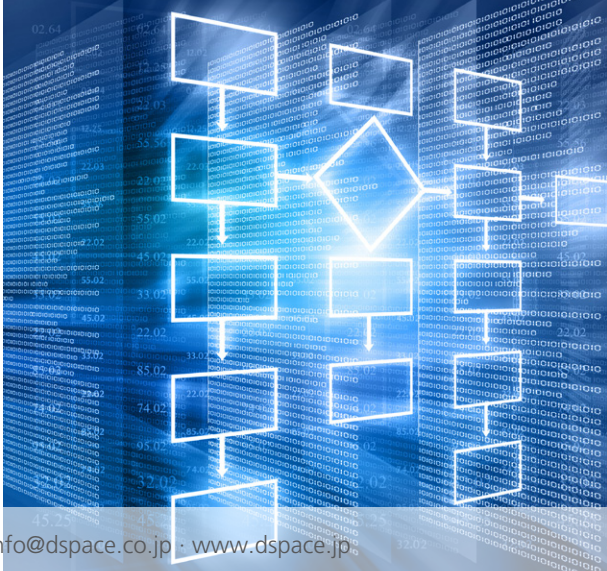


# Solving the Complexity Puzzle

機能開発の多くは、さまざまな開発パートナーの多数のソフトウェア開発者で構成された作業グループ間で行われています。このことが開発プロセスにとって大きな課題となってきました。そのため、効率的なモデルベースのソフトウェア開発を行うための信頼性の高いツールチェーンが非常に重要になります。Simulink/TargetLink と Model Engineering Solutions 社製のツールが、最適なソリューションを提供します。



大規模分散開発における複雑性の  
克服および整合性の構築



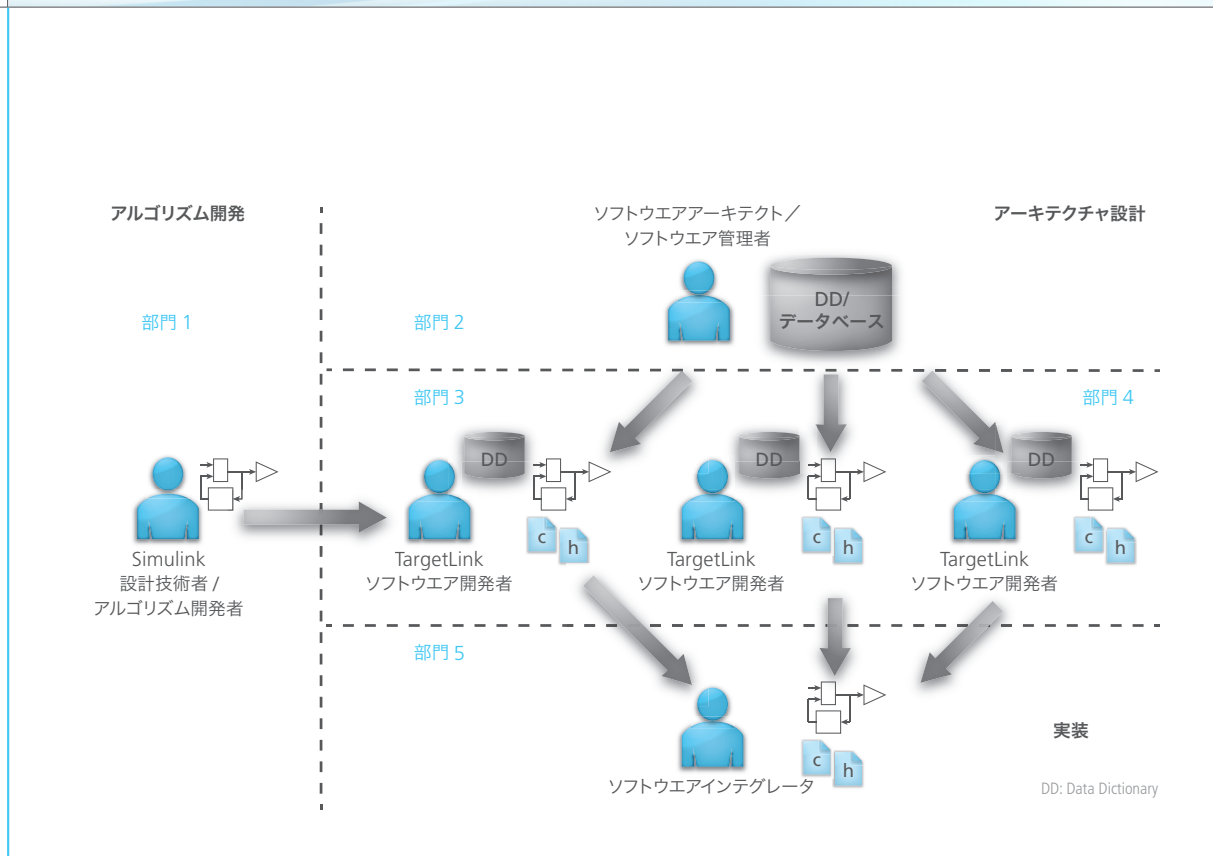


図 1：大規模なチームによる分散開発。モデルは「アーキテクチャ設計」と「アルゴリズム開発」のフェーズで定義され、サブファンクションの開発を通して具体的な機能が追加されます。その後すべてのモデルを収集し、妥当性を確認して ECU に実装します。

### 課題：分散開発

ソフトウェア機能の実行可能な仕様である Simulink®/TargetLink® モデルは、コードだけでなく、A2L ファイル、AUTOSAR XML ファイル、およびソフトウェア文書などの成果物も生成します。コードの設計および量産コードの自動生成が個々のソフトウェアコンポーネントと機能のみに適用さ

れるという開発ステップの場合、それらを統合するまで開発者は不整合の有無を確認することができません。また多くの場合、それ以前の開発ステップをテストするメカニズムが存在しません。車両機能の複雑化が進み、開発環境が多くの作業グループに分散している現在では、この問題への対処はこれまで以上に重要になって

います。幅広い機能の開発をモジュール型かつ分散型の環境で効率的に行うためには、開発者が開発メカニズムを調整し、Simulink/TargetLink に合わせてツールチェーンを変更する必要があります。

### モデリングガイドラインによる整合性の向上およびエラー頻度の軽減

Simulink/Stateflow® は多くのモデリング機能を提供しますが、そのすべてが効率的な量産コード生成に使用できるわけではありません。そのため、多くの開発者が同じソフトウェアの開発に関わる場合、

方法	利点
モデリングガイドライン	<ul style="list-style-type: none"> <li>■ 整合性</li> <li>■ エラー頻度の低減</li> <li>■ 修正作業の削減</li> </ul>
再利用(ライブラリ、参照先モデル)	<ul style="list-style-type: none"> <li>■ モジュラー方式の開発</li> <li>■ モデルの整理および階層により判別が容易</li> <li>■ 同じモデルの再利用により開発工数を削減</li> </ul>
シングルソース仕様	<ul style="list-style-type: none"> <li>■ Data Dictionary のソフトウェアおよびインターフェース仕様により、開発チームメンバー間でのやり取りが容易</li> </ul>
インクリメンタルコード生成	<ul style="list-style-type: none"> <li>■ レビュー期間の短縮</li> <li>■ コード生成の迅速化</li> <li>■ ソフトウェアの統合とテストが容易</li> </ul>
Data Dictionary からコードを生成	<ul style="list-style-type: none"> <li>■ 共有の変数を 1 つのファイルに生成</li> </ul>
TargetLink Data Dictionary と Model Compare による差分&マージメカニズム	<ul style="list-style-type: none"> <li>■ インターフェース定義およびモデルへの変更を追跡可能</li> </ul>
M-XRAY による複雑性の分析	<ul style="list-style-type: none"> <li>■ 適切なモデルのパーティション化の表示</li> </ul>

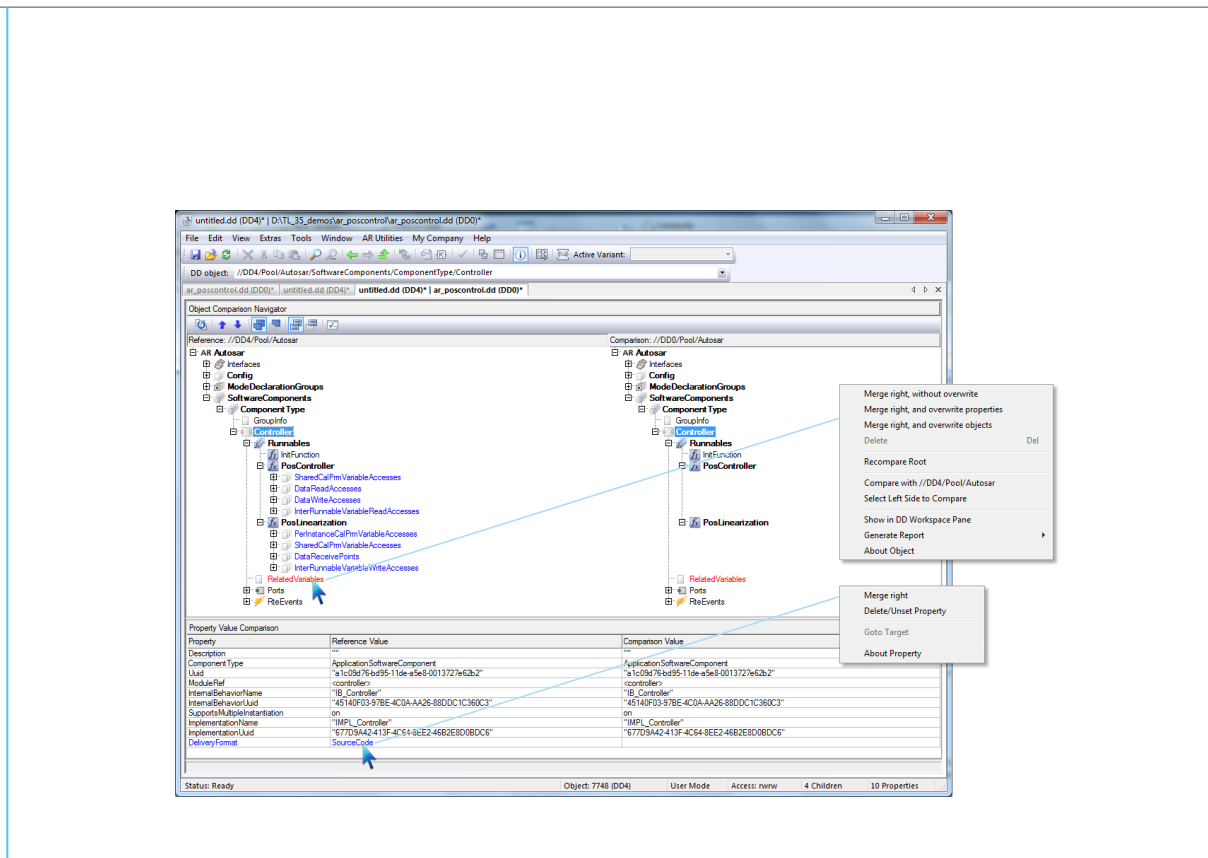


図 2: 異なるソフトウェア仕様の比較

モデルの欠陥リスクを低減するモデリングガイドラインの存在が特に重要になります。こうしたガイドラインを遵守することで、作業のやり直しが最小限で済み、モデリングスタイルの統一やテストの単純化が可能になります。また、このガイドラインはレビュー時のリファレンスとしても役立ちます。開発チーム間でのモデルや機能のやり取りも容易になります。MES Model Examiner® などのガイドライン自動チェック用ツールを使用すると、ガイドラインの違反をチェックして修正することができます。

### モデルの分割と再利用

分散開発プロセスにとって、シングルソースの原則は不可欠な要素です。「シングルソース」とは、設計から閉ループ制御や統合までのさまざまな開発フェーズを通して同じモデルが使用されることを意味します。Simulink/TargetLink は、こうしたメカニズムを次の方法で実現します。

- Simulink のライブラリ機能を使用した、マルチインスタンス化が可能なモデルパーツの再利用
- モデルリファレンス機能を使用した複数のモデルの統合

### シンプルなやり取りと管理

大規模な開発チームでは、機能開発、ソフトウェアアーキテクチャの管理、およびソフトウェアの開発と統合といったタスクを一人の担当者のみで行うことはほとんどありません。実際には、多くのチームメンバーが同じ情報にアクセスします (図 1)。設計技術者は主に仕様をやり取りし、それを編集して保存することを繰り返すため、その仕様の整合性維持が必要となります。TargetLink では、専用ツールとして TargetLink Data Dictionary (TL-DD) を提供しており、XML や AUTOSAR

XML などのさまざまな交換フォーマットがデフォルトでサポートされています。モデル内のデータオブジェクトは Data Dictionary 内のデータオブジェクトとリンクされているため、データからアルゴリズムを切り離すことができます。データは、モデルおよび Data Dictionary 間で同期します。

#### 強力なインクリメンタルコード生成

インクリメンタルコード生成は、分散型のモデルベース開発におけるもう 1 つの重要な手法です。コードは個々のソフトウェアユニットごとにインクリメンタル生成されます。個々のソフトウェアユニットがそれぞれ独立しているため、小さな機能に対して行った変更によるソフトウェア機能全体への影響は最小限に抑えられます。コードの生成は変更されたユニットに対してのみ行われ、その他のコード部分は変更されません。そのため、手作業でのレビューの時間を短縮でき、コード生成にかかる時間も最小限になります。これにより、大規

模な機能開発をより効率的かつ迅速に行えるようになります。

#### Data Dictionary からのコード生成

コードは、モデルとは別に Data Dictionary から直接生成されます。グローバルまたは共有の変数は Data Dictionary から 1 つのファイルに生成されます。この方法は次のような場合に使用できます。

- インターフェース変数などのすべてのグローバル変数を含み、それらの変数へのアクセス権を持つファイルが作成される場合。
- さまざまな機能のすべての適合パラメータが、単一の適合パラメータファイルから生成される場合。
- 自動または手動で生成したレガシーコードで使用する変数が 1 つのファイルとして生成される場合。

#### 効率的な差分&マージメカニズム

新しいソフトウェアを開発する場合、特に、

異なる部署やサプライヤがプロジェクトに関与する場合には、設計技術者は変更を識別できる必要があります。設計技術者とインテグレータは、変更したソフトウェアの成果物を相互にやり取りします。そのため、変更を識別できる信頼性の高いツールチェーンが不可欠になります。TargetLink Data Dictionary は、異なるバージョンを比較して変更部分を表示するために必要な機能を備えています (図 2)。また、これらの変更を追跡することで、モデルへの影響を確認できます。dSPACE の Model Compare を使用すると、容易かつ総合的にモデルを比較することができます。DD メカニズムはインターフェースの定義を自動的に更新し、変更が行われた場合でも整合性を維持します (図 2)。

#### モデルアーキテクチャの妥当性確認

複数の TargetLink モデルを分散開発に使用する場合、それらをサブファンクションやサブシステムに分割する必要があります。また、個々のサブシステムの複雑

図 3 : M-XRAY の分析レポート

構造の概要：主要部分

名前	複雑度	*	名前	複雑度	*	名前	複雑度	*	名前	複雑度	
燃料システム	96	├──	EGO センサ	33							
			MAP センサ	33							
			エンジン回転数	33							
			エンジンガスダイナミクス	33		├──	混合&燃焼	59	──	システム遅延	9
						├──	スロットル&マニホールド	42	├──	吸気マニホールド	67
									└──	スロットル	350
			空燃比コントローラ	58		├──	空気流量カリキュレータ	780			
						├──	燃料計算	75	──	切り替え補正	200

度を最小限に抑えることで、発生し得るエラーを減らし、サブシステムの可読性や保守性を維持する必要もあります。さらには、複雑度の低減を規定するISO 26262といった安全規格の要件も満たす必要があります。MES Model Examiner AddOn M-XRAYを使用すると、モデルの複雑度を自動的にチェックすることができます。M-XRAYは全体モデルおよび個々のサブシステムの複雑度を計算して評価します。さらに、TargetLinkモデルの定量的評価に関するすべてのモデル評価指標をリストアップします（「補足情報」を参照）。M-XRAYは、モデルの階層構造や各サブシステムの複雑度の概要を示す分析レポート（図3）を生成します。これにより、モデルの複雑度の評価が容易になり、極めて複雑なサブシステムの識別もできるようになります。■

局所的複雑度に関する基準値

基準値	評価
MV ≥ 750	高
MV < 750	中
MV < 300	低

## まとめ

大規模な開発チームでも、複雑かつ分散型の開発環境において整合性を効率的に管理できる方法は数多く存在します。モデルのパーティショニングやインクリメンタルコード生成といった手法、および変更を追跡できるツールの活用は特に有効です。設計技術者は、パーティション化を評価する場合、計測結果や評価指標を使用できます。このアプローチにより、幅広い機能の開発が容易になり、開発したサブファンクションのやり取りも効率化し、エラーも減少します。

## 補足情報

### モデルの評価指標

開発者は、TargetLinkモデルの比較に評価指標を使用することで、モデルの複雑度と品質を評価できます。ISO 26262などの安全規格では、セーフティクリティカルなモデルの複雑度の評価について規定しています（ISO 26262-6、5.4.7、表1を参照）。モデル評価指標は、テストおよびレビューに必要な作業工数の見積もりにも使用できます。開発者は、さまざまな開発段階で評価指標値を取得することにより、モデルの開発状況を監視し、特に複雑でエラーの発生しやすいモデル部分を非常に早期の段階で識別することができます。

モデルの複雑度の測定には、ブロック数、モデリング深度、インターフェース幅、または循環的複雑度などの評価指標も使用できます。ただし、こうした評価指標はプログラミング概念に基づいているため、評価方法としては適切でない場合もありま

す。Simulinkはデータフロー指向であるため、たとえば、モデルの循環的複雑度の評価はそれほど参考になりません。

ハルステッド複雑度測定から派生したモデルボリューム（MV）の測定は、モデルの複雑度を評価する重要な方法として業界で認知されつつあります。これにはモデルブロックだけでなく、ブロック間のリンクや、重み、複雑度、およびブロックのリンクに使用される信号なども含まれるため、開発者はMVの測定によりモデルの複雑度を評価できます。

MES Model Examiner® と M-XRAY AddOnは、TargetLinkモデルをモデル評価指標を使用して分析し評価する場合に使用します。M-XRAYはモデルを分析し、モデルのボリュームと該当するすべての評価指標を計算します。次に、その結果を簡潔に分かりやすく整理されたテーブル形式で表示します。このツールによ

り、モデルの複雑度の分布を効率的に計算し、複雑度を最小限に抑えることができます。■

### 参考文献：

Stürmer, I., Pohlheim, H., Rogier, T.: "Calculation and Visualization of Model Complexity in Model-based Design of Safety-related Software", (in German) in Keller, B. et. al., Automotive - Safety & Security, Shaker, pp. 69-82, 2010.