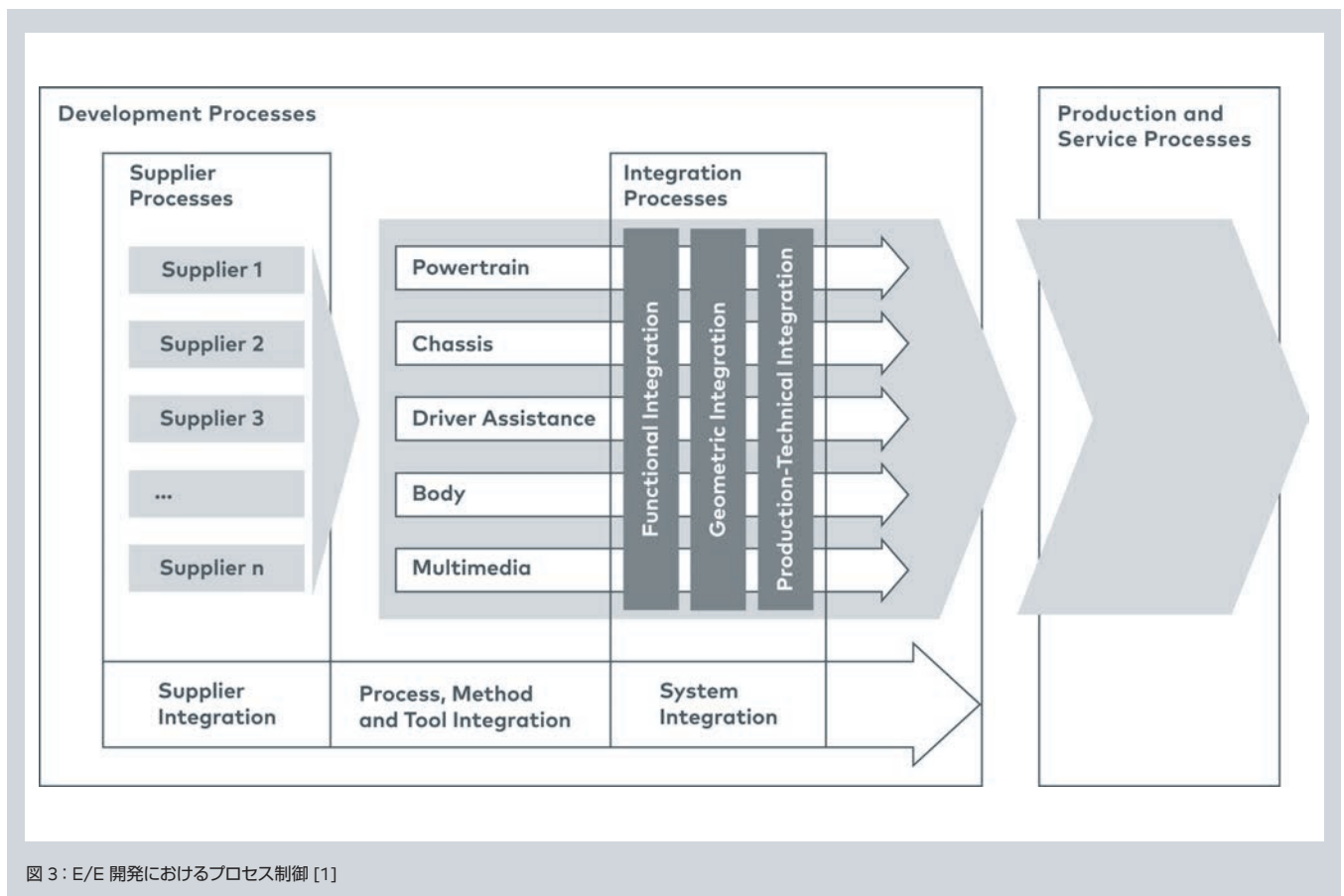


図 3：E/E 開発におけるプロセス制御 [1]

このような開発プロセスには、適切な IT システムによるサポートが必要です。それには、機械設計、E/E 開発、ソフトウェアおよびハードウェア開発など、個々の開発領域に応じた専用の開発

ツールが使用されます (図 4)。これらから得られた開発結果はすべて、以後の量産プロセスやサービスプロセスをサポートする製品ライフサイクル管理システム (PLM) に送られます。

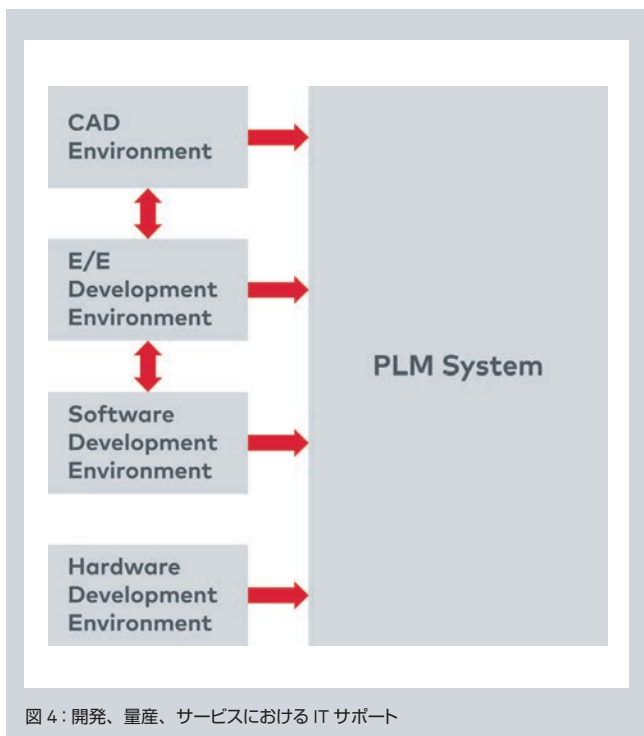


図 4：開発、量産、サービスにおける IT サポート

成功の鍵は交換形式の標準化

E/E 開発、特に自動車メーカーとサプライヤーが直接やり取りする場面では、E/E 設計要素の交換形式を標準化するメリットは以前から認識されていました。

現在では、それに必要な標準規格の策定、集約が行われ、幅広く採用されて、長年にわたって適用されています。

- > ReqIF/RIF は要求、機能記述、テストケースの交換に使用されます
- > AUTOSAR からは、包括的な記述フォーマットを使用した設定が可能な、ECU 用のベーシックソフトウェアのプラットフォームが提供されています。ここで最も重要なフォーマットは AUTOSAR System Description で、これはソフトウェアアーキテクチャー、ECU ネットワーク、そして CAN、LIN、FlexRay、Ethernet 通信といった領域の、自動車メーカーと ECU サプライヤーの間で交換されるデータを完全にカバーします

> KBL はワイヤーハーネスとジオメトリ情報の交換に広く使用されている標準規格ですが、今後数年間、同じ標準化団体が策定した標準規格 VEC により補われ、中期的にはこれに置き換えられる見込みです

このように、サプライヤーとメーカーの間の E/E データ交換のかなりの部分がかバーされています。このようなフォーマットの「デジタル」仕様書が文字ベースの仕様書の補足として付属したり、ときには完全に文字ベースの仕様書と置き換えられるケースが増えています。

E/E 開発のためのチーム用プラットフォーム

これらの E/E 仕様書の要素の作成と保守は常にチームや部門間で分担されます。そしてこれが企業の枠を越えて行われることも少なくありません。チームがベースで行われるこのような作業の実現には、開発ツールの使用が必要です。ここですぐ直面するのが、チームとデータベースのための、以下のような基本的な要求を満たした適切なプラットフォームはどれかという問題です。

- > マルチユーザー操作：競合の可能性を極力抑えながら、複数のユーザーが設計要素を同時に、並行して処理できる
- > テストケースまでの実装の全行程で要求を追跡できる（特に ISO 26262 で求められている安全関連機能に関して）
- > 作成される E/E 設計要素を細かい粒度でアーカイブ、バージョン管理、再利用するコンセプト
- > リリースおよびプロジェクトのプランニング機能に加え、透視的なプロジェクト追跡機能も統合された、統制の取れた変更およびリリース管理

- > データストレージに対するシングルソースの原則に基づき、E/E 設計要素の保守負担を制御可能
- > 各種の役割をサポートするための権限と成熟度レベルを細かい粒度で指定するコンセプト
- > 多くのユーザーと大量のデータを保有し、開発拠点多各地に分散している大規模な組織のサポート
- > 顧客とサプライヤーの統合
- > 正当な経費内の、バックアップのサポートも含めた管理と運用
- > E/E 開発環境の今後の開発に整合性があり、既存データの移行も可能であるため、将来も継続して使用できる
- > 既存の IT、プロセス、ツール環境に柔軟に統合できるオープン性

E/E 開発環境の実装には、以下に示すように、さまざまな原則に基づく多様な方法があります。

1. ドキュメントベースの作業方法：

さまざまなオーサリングツールを使用し、作成した設計要素ファイルは一元的なデータベースに保存（図 5a）

長く使われてきた実績のあるオーサリングツールとそのデータセットには、それを使用するメリットもありますが、反面、チームワークの粒度が荒く、そこで作成されたファイルについてしかトレサビリティを確保できないというデメリットもあるため、シングルソースの原則に従って、チーム全体が詳細な E/E 設計要素のレベルで密接に協力することは不可能です。個々のオーサリングツールの枠を越えての整合性のある開発は難しく、手動による作業が別途必要になります。

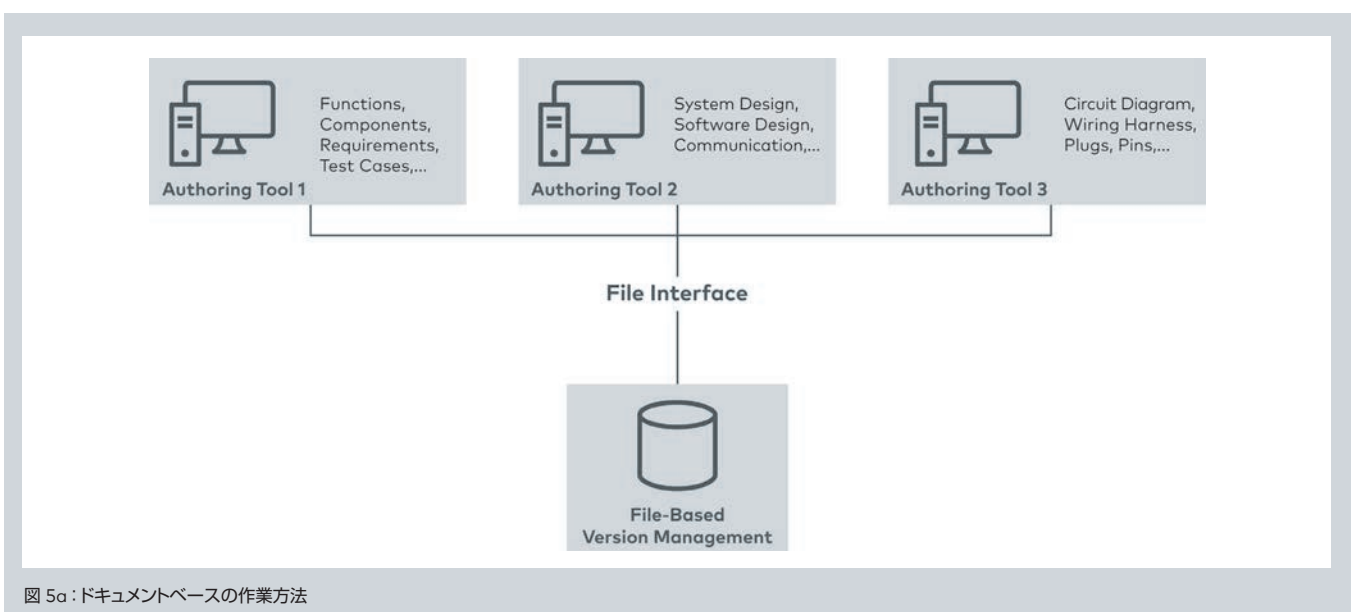
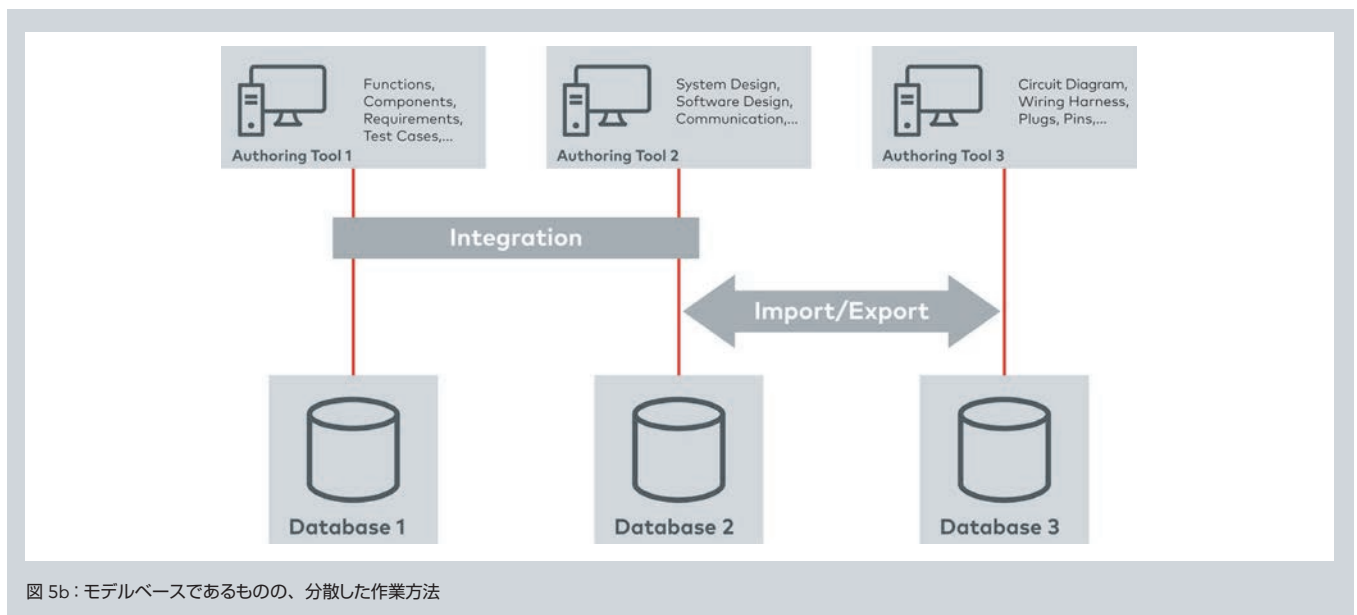


図 5a：ドキュメントベースの作業方法



一般的に使われているオーサリングツールの多くは、今日でもこの作業方法にしか対応していませんが、これは作業拠点が1つの場合にしか適さず、チームでの利用には不十分です。

運用面では、多種多様なオーサリングツールをロールアウトしたうえ、共通のデータベースを運用および保守しなければなりません。このようなファイルベースの作業方法は、隣接するソフトウェア開発の領域で使用され、ファイルベースのマージ、更新、バージョン管理、分岐などのプロセスによってサポートされます。

ただし、このような仕組みにも制限があります。たとえば、大きいファイルに含まれているソフトウェア機能と要求とを簡単にリンクすることはできません。E/E 開発では、設計要素の粒度は要求、機能、シグナル、ECU、バス、ピン、プラグのように非常に細かくなっています。

これらの設計要素間の関係は非常に重要ですが、これらの設計単位が大きいファイル内にいわば「紛れ込んで」いる場合、その追跡は困難です。そのため E/E 開発には、さまざまな E/E 設計要素とその関係を細かい粒度で管理できる、モデルベースのアプローチが必要です。

2. モデルベースであるものの、分散した作業方法:

さまざまなオーサリングツールを使用し、作成した設計要素ファイルは一元化されていないデータベースに保存(図 5b)

多くのオーサリングツールは上のようなニーズを認識し、その対策として独自のチームおよびデータベースプラットフォームを同梱していますが、必要なユースケースがすべてサポートされるわけではありません。そのため、一目見ただけでも、各種のユースケースに応じて個別のソリューションをオーサリングツールに統合する必要があるのは明らかです。実績のあるオーサリングツ

ールは、チームが手軽に使用できる点で有利ですが、統合の制限というマイナス点は残ります。

運用面では、複数のデータベースとオーサリングツールを保守しなければならず、ときにはその統合も必要になります。この場合の接続はポイントツーポイントで行われます。統合は OSLC などの標準規格を使用すれば可能で、統合された設計要素は、シングルソースの原則に基づいて 1 つの場所に保存されますが、それらは表示はもちろん、多様なオーサリングツールを使用して編集することもできます。

ただし、このような形で生じるデータベースの境界を越えた関係は堅牢なものではなく、確実なバージョン管理やアーカイブはできません。さらに、ツールのバージョン更新の際にデータの移行が可能であったとしても、そのような関係がデータ移行中に破壊されたり、あるいは新しいバージョンのツールで新しい関係を導入できたとしても、元の関係を復元できないことがあります。

そのため、プロキシオブジェクトを用いた重複および余剰データの管理も多く許容されていますが、それらの関係は周期的なエクスポート/インポートと更新プロセスを使用して確認しなければなりません。その他にも、前と同様に、統合されたデータモデルがない、ツールの境界が常に意識される、アーキテクチャーが多様である(リッチクライアント/ Web クライアントなど)、リリース、バージョン管理、バックアップ、移行のコンセプトが異なるといったデメリットがあります。この結果、統合や運用の負担が増加します。

3. モデルベースかつ統合された作業方法：

整合性のあるオーサリングツールを使用し、作成した設計要素ファイルは一元的なデータベースに保存 (図 5c)

モデルベースであるものの分散した作業方法の弱点は、整合性のあるオーサリングツールを使用し、一元的なデータベースにデータを保存することで解決できます。以前は、整合性のある E/E エンジニアリングのデータモデルと一元的なデータストレージが有効だと判断した現場では、独自技術に基づく、プロプライエタリでの開発 (多くはデータベース) がソリューションとして投入されていました。これは、そのようなソリューションが市販されていなかったことによります。

プロプライエタリのソリューション - 多大な手間とコスト

これにははっきりとしたメリットとデメリットがあります。プロプライエタリのソリューションでは、その組織でのデータモデルとプロセスを、極めて的を絞った形でサポートできます。しかし、プロプライエタリのソリューションは開発、保守、テストの負担の増大を招きます。また、多くの場合、ユーザーインターフェイスも最先端とはいえません。

幅広い設定オプションを持つ製品ソリューション - 実際的で経済的

幅広い設定オプションを持つ製品ソリューションは、効率向上によって経費を節減でき、統合の手間も大幅に軽減できる点で、

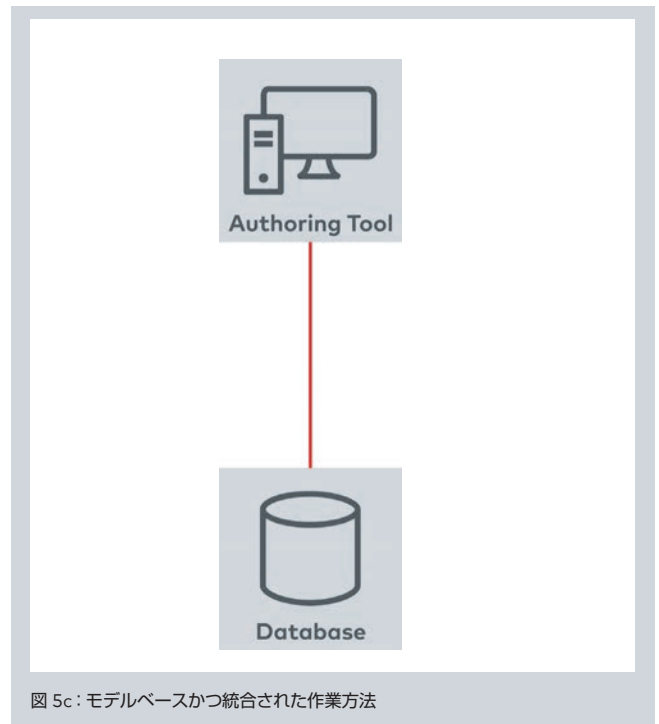


図 5c: モデルベースかつ統合された作業方法

購入コストを簡単に正当化できます。このアプローチに忠実に従うのが PREEvision です。先に述べた標準規格が自動車業界に採用され、整合性のある、完全な形の自動車用 E/E データモデルがはじめて設計できるようになりました。

PREEvision の E/E データモデルは、要求からテストに至るすべての実装ステップで、整合性のあるモデルベースの作業方法を支える基盤となります。

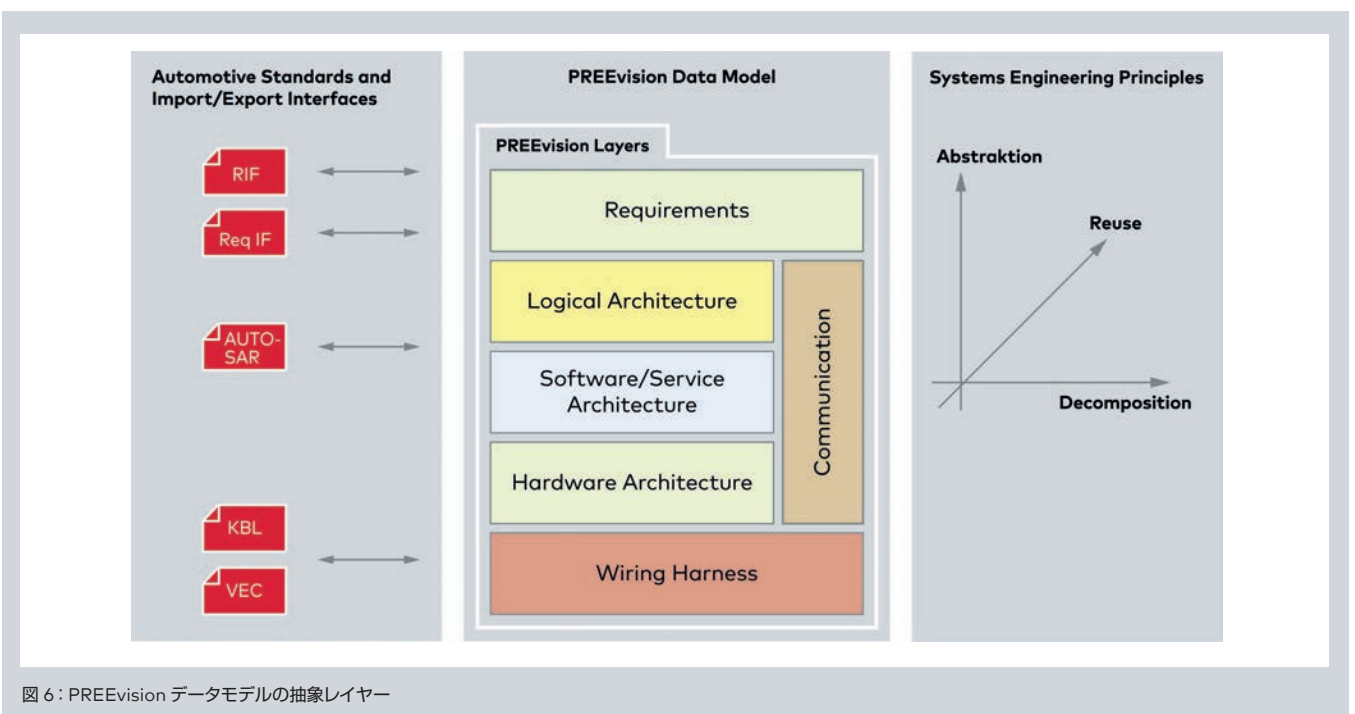


図 6: PREEvision データモデルの抽象レイヤー

E/E アーキテクチャーに求められるあらゆる面が、1つの統合されたアプローチでモデル化されています。

- > 機能、カスタマーフィーチャー、要求
- > ネットワークの特性と機能の分散
- > ソフトウェアおよび通信の特性
- > ハードウェア、ワイヤーハーネス、ジオメトリーの特性

PREEvisionはこのアプローチを採用するために、システムズエンジニアリングで実績のある「抽象化」、「デコンポジション」、「再利用」の3つの原則に従い、プロダクトラインとプロダクトバリエーションのモデリングを多様なモデルレイヤーを用いてサポートします(図6)。

データモデルのレイヤーを見てみると(図6)、これは設置スペースや配線経路を表すジオメトリーレイヤーから、ワイヤーハーネスレイヤー、電気配線レイヤー、そしてECUネットワークレイヤーに至る抽象レイヤーを垂直統合して表現されています。ソフトウェアや通信と詳細はそれに並行してモデル化されます。

論理機能アーキテクチャーのレイヤーでは、ハードウェアとソフトウェアとを抽象化して記述します。要求やカスタマーフィーチャーのレイヤーではさらに高度な抽象化が行われます。PREEvisionデータモデル図の水平方向ではデコンポジションがサポートされます。各レイヤーには、ボトムアップまたはトップダウンで使用できる階層のコンセプトが用意されています。これらの2つに直行する第3の方向では、再利用とバリエーションのコンセプトがそれぞれのモデルレイヤーと階層レベルで使用可能となっています。

さらなるメリットを与えてくれるのが、ハードウェアおよびソフトウェアの実装から抽象化した論理機能アーキテクチャーです。車両機能の実装に用いられるハードウェア、ソフトウェア、通信のテクノロジーは自動車の世代ごとに変わりますが、車両機能の制御チェーンは多くの場合、長年にわたって変化しません。ここにはその制御チェーンを記述できます。これはパワートレイン、シャーシ、運転支援およびボディーのドメインに属する、制御とモニタリングの機能のほか、マルチメディアシステム用の機能にも適用

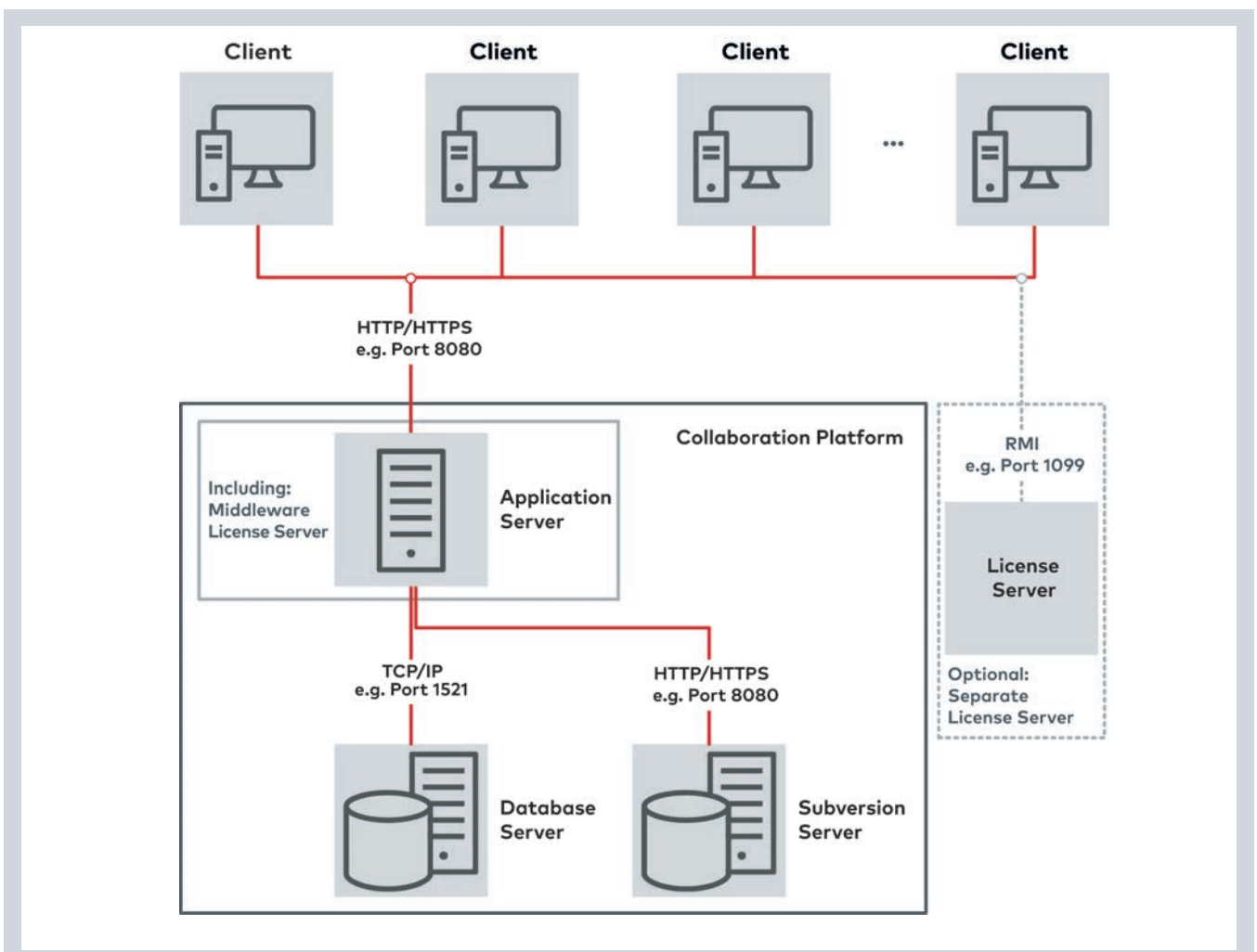


図7: PREEvision Collaboration Platform のアーキテクチャー

されます。これらレイヤーはすべて、いわゆる「マッピング」によって相互に結ばれています。

機能のビヘイビアモデリング用のツールを使用して、シミュレーション、ラビッドプロトタイピング、コード生成用に作成されたファイルは、このモデルのコンテキストのさまざまな場所に保存できます。

仕様フォーマットである RIF、ReqIF、AUTOSAR、KBL の作成と処理のため、インポート／エクスポートインターフェイスも用意されています。PREEvision データモデルは、ここで交換されるデータを、理解が容易で編集しやすい、再利用可能なエンジニアリング用の設計要素、たとえば ECU、プラグ、ピン、ソフトウェアコンポーネント、シグナル、メッセージなどに集約し、それらの間の必要な関係を提供します。編集には高機能なエンジニアリング機能、グラフィカルなダイアグラム、テーブルベースのエディターなどを利用できます。

統合型のモデルベースのアプローチによって、解析対象となるアイテムがドメイン全体で可視化され、シングルソースの原則がサポートされます。モデルのサブセクションやモデル全体の整合性と完全性は、設定可能な検証機能を通じてチェックできます。安全面の検討時には特にそのメリットが発揮され、機能安全のためのプロセスを、ハザード分析およびリスクアセスメント、故障モード影響解析 (FMEA)、フォールトツリー解析などの主要な手法のツールで、既存のプロダクトラインのデータを使用してサポートできます。

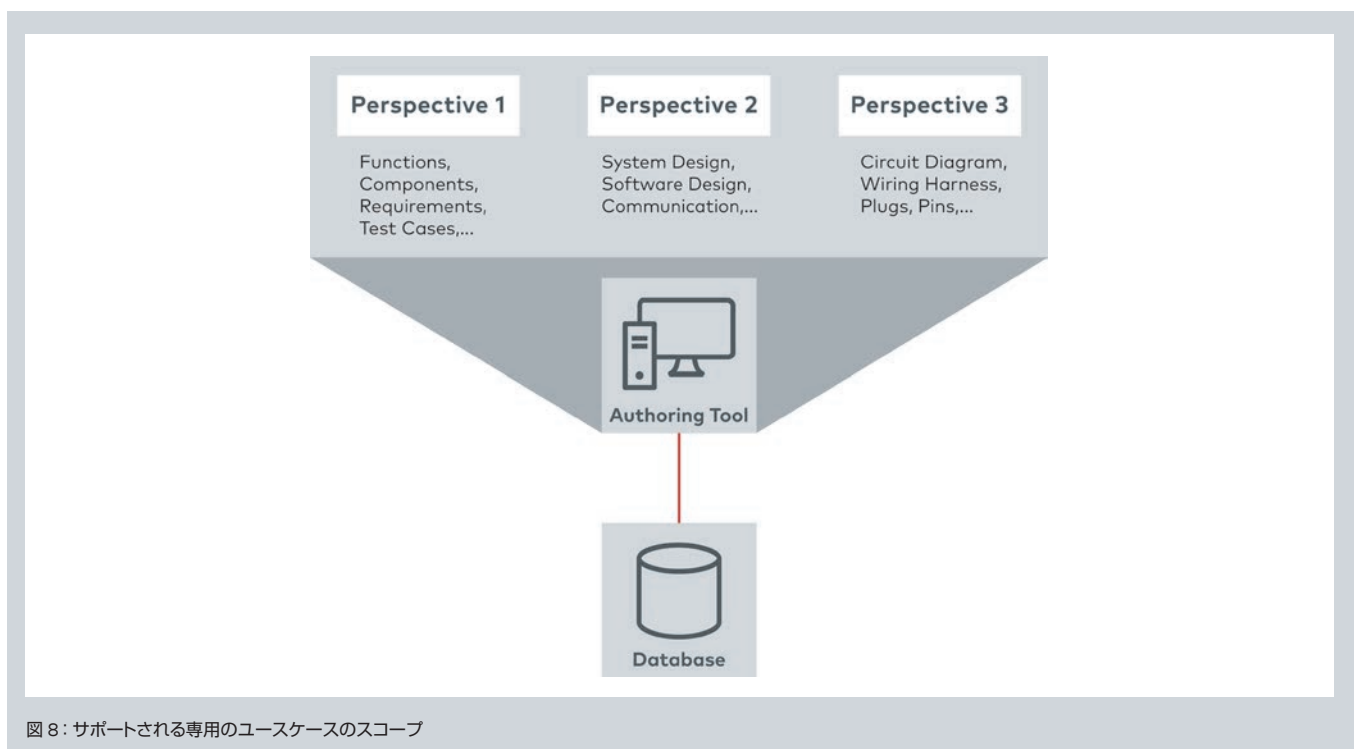
フィーチャー指向ドメイン解析 (FODA) の原則に準拠したフィーチャーモデルをカスタマーフィーチャーレイヤーで使用する

ことで、ある車両バリエーションに有効な一連のフィーチャーを定義できます [2]。これらのフィーチャーは、いずれも他のレイヤー上にある、対応する設計要素とリンクできるため、選択したバリエーション用の E/E アーキテクチャーを導出、編集、チェック、エクスポートすることが可能です。

PREEvision は 3 階層のアーキテクチャーを持つコラボレーションプラットフォームオプションを通じてチームワークをサポートします (図 7)。これにより、複数のユーザーが 1 つまたは複数のプロダクトラインに並行して手を加えることができます。E/E の設計要素はいずれもきめ細かいバージョン管理の対象となっており、チケットへの紐付けによって連携の取れた変更およびリリースプロセスが実現されます。

役割と権限、アプリケーション固有の追加属性、成熟度レベル、レポート、テーブルエディターおよびビュー、スクリプト、検証機能のほか、専用の、自動化も可能なインポート／エクスポート用インターフェイスも含む包括的な設定オプションが、成熟した、実績のあるオープンなデータモデルと、数多くの開発支援機能と共に標準で付属しています。

これによって、各開発組織の固有のニーズを柔軟にマッピングし、既存のプロセスおよびツール環境に PREEvision を統合することが可能になります。さらに、フィルターとエディターの事前設定を用いたいわゆる「パースペクティブ」を使用して、専用のユースケースに合わせた、ユーザーフレンドリーな作業用インターフェイスを設定することもできます (図 8)。



まとめと展望

最後になりましたが、PREEvision に対するベクターの徹底した保守と開発により、将来のプロセスが自動車業界で進む急速なイノベーションに後れを取ることなく、確実にサポートされることを申し添えます。新しいバージョンには、データベースのインベントリーデータを移行するメカニズムが付属します。これによってデータモデルの後続開発と新しい PREEvision への切替えを、過剰な負担なく行えるようになります。これは重要なタスクですが、プロプライエタリ開発では、差し迫った事態が生じない限り、その意義はなかなか実感されません。

PREEvision のアプローチは長年にわたって量産車での使用に応用されており、要求管理からアーキテクチャー設計、機能指向システムおよびコンポーネント設計、ソフトウェアおよび通信設計、ワイヤーハーネス開発、そして量産のためのテストとリリースに至る E/E 開発プロセス全体を、整合性の取れた、モデルベースの、そして将来性を考慮した形でサポートしています。

参考文献：

[1] Hans-Georg Frischkorn, Herbert Negele, Johannes Meisenzahl, BMW Group, München: The Need for Systems Engineering. An Automotive Project Perspective. Key Note at the 2nd European Systems Engineering Conference (EuSEC 2000), München, 13. September 2000.

[2] Kang, Kyo c. et al: Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical Report CMU/SEI-90-TR-21 ESD-90-TR-222, Software Engineering Institute, Carnegie Mellon University, Pittsburg, Pennsylvania, USA, November 1990. www.sei.cmu.edu/reports/90tr021.pdf

執筆者：



Dr. Thomas Beck

ベクター・グループ CEO。ベクター本社 (Vector Informatik GmbH; ドイツ・シュツットガルト) にて勤務。



Clemens Reichmann (Dr.-Ing.)

「PREEvision」のテクニカル・ディレクター。ベクター本社 (Vector Informatik GmbH; ドイツ・シュツットガルト) にて勤務。



Jörg Schäuuffele (Dipl.-Ing.)

「PREEvision」のプロダクト・マネージャー。ベクター本社 (Vector Informatik GmbH; ドイツ・シュツットガルト) にて勤務。

本稿は、ドイツで発行された『“ATZ elektronik” magazine Issue 06 – December 2016』に掲載された記事内容を和訳したものです。

画像提供元：

見出し画像および図 1～8：Vector Informatik GmbH

■ 本件に関するお問い合わせ先

ベクター・ジャパン株式会社 営業部

(東京) TEL: 03-5769-6980 FAX: 03-5769-6975

(名古屋) TEL: 052-238-5020 FAX: 052-238-5077

E-Mail: sales@jp.vector.com