

Simulink プロジェクト ワークショップ°

MathWorks
Version1.0

目的

- 本資料は、サンプルシナリオに沿って Simulink プロジェクトの活用をご提案します。
 - 紹介する Simulink プロジェクトは R2015a を利用しております。
 - Simulink プロジェクト以外の機能も含んでおります。
 - 全ての機能の網羅的な説明ではなく、シーンごとでの利用方法になります。

Demo Simulink プロジェクトの機能を利用した操作を示します

Demo Simulink プロジェクト + 検証関係のツールボックスの機能を利用した操作を示します

目次

- Simulink Projects とは？
- モデルベース開発形態とプロジェクト管理要求

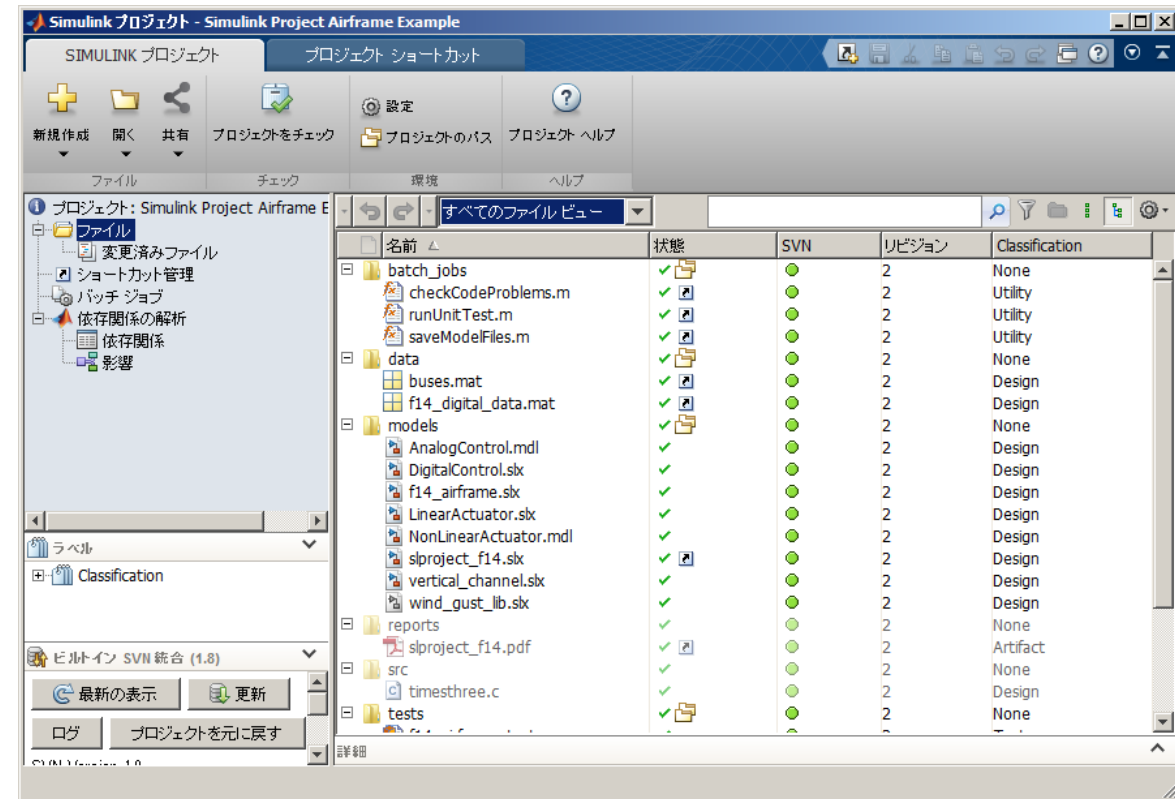
- サンプルシナリオ
 1. プロジェクトの開始
 2. ファイルの作成・編集
 3. モデルの分割と担当の割り振り
 4. プロジェクトの共有
 5. 並行開発

- まとめ

Simulink プロジェクトとは？

いろいろな作業を整理・管理したり、関係するファイル、設定、定義したタスクを管理して共有したり、ソース管理ツールと連携することができます

- ① まとめて表示
- ② 作業環境の設定
- ③ ショートカットの作成
- ④ ラベルの付加
- ⑤ 依存関係、差分の確認
- ⑥ ソース管理



- セキュリティ (ユーザー管理、構成要素へのアクセス権設定) は外部ツールの設定で行います。
- Simulink プロジェクトは R2011b で Simulink に追加された新機能です。

モデルベース開発形態とプロジェクト管理要求

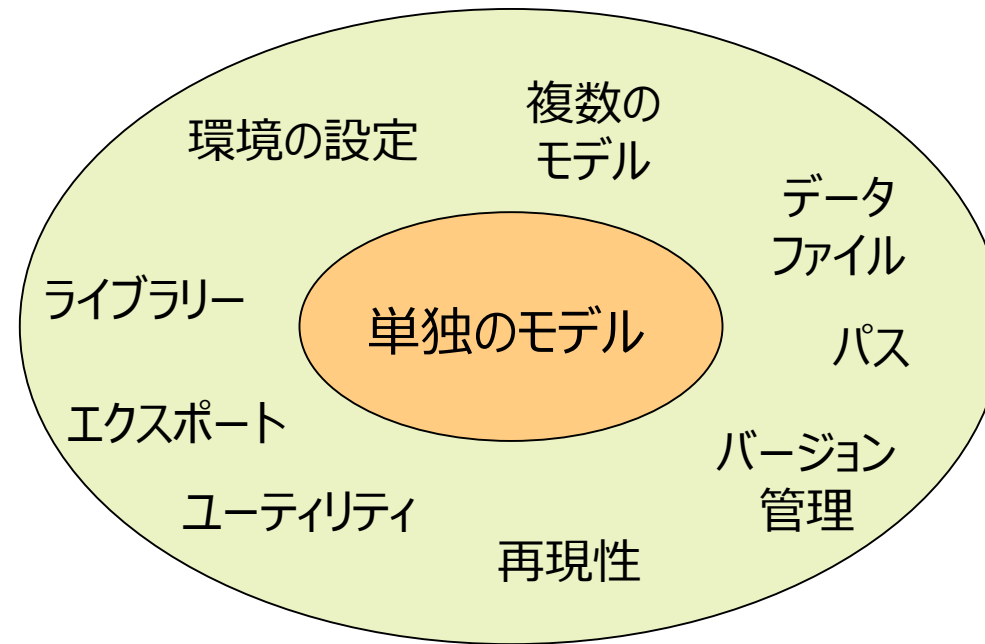
単独のユーザー

単独のモデル

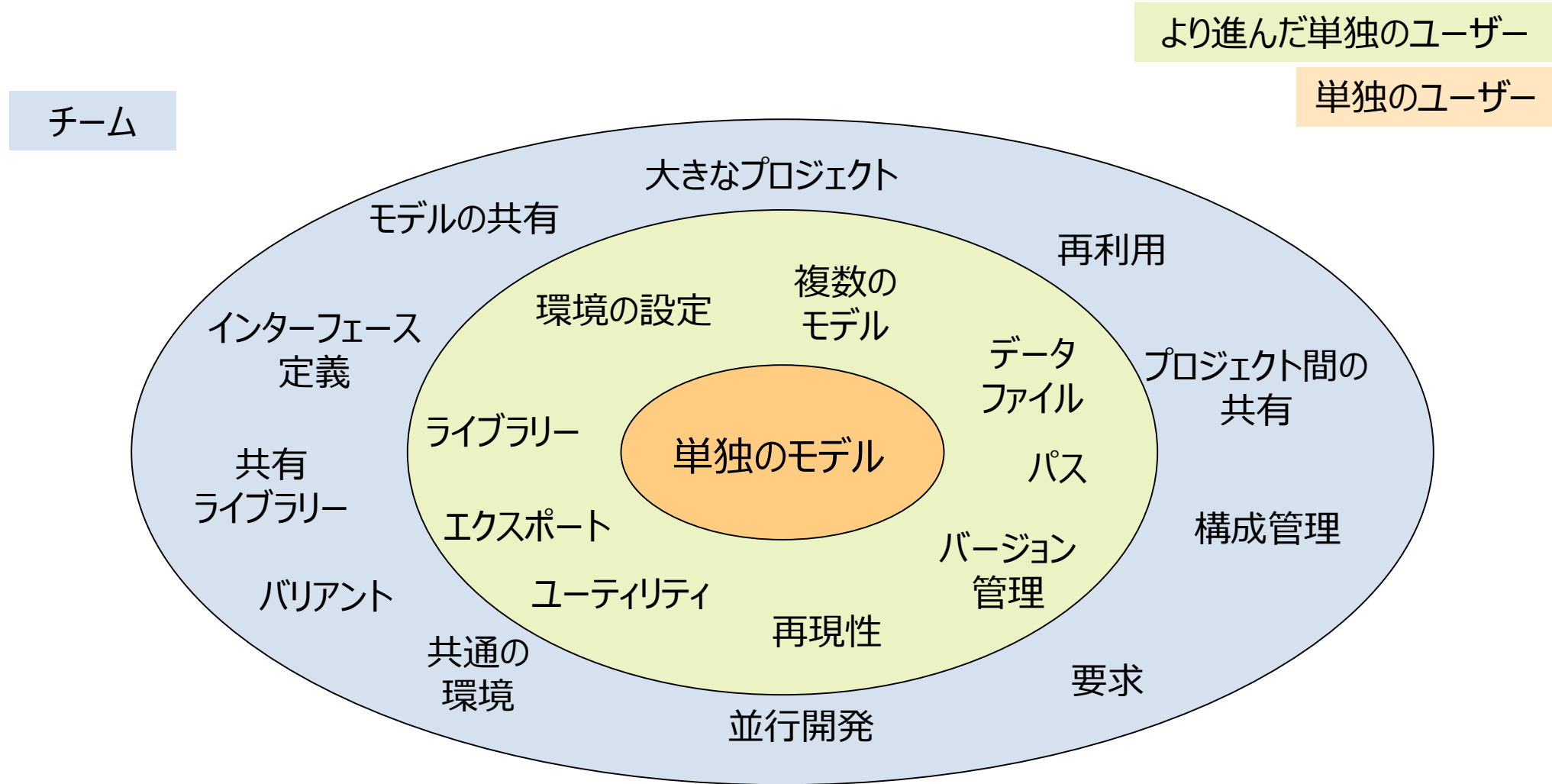
モデルベース開発形態とプロジェクト管理要求

より進んだ単独のユーザー

単独のユーザー

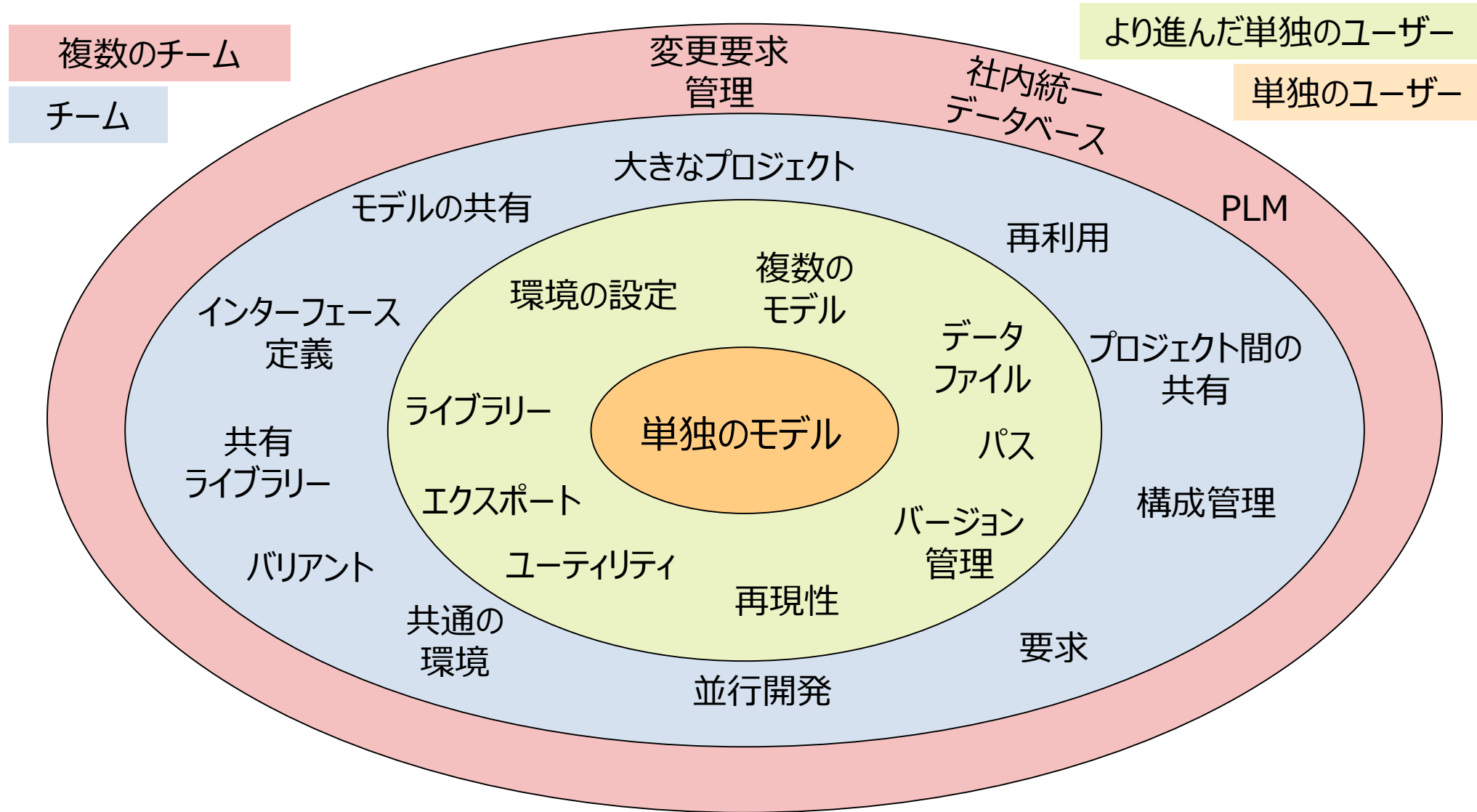


モデルベース開発形態とプロジェクト管理要求



➡ Simulink プロジェクトと構成管理ツールの活用

モデルベース開発形態とプロジェクト管理要求



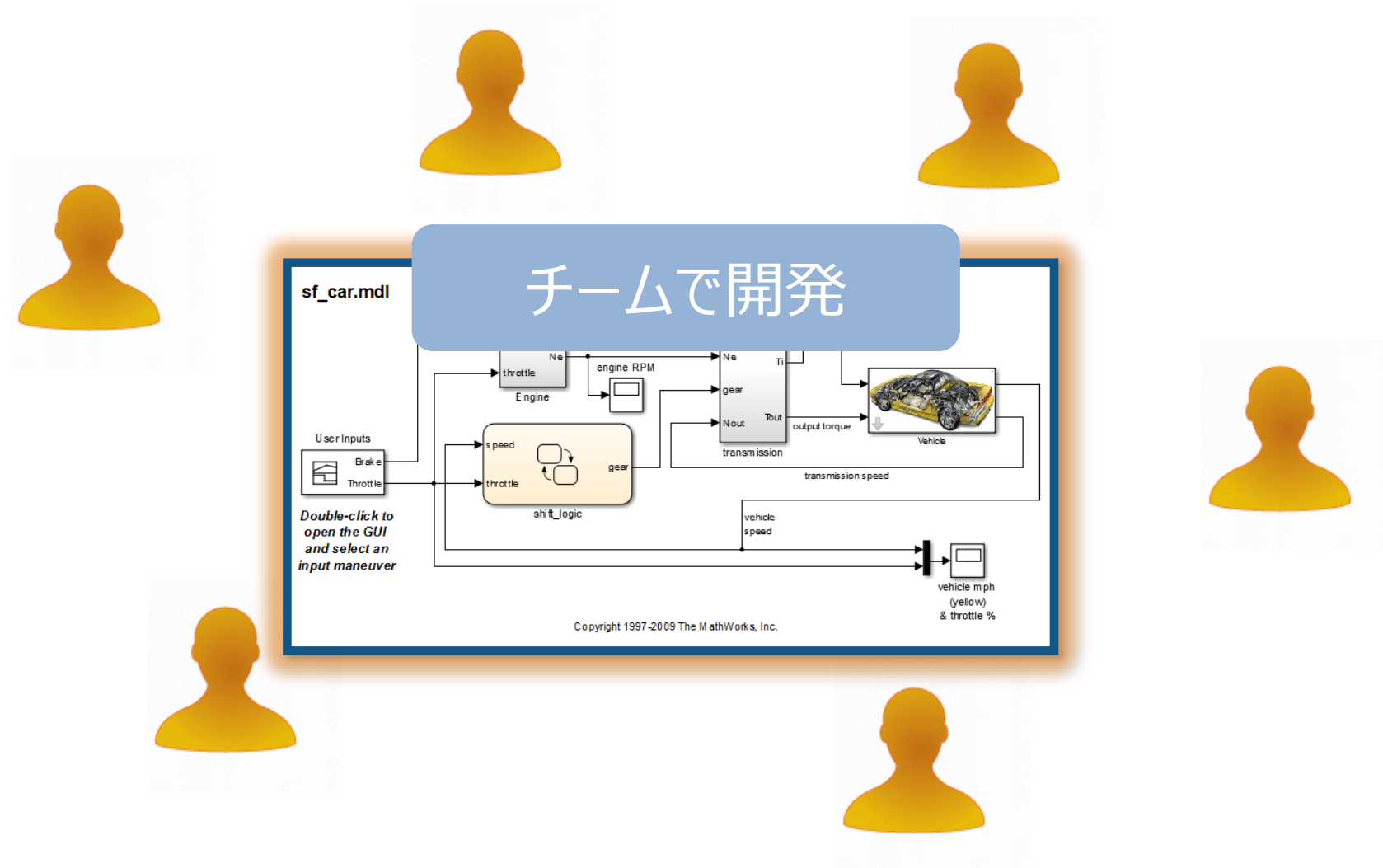
➡ Simulink プロジェクトと構成管理ツールの活用 + MBDプロセス

大規模な構成管理ツールとの使い分け（例）

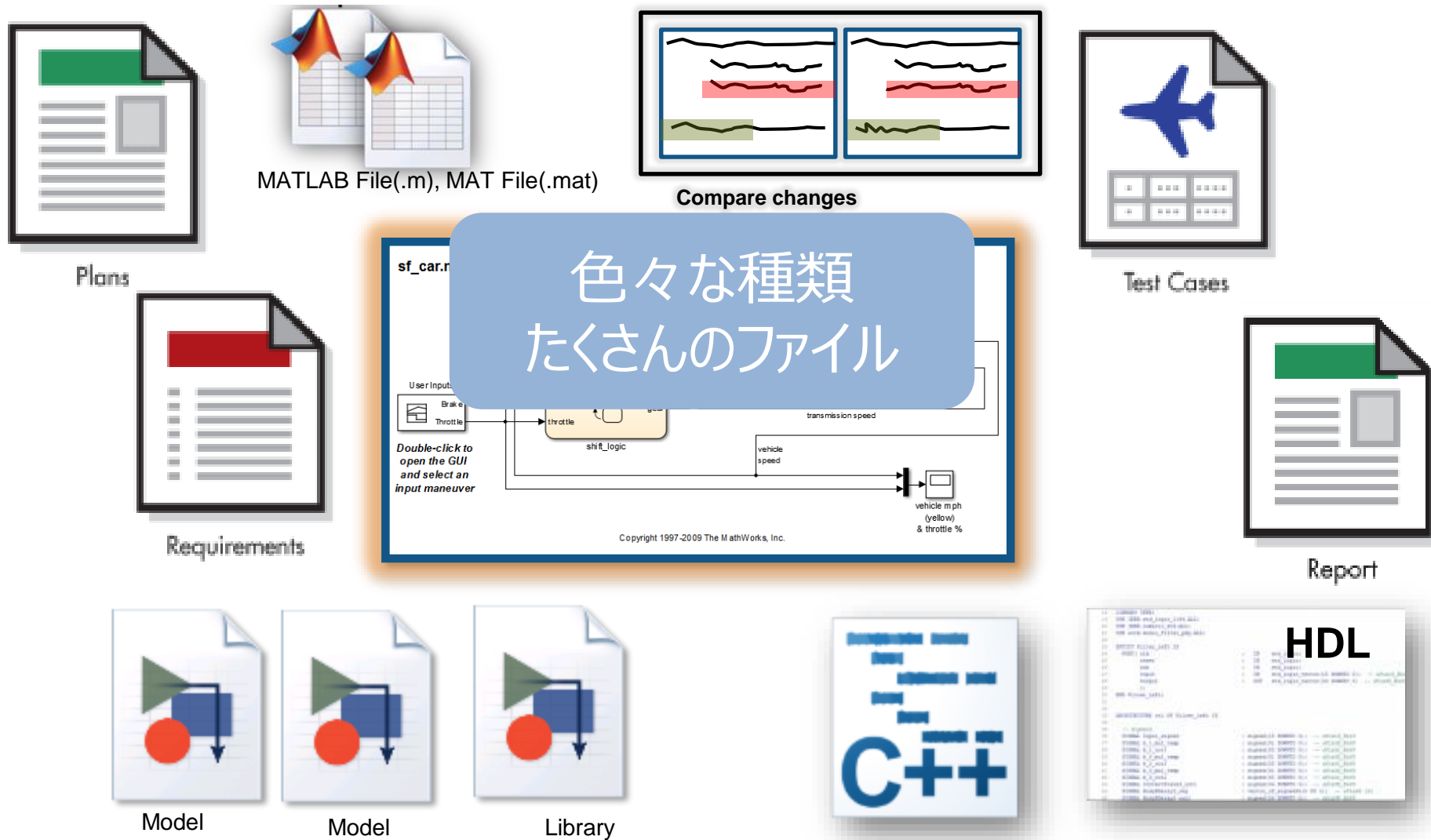
一例として、Simulink プロジェクトを日々の開発作業用ツールとして、大規模な構成管理ツールをプロセス管理用ツールとして使い分けることができます

ツール	使用例
Simulink プロジェクト	日常業務にてモデルを作成・修正する際、作業端末内でモデルを管理するために使用します。
大規模な構成管理ツール	プロセス・マイルストーン毎に、成果物（Simulinkプロジェクトで管理していたモデルや各種レポート類）をデータベースに格納するために使用します。

質問：何人の人が1つの制御設計に関わりますか？



質問：1つの制御設計にいくつのファイルがありますか？



チーム開発での課題



サンプルシナリオ

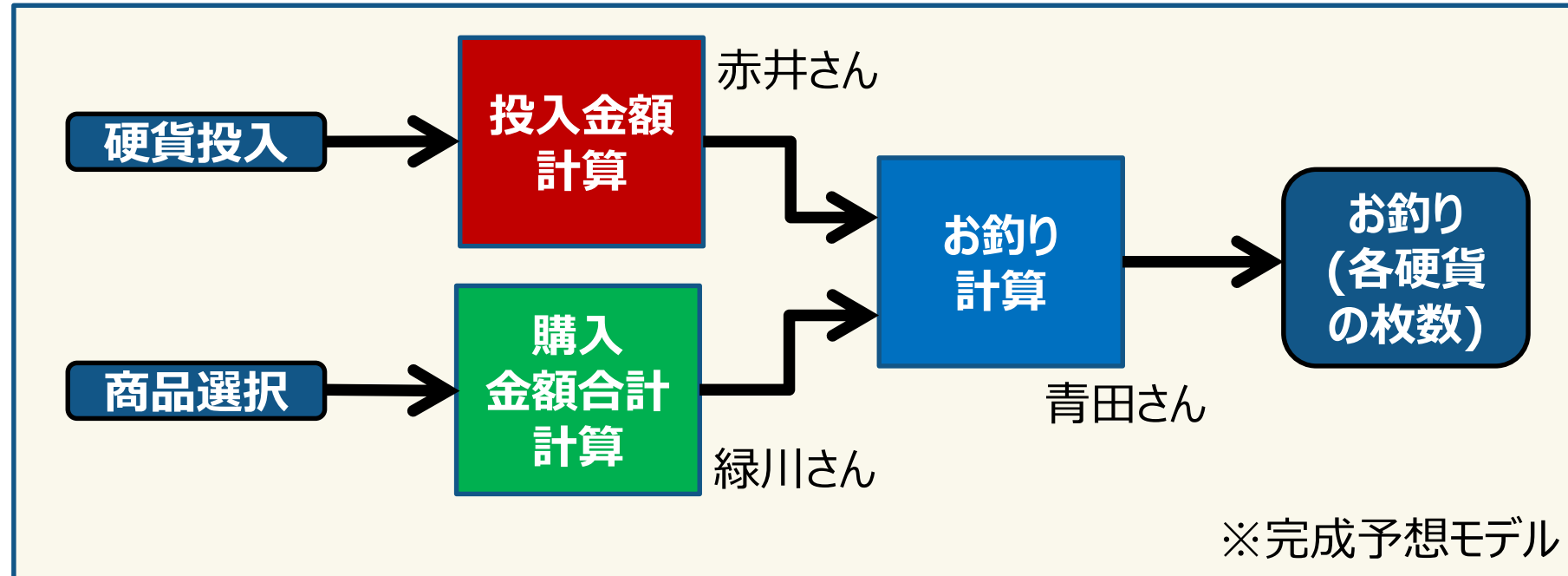
- チームメンバー
 - 赤井TL (チームリーダー)
 - 青田さん
 - 緑川さん
- プロジェクト概要
 - 「自動販売機プログラム仕様書」に基づき、自動販売機の計算プログラムを作成する
 - 投入金額の計算を行う
 - 購入商品の金額を計算する
 - お釣りを計算する



自動販売機の機能仕様

- 硬貨の投入
 - 投入された硬貨により投入金額の合計を求める。
 - 硬貨は、500円、100円、50円、10円の4種類。
- 商品の合計の計算
 - 投入金額が商品の値段以上の場合、商品を購入でき、商品の合計金額を計算する。
- 商品の値段
 - 150円, 120円, 100円から選択。パラメータで変えられるようにする。
- おつりの計算
 - 商品が選択されれば、お釣りを計算する。
 - お釣りとして出力する各硬貨の枚数を計算する。

サンプルシナリオ



■ 赤井TL の役割

- プロジェクトを新規に設定
- トップモデルを作成
- 役割分担を決める
- 作成モデルの確認をし、承認を行う

■ 青田さん、緑川さんの役割

- 仕様書に基づいて担当箇所のモデルを作成する
- コンポーネントのテストを行う
- モデルが完成したら承認依頼をする

サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

3. モデルの分割と担当の割り振り

4. プロジェクトの共有

5. 並行開発

サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

3. モデルの分割と担当の割り振り

4. プロジェクトの共有

5. 並行開発

シナリオ①：プロジェクトの開始

- 担当
 - 赤井TL
- タスク
 - 新規プロジェクトを設定
- どうやって？

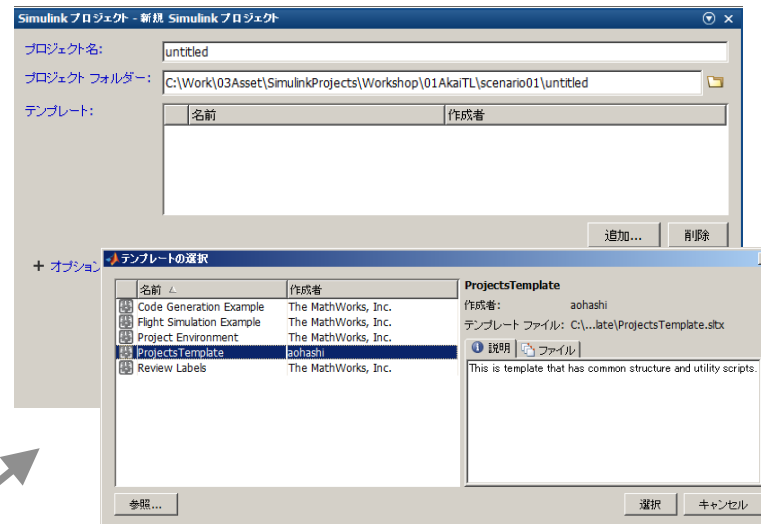


⇒ Simulink プロジェクトで開始

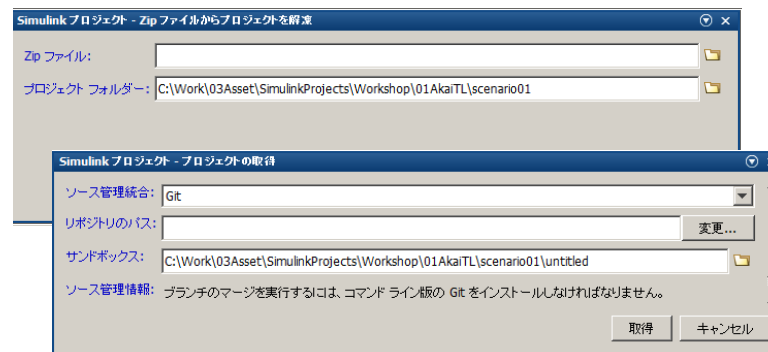
シナリオ①：プロジェクトの新規作成

「ホーム」タブ→Simulink プロジェクトを選択

空のプロジェクト/テンプレート選択



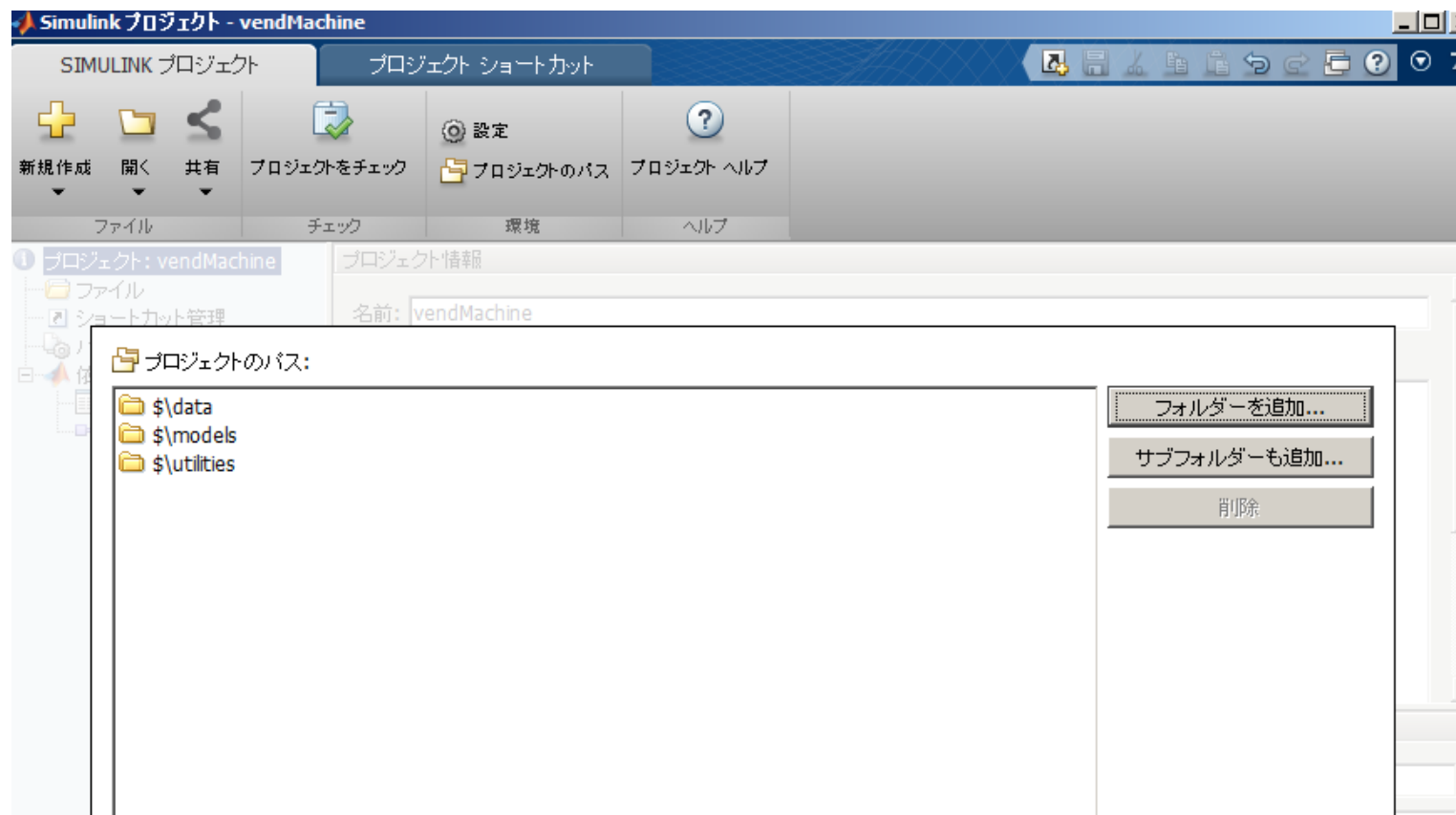
アーカイブから/ソース管理から



- プロジェクト名：
 - 新規作成するプロジェクト名を指定
- プロジェクトフォルダー：
 - プロジェクトを管理するフォルダー
- テンプレート：
 - 標準構成として登録済みのファイル
 - 作成時プロジェクト名を利用
- ZIPファイル：
 - 標準構成が保存されたファイル
 - プロジェクト名はZIP内の情報を利用
- ソース管理統合：SVN, Git
- レポジトリのパス：
 - 既存のレポジトリ名を指定
- サンドボックス：
 - プロジェクトを管理するフォルダー

シナリオ①：プロジェクトのパス指定

プロジェクトで利用するフォルダのパスを指定することで、Simulink プロジェクト起動時,終了時に自動的にパスを設定,解除をします

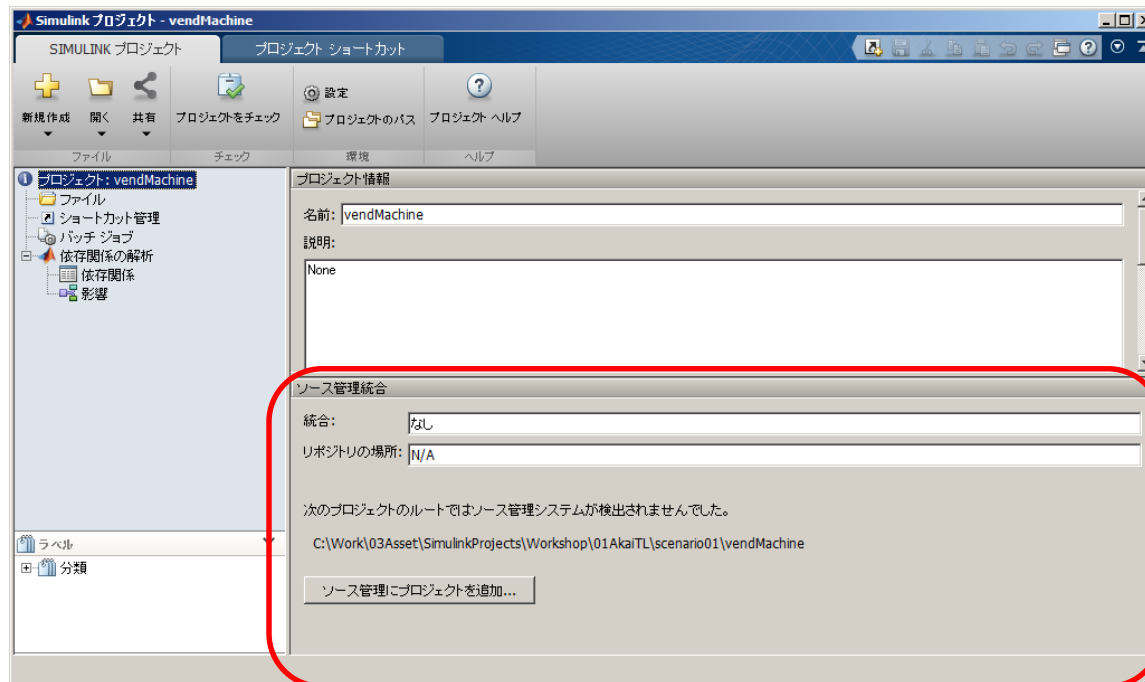


シナリオ①：ソース管理の設定

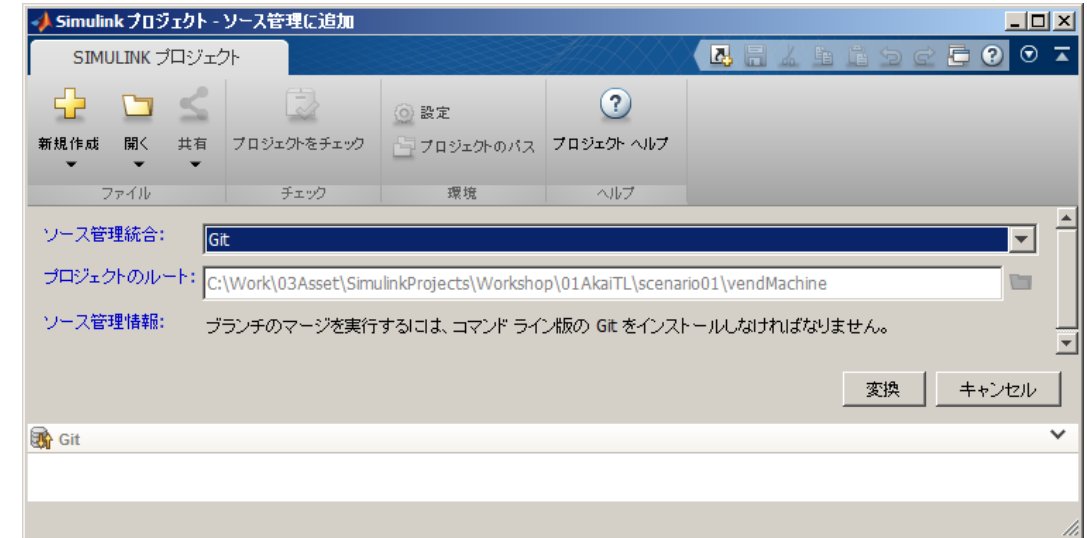
Simulink プロジェクトが利用できるソース管理ツール

- Subversion
- Git
- その他の構成管理ツール(アダプターキット利用時)

ソース管理情報

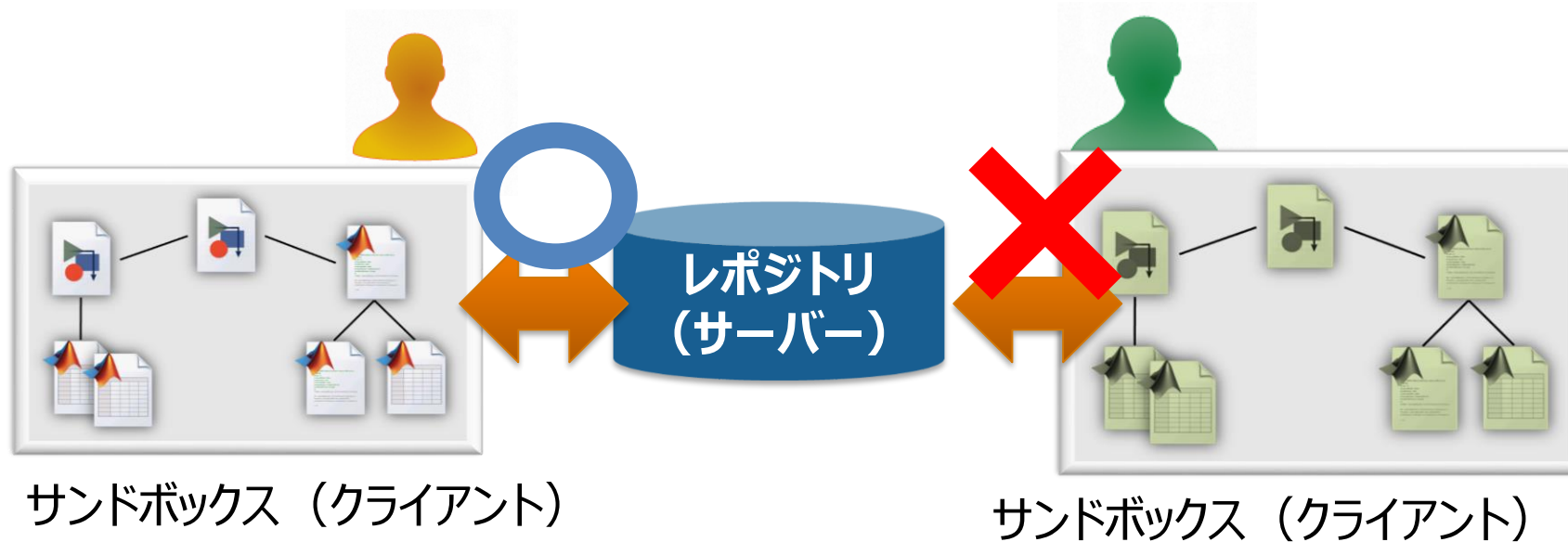
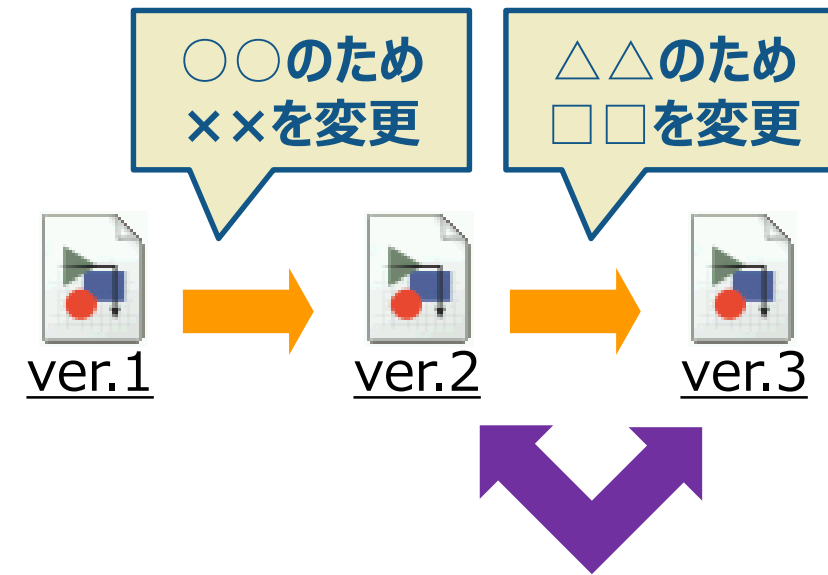


ソース管理設定画面



ソース管理ツールの機能

- バージョン管理
- ログ（更新履歴）
- 差分解析ツールの呼び出し
- ファイルの共有
- 排他処理／アクセス制御
- リリース管理／派生バージョン管理



シナリオ①：プロジェクトの開始のポイント

- テンプレートから容易にスタート
 - 組織内で標準のフォルダ構成でファイルを分類
 - 無秩序なフォルダ・ファイル構成を避ける
 - スクリプトを共有
 - よく使う処理（パスの追加等）を固定
 - 手順やノウハウを再利用、標準化
- ソース管理ツールを設定
 - ソース管理ツールの各種機能と連携
 - Subversion
 - Git
 - アダプターキット



サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

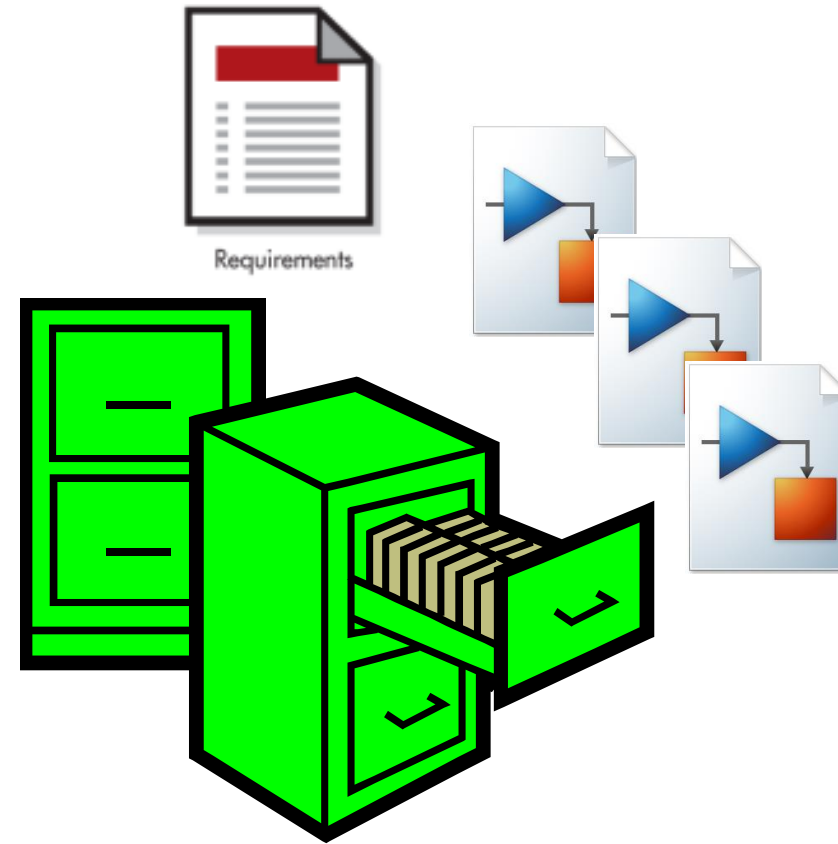
3. モデルの分割と担当の割り振り

4. プロジェクトの共有

5. 並行開発

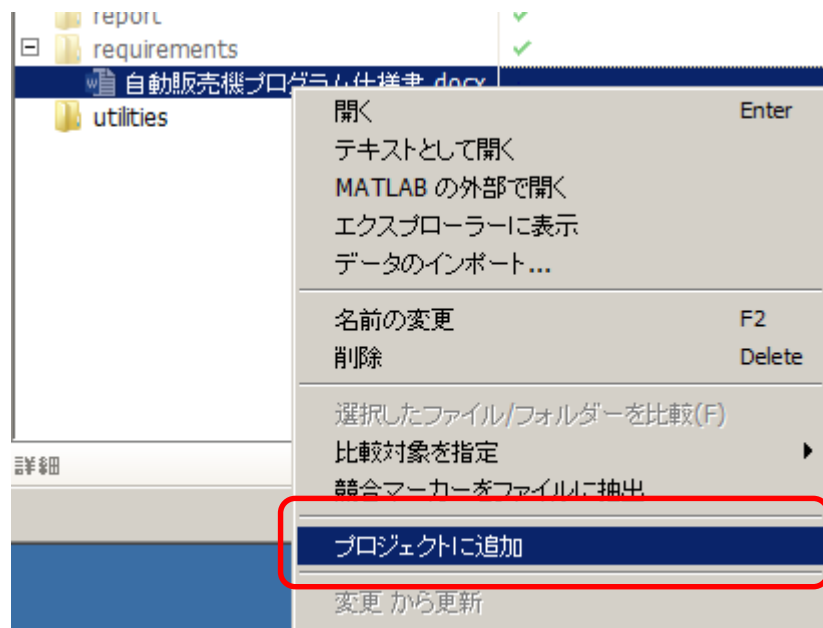
シナリオ②：ファイルの作成・編集

- 担当
 - 赤井TL
- タスク
 - 機能仕様書をプロジェクトに登録
 - 初期モデルを作成
 - リビジョンの確認



シナリオ②：ファイルをプロジェクトファイルとして追加

新規のファイルをプロジェクトして追加することで、Simulink プロジェクト内の解析対象になります



名前 ▲	状態
batch_jobs	✓
data	✓
models	✓
report	✓
requirements	✓
自動販売機プログラム仕様書.docx	•
utilities	✓

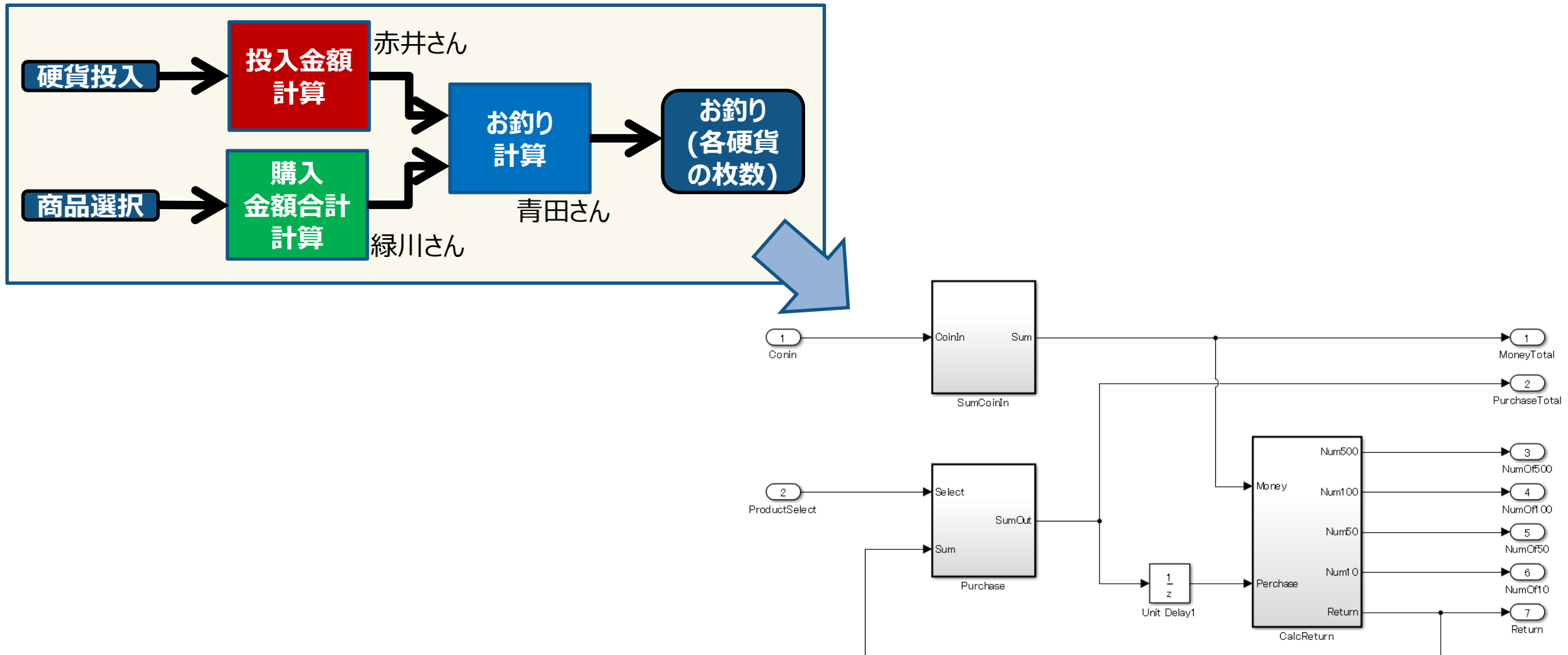
プロジェクトへ未追加の場合、状態カラムは「•」

名前 ▲	状態
batch_jobs	✓
data	✓
models	✓
report	✓
requirements	✓
自動販売機プログラム仕様書.docx	✓
utilities	✓

プロジェクトへ追加すると、状態カラムは「✓」

初期モデル

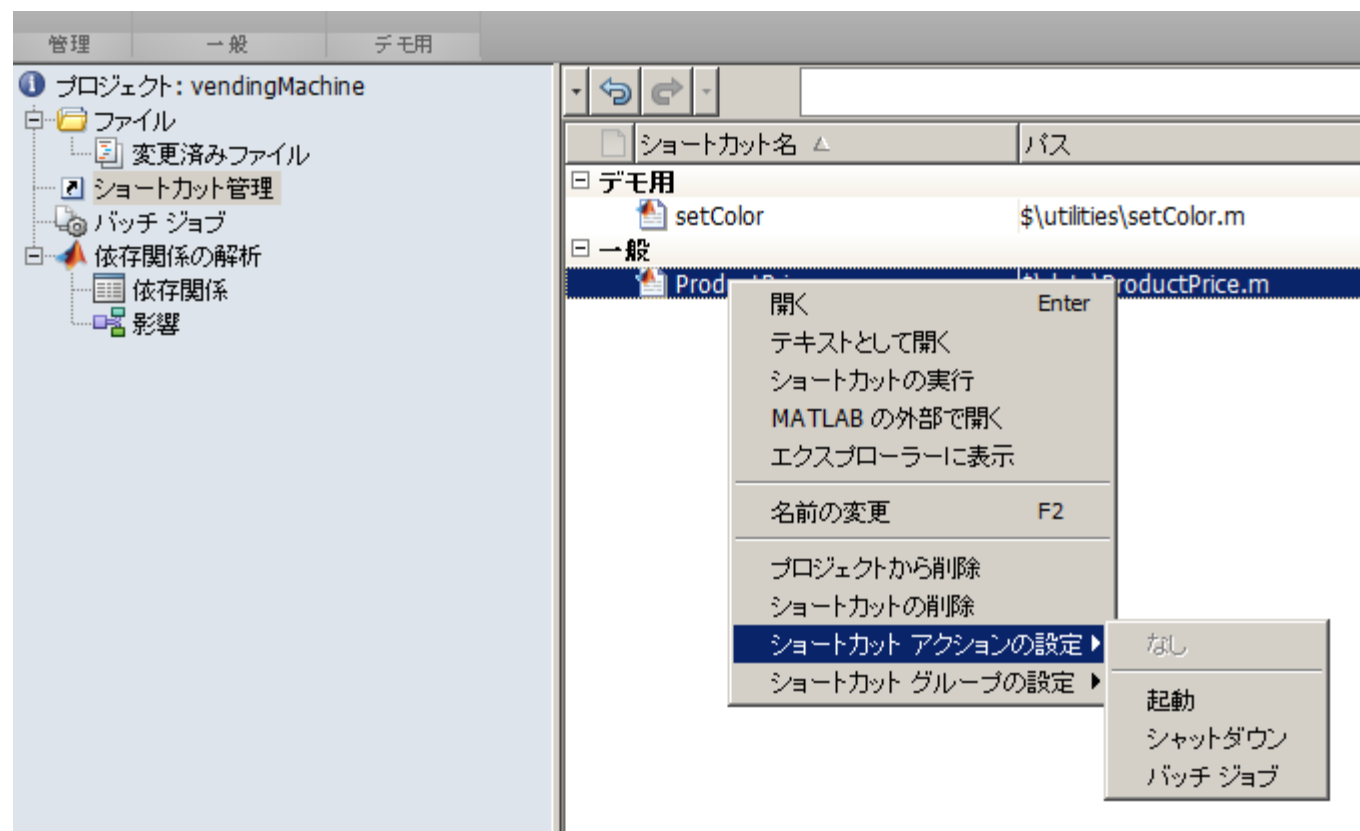
制御モデル全体のアーキテクチャを定義するモデルを作成します



シナリオ②：ショートカット設定

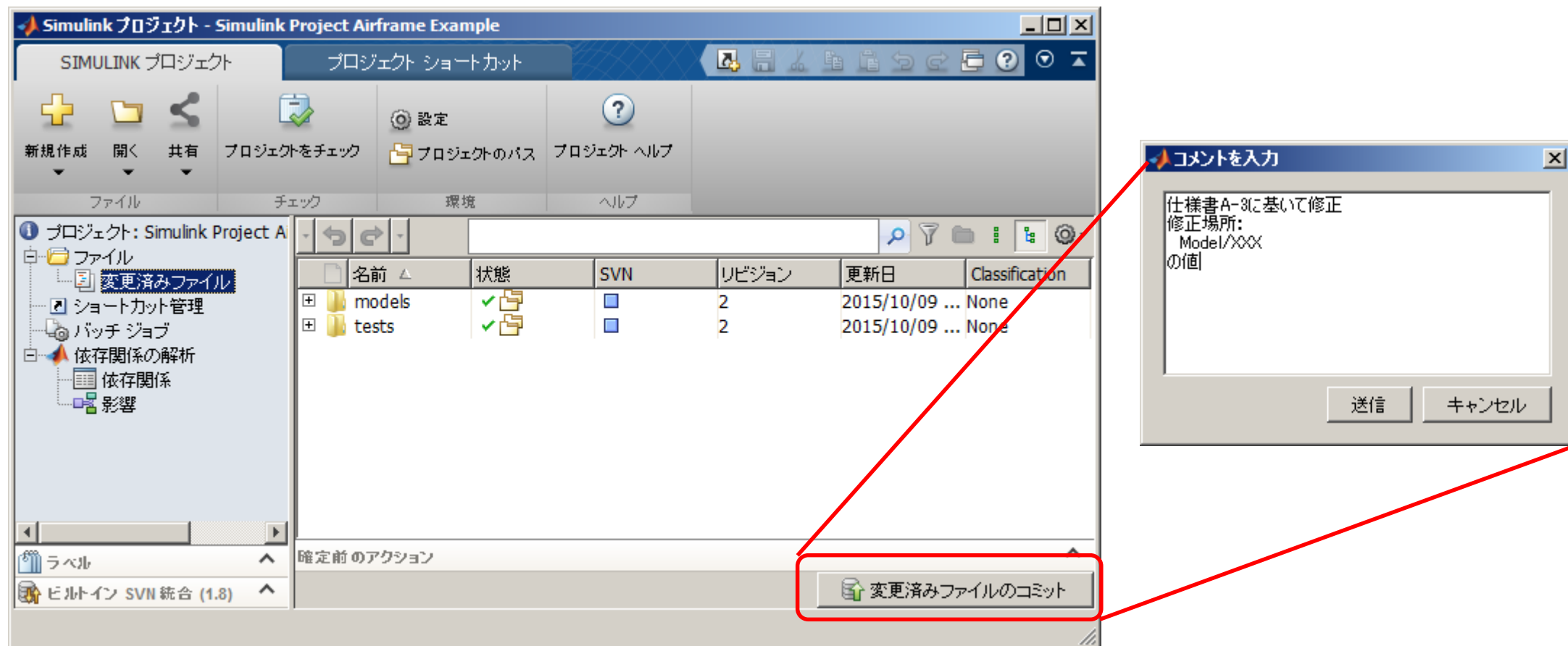
使用頻度の多いスクリプトを Simulink プロジェクトで簡便に利用するため、ショートカットへ登録します

- 起動
- シャットダウン
- バッチジョブ



シナリオ②：レポジトリの更新

チームメンバーとファイルを共有するために、サンドボックス内のファイル修正をレポジトリへ送信します



シナリオ②：リビジョンの活用

- リビジョンの表示
 - 履歴一覧を表示
- リビジョンと比較
 - 差分ツールを起動
- 元に戻る
 - 前のリビジョンの状態へ戻す

名前 ▲	状態	SVN	リビジョン
+ batch_jobs	✓	●	2
+ data	✓	●	2
- models	✓	■	2
AnalogControl.mdl	✓	●	3
DigitalControl.slx	✓	■	2
f14_airframe.slx	✓	●	2
LinearActuator.slx	✓	●	2
NonlinearActuator.mdl	✓	●	2

SVNカラム： サンドボックス内でのファイルのステータスを表示

- 最新状態
- 変更されており、レポジトリにコミットしていない状態
- + レポジトリに存在しない状態

リビジョンカラム： サンドボックス内でのリビジョン番号

リビジョンを指定した差分

指定したリビジョンと現在のファイル間の差分を表示

The first screenshot shows a file browser window with a context menu open. The menu item 'リビジョンと比較' (Compare with Revision) is highlighted with a red box. The second screenshot shows the 'Compare' dialog box with 'DigitalControl.mdl' selected. The third screenshot shows the resulting comparison of two Simulink models, highlighting differences in the 'Pitch Sensor Filter' block.

シナリオ②：ファイルの作成・編集のポイント

- プロジェクトへの追加
 - 管理するファイルと一時ファイルを区別
- ソース管理ツール機能の呼び出し
 - コミット
 - 履歴の確認
 - 差分の確認
 - 元のバージョンに戻る



サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

3. モデルの分割と担当の割り振り

4. プロジェクトの共有

5. 並行開発

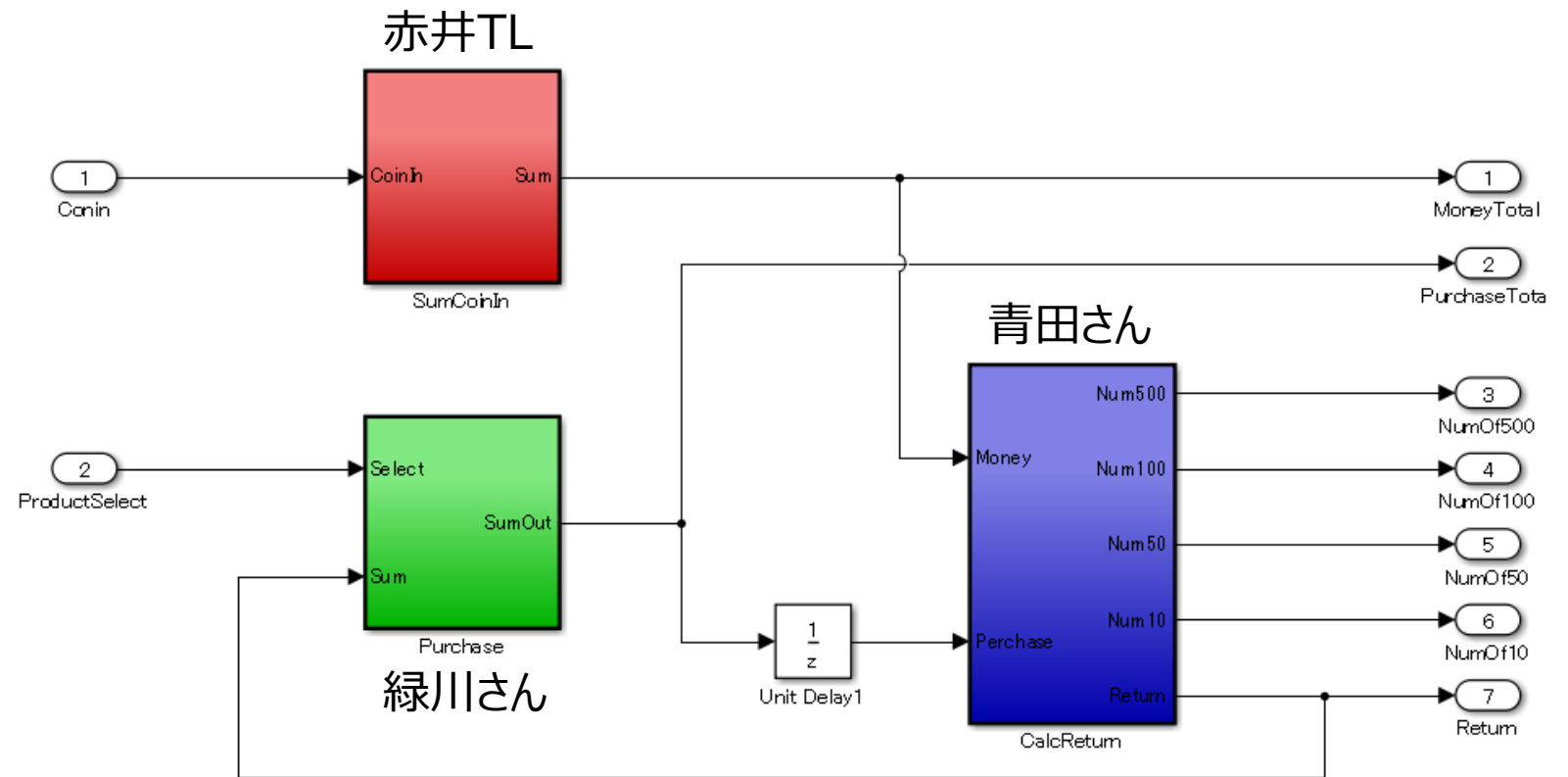
シナリオ③：モデルの分割と担当の割り振り

■ 担当

- 赤井TL

■ タスク

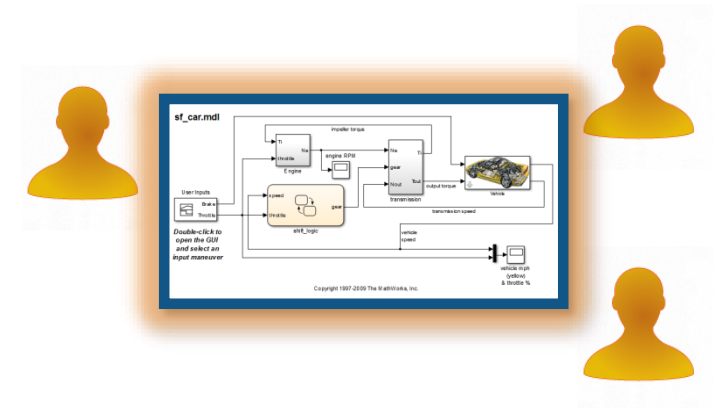
- モデルを分割
- 依存性解析
- 担当者の割り当て



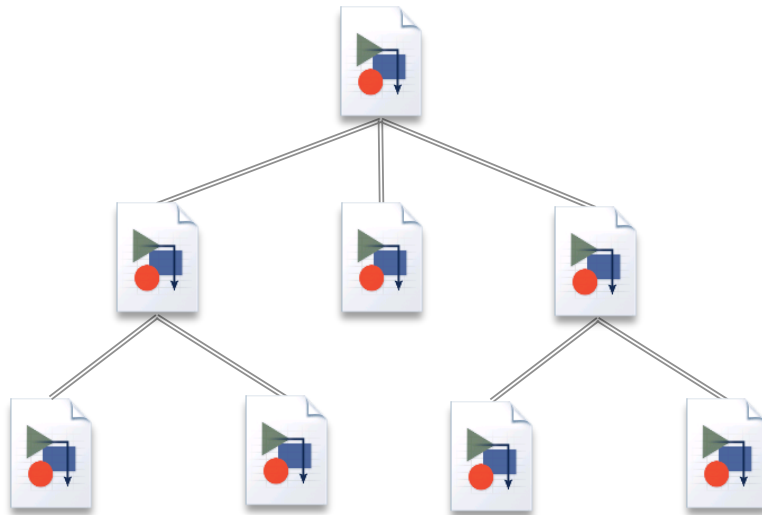
シナリオ③：デザイン分割

- **「1制御 = 1モデル」**の課題

- 同時に編集ができない
- マージが大変
- コンフリクトやデグレードが起きる可能性



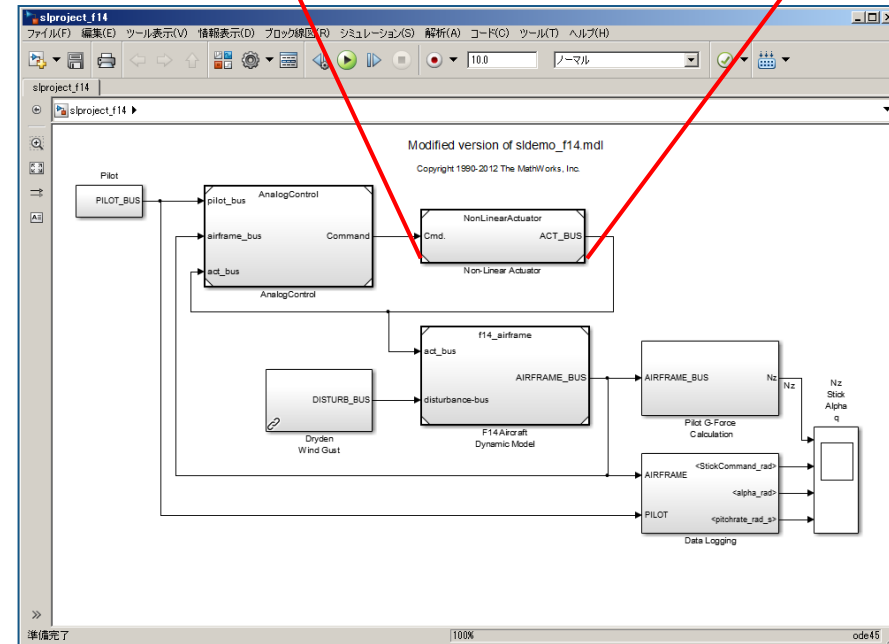
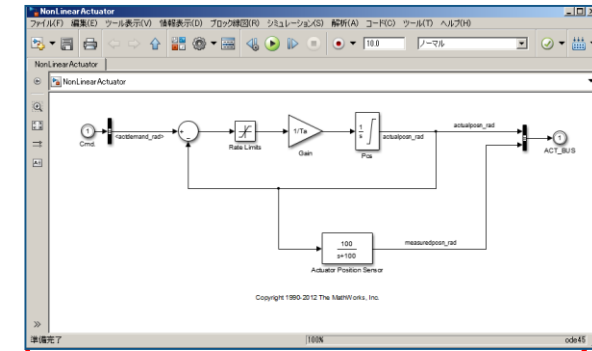
- モデルを複数人で扱うためには **複数のファイルに分割**



⇒ モデル参照

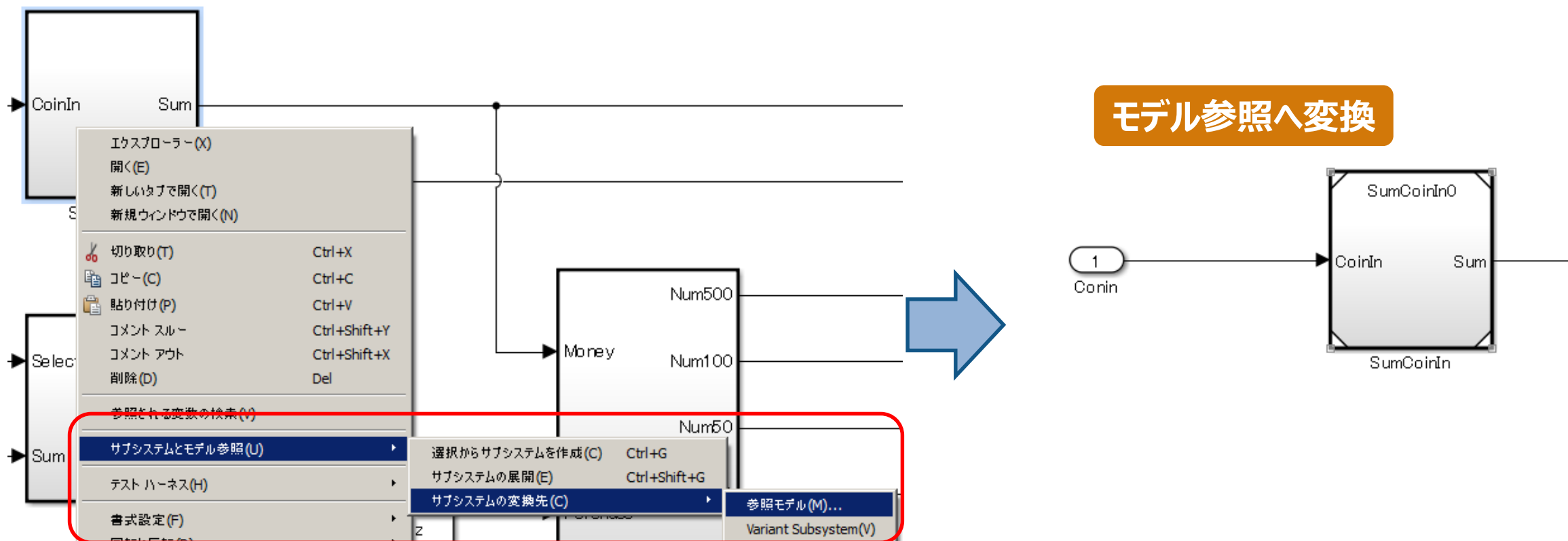
モデル参照

- モデル参照とは
 - モデル内のサブシステムとして別のモデルを指定
- 特徴
 - シミュレーションや仕向け切り替えを効率化
 - アクセラレータモード
 - パラレルビルド
 - バリエント



サブシステムからモデル参照への変換

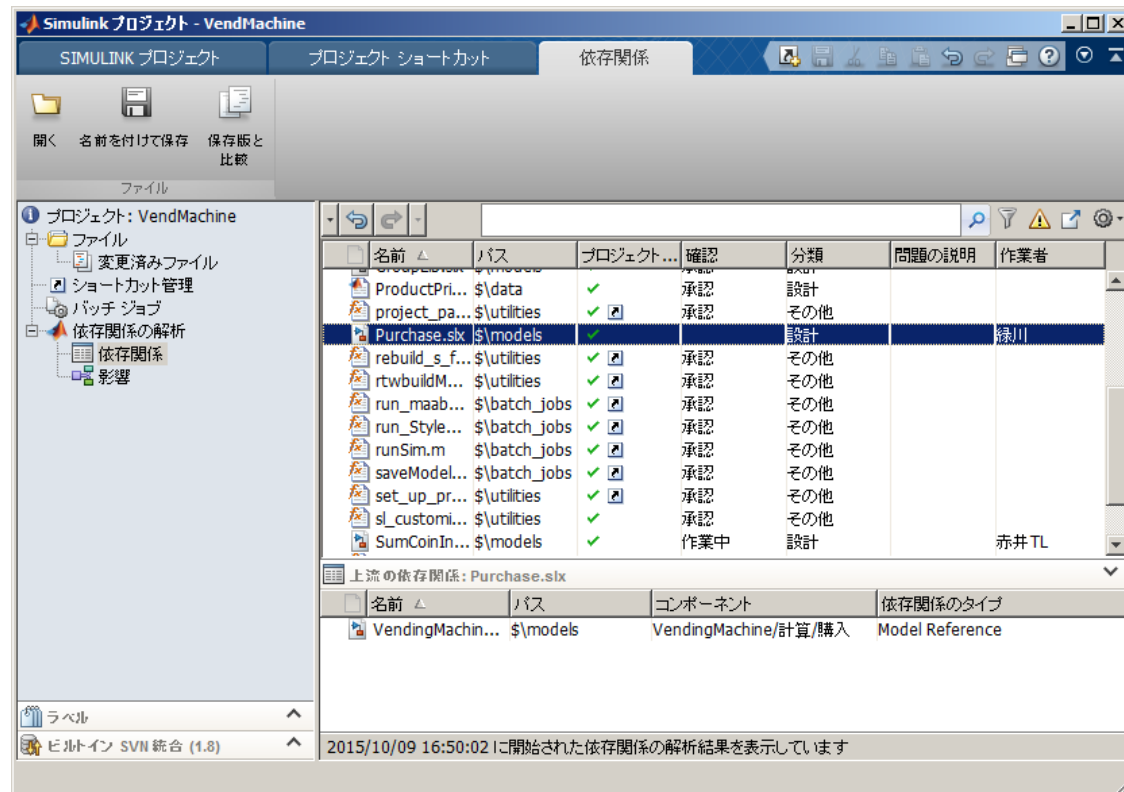
分割して開発するために、モデル内の各コンポーネントを1つのモデルとして切り出してモデル参照へ変換



シナリオ③：依存性解析

デザイン分割した結果、モデル参照関係を確認が可能

- プロジェクトに追加されているファイルが解析対象
- モデルだけでなく、MATLAB ファイルの参照関係も含む

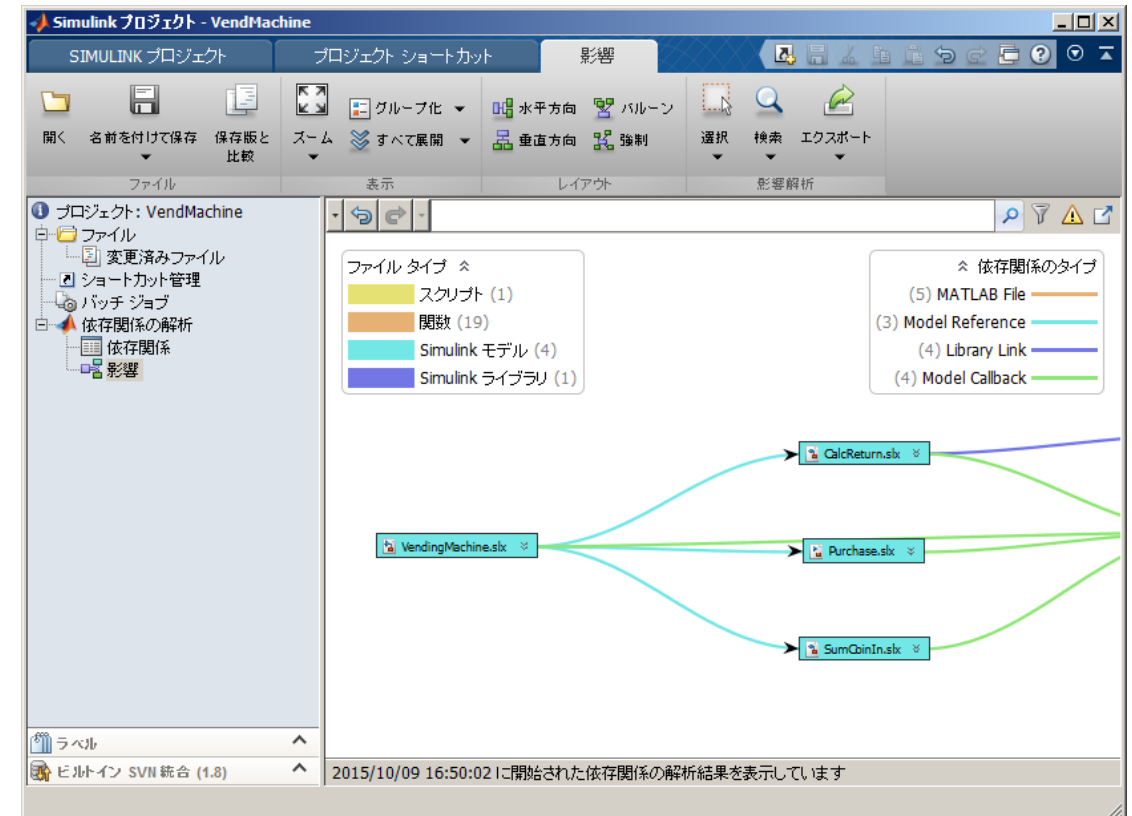


名前	パス	プロジェクト...	確認	分類	問題の説明	作業中
ProductPri...	\$\data		承認	設計		
project_pa...	\$\utilities		承認	その他		
Purchase.slx	\$\models		承認	設計		緑川
rebuild_s_f...	\$\utilities		承認	その他		
rtwbuildM...	\$\utilities		承認	その他		
run_maab...	\$\batch_jobs		承認	その他		
run_Style...	\$\batch_jobs		承認	その他		
runSim.m	\$\batch_jobs		承認	その他		
saveModel...	\$\batch_jobs		承認	その他		
set_up_pr...	\$\utilities		承認	その他		
sl_customi...	\$\utilities		承認	その他		
SumCoinIn...	\$\models		作業中	設計		赤井TL

上流の依存関係: Purchase.slx

名前	パス	コンポーネント	依存関係のタイプ
VendingMachin...	\$\models	VendingMachine/計算/購入	Model Reference

2015/10/09 16:50:02 に開始された依存関係の解析結果を表示しています



ファイル タイプ

- スクリプト (1)
- 関数 (19)
- Simulink モデル (4)
- Simulink ライブラリ (1)

依存関係のタイプ

- (5) MATLAB File
- (3) Model Reference
- (4) Library Link
- (4) Model Callback

2015/10/09 16:50:02 に開始された依存関係の解析結果を表示しています

シナリオ③：カテゴリ/ラベル

- ファイルのカテゴリ分け設定が可能

ラベルで担当者割り振りを明記

名前	確認	分類	SVN	リビ...	担当者
SumCoinIn.slx	✓	設計	●	5	赤井TL
Purchase.slx	✓	設計	●	5	緑川
CalcReturn.slx	✓	設計	●	5	青田

カテゴリ/ラベル一覧

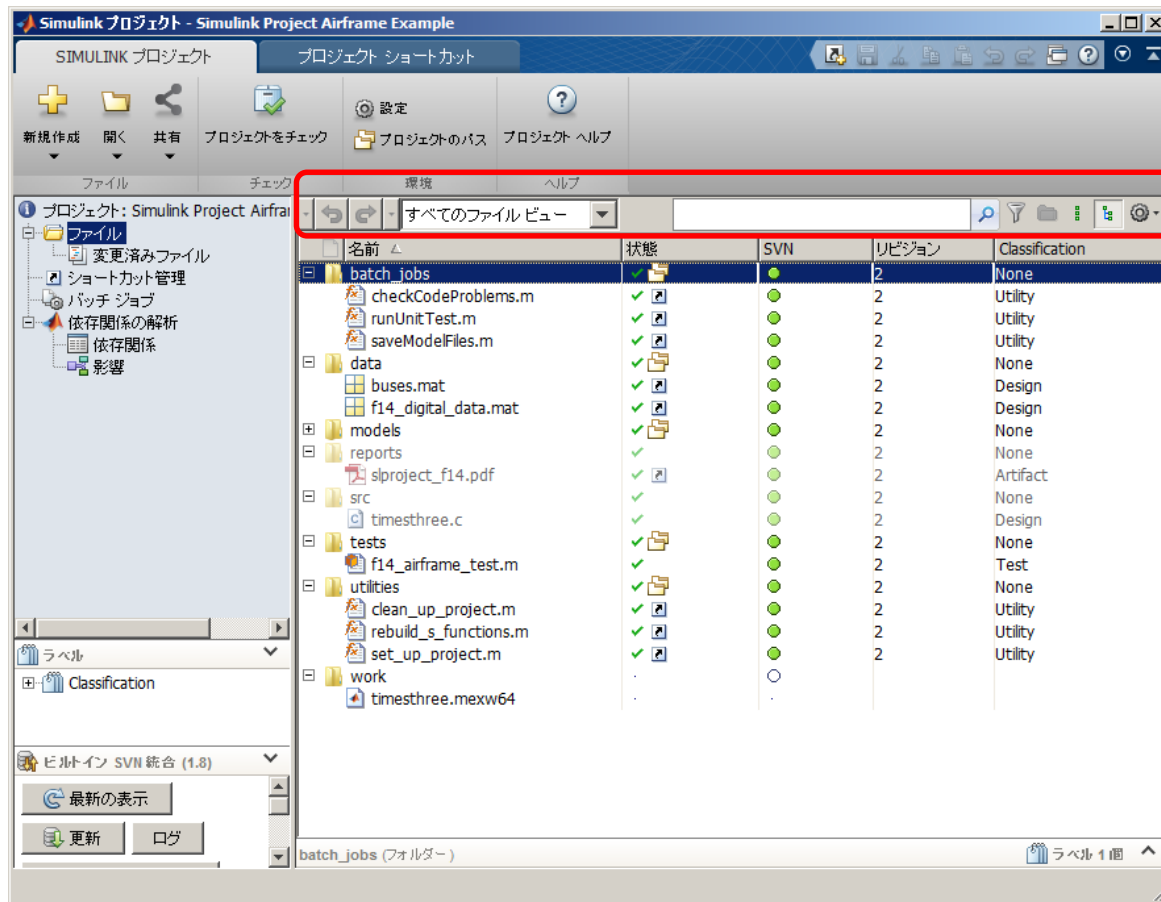
The screenshot shows the Simulink Project - VendMachine window. The left pane displays the project structure with folders like 'batch_jobs', 'data', and 'models'. The right pane shows the project settings, including the 'モデルのバージョン' (Model Version) section. A dialog box is open for assigning a label to the 'CalcReturn.slx' file. The dialog has fields for '作業中' (In Progress), '承認' (Approved), '破棄' (Discard), and '追加情報が必要' (Additional information required). The '作業中' field is currently set to 'スケルトンモデル' (Skeleton Model). The '承認' field is set to '作業中' (In Progress). The '破棄' field is set to '承認' (Approved). The '追加情報が必要' field is set to '追加情報が必要' (Additional information required). The '作業中' field is currently set to 'スケルトンモデル' (Skeleton Model). The '承認' field is set to '作業中' (In Progress). The '破棄' field is set to '承認' (Approved). The '追加情報が必要' field is set to '追加情報が必要' (Additional information required).

ドラックでラベル追加

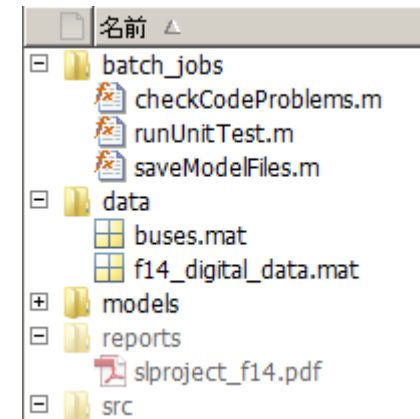
ラベルにコメントが追加可能

シナリオ③：ファイルリスト

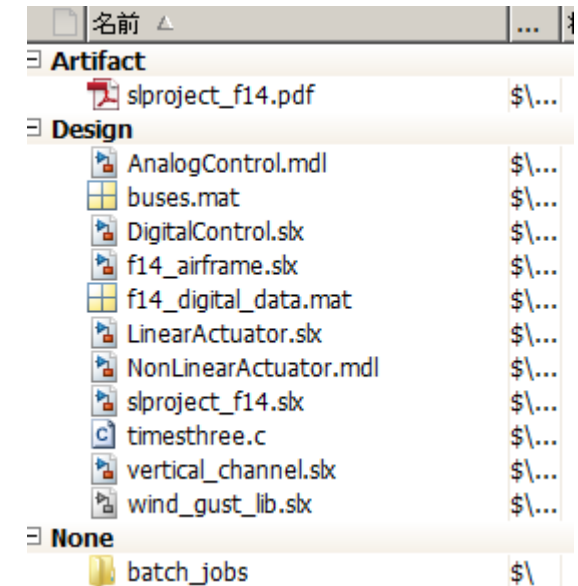
- すべてのファイルビュー：作業用の一時ファイルを含む
- プロジェクトファイルビュー：管理するファイル



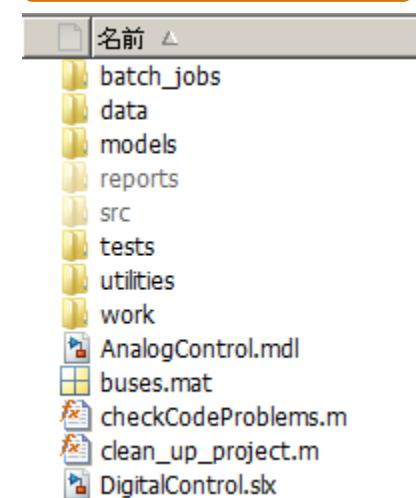
ツリー表示



グループ表示



リスト表示



シナリオ③：モデルの分割と担当の割り振りのポイント

- モデルの分割
 - モデル参照 => モデルを複数のファイルに分割
 - モデル参照による大規模モデルの効率化
 - アクセラレータモード、モデルバリエーション
- 依存性解析ツール
 - 必要なファイルを検索
 - 参照関係をグラフ表示
- カテゴリ/ラベル
 - ファイルを分類
 - ソートやフィルタが可能



サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

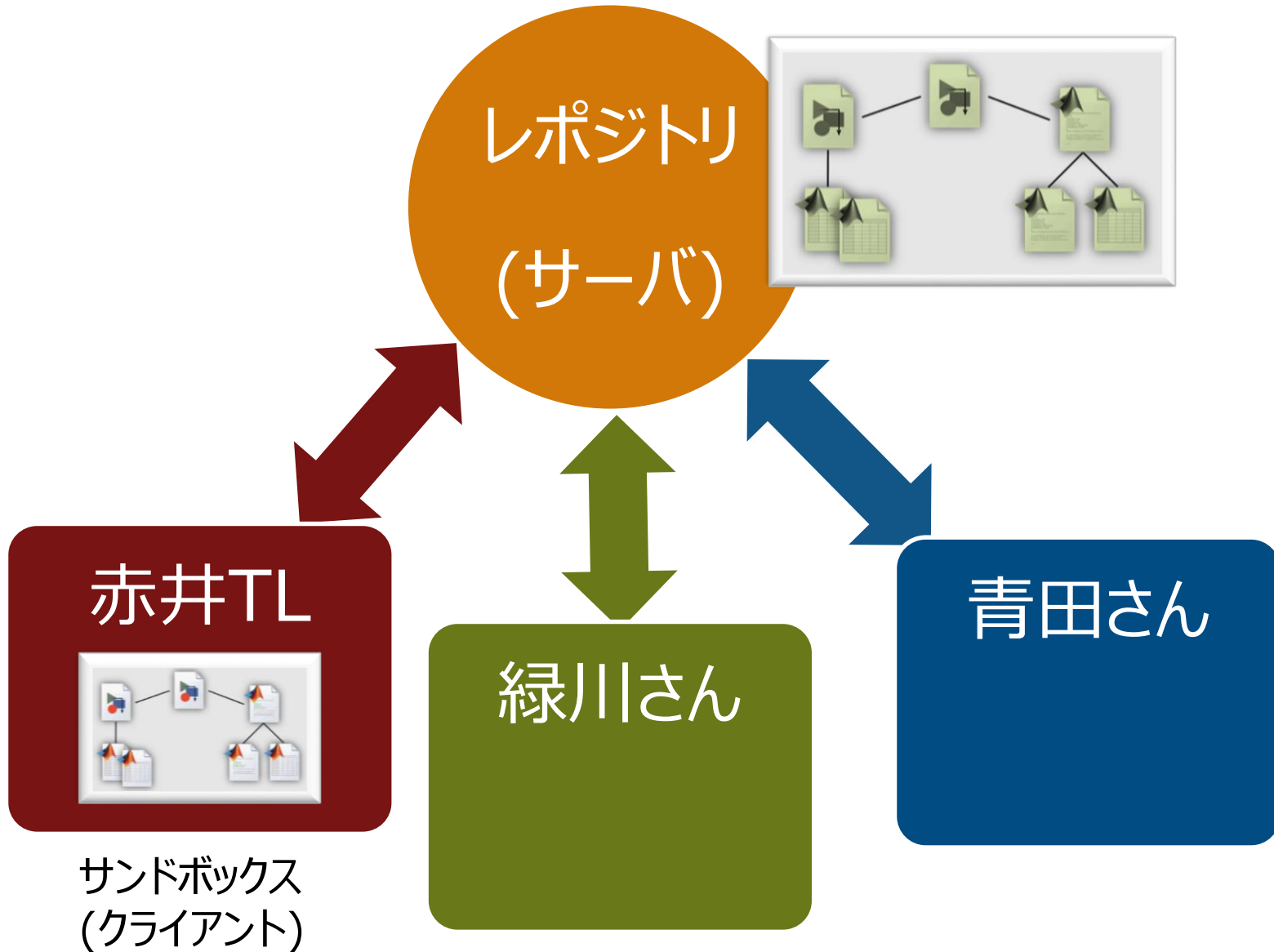
3. モデルの分割と担当の割り振り

4. プロジェクトの共有

5. 並行開発

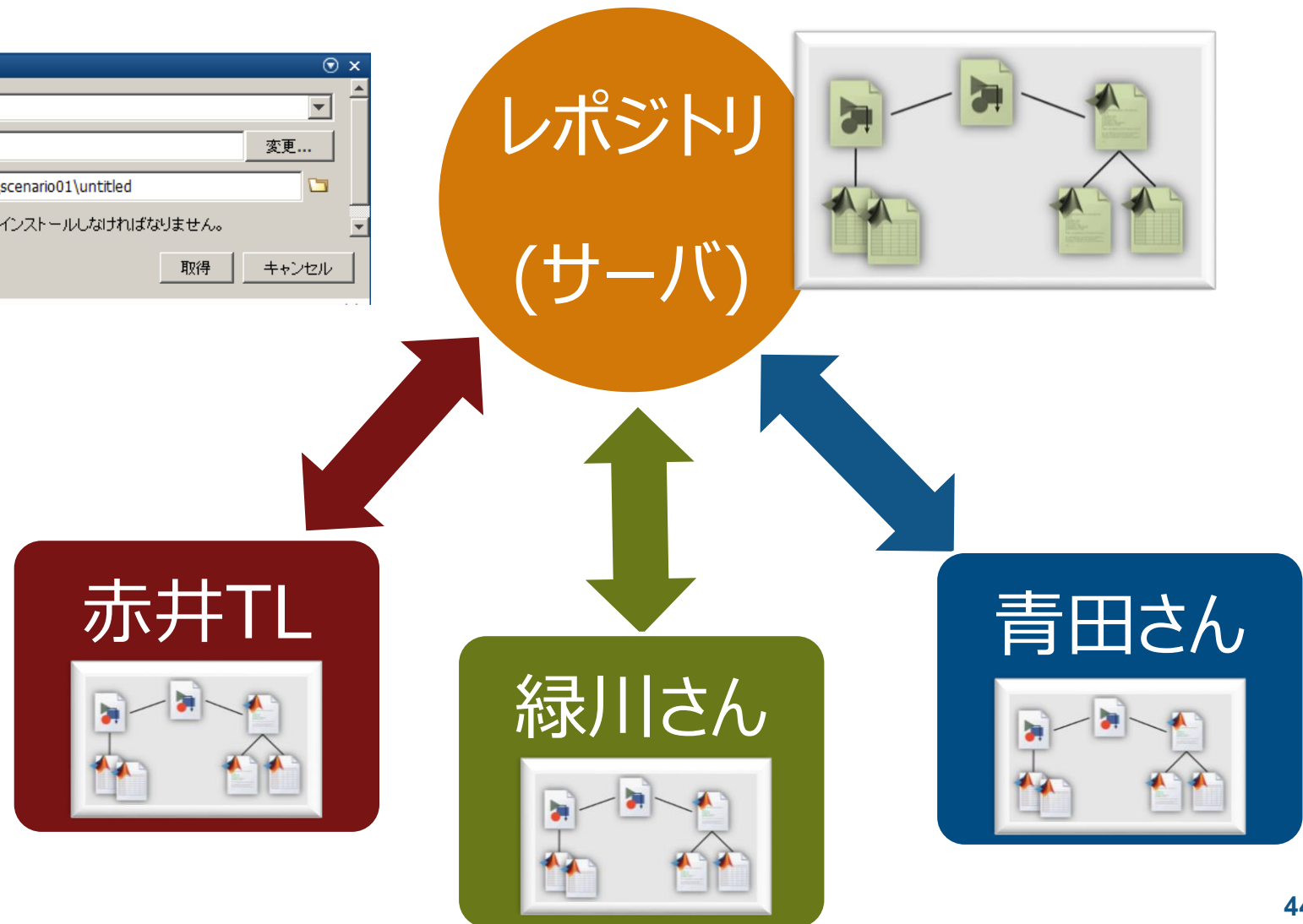
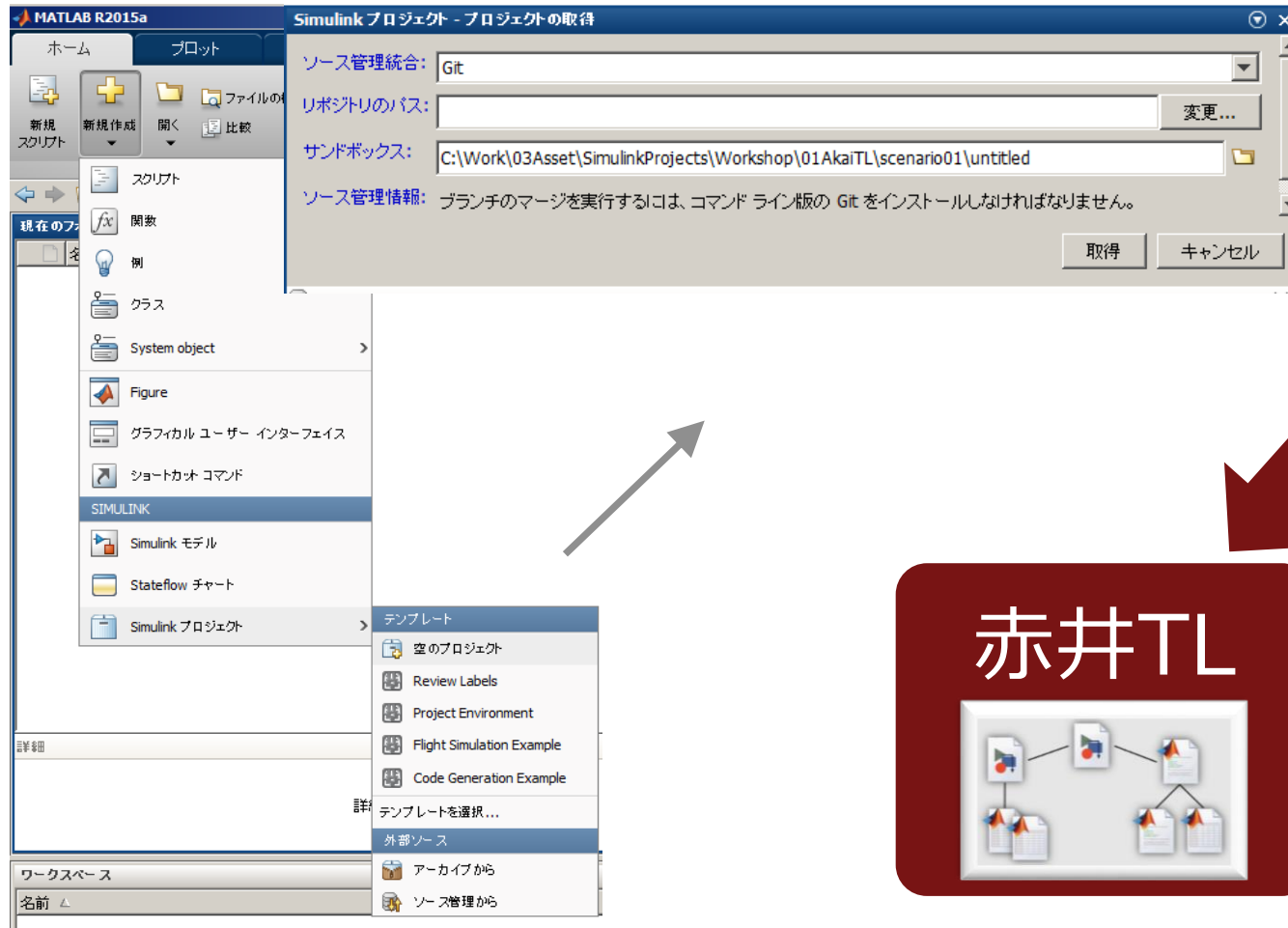
シナリオ④：プロジェクトの共有

- 担当
 - 青田さん、緑川さん
- タスク
 - プロジェクトファイルを取得
 - 最新版のファイルを取得



シナリオ④：プロジェクトファイルを取得

レポジトリに登録されているファイルをSimulink プロジェクトでローカルフォルダへ取得します



シナリオ④：最新ファイルの取得

- 更新：レポジトリから最新のファイルを取得

The screenshot shows the Simulink Project - VendMachine interface. The '更新' (Update) button is highlighted with a red box. The interface includes a sidebar with project navigation, a main table of project files, and a bottom section for model details and preview.

名前	状態	SVN	リビジョン	確認	作業中	分類
batch_jobs	✓	●	39			なし
data	✓	●	2			なし
models	✓	■	36			なし
CalcRet...	✓	■	35	作業中: スケル...	青田	設計
GroupLi...	✓	●	23	承認		設計
Purchas...	✓	●	36		緑川	設計
SumCoi...	✓	●	35	作業中	赤井 TL	設計
Vending...	✓	■	32	承認		設計
report	✓	●	2			なし
requirements	✓	●	37			なし
utilities	✓	●	39			なし
work	✓	●	2			なし

Model details section:

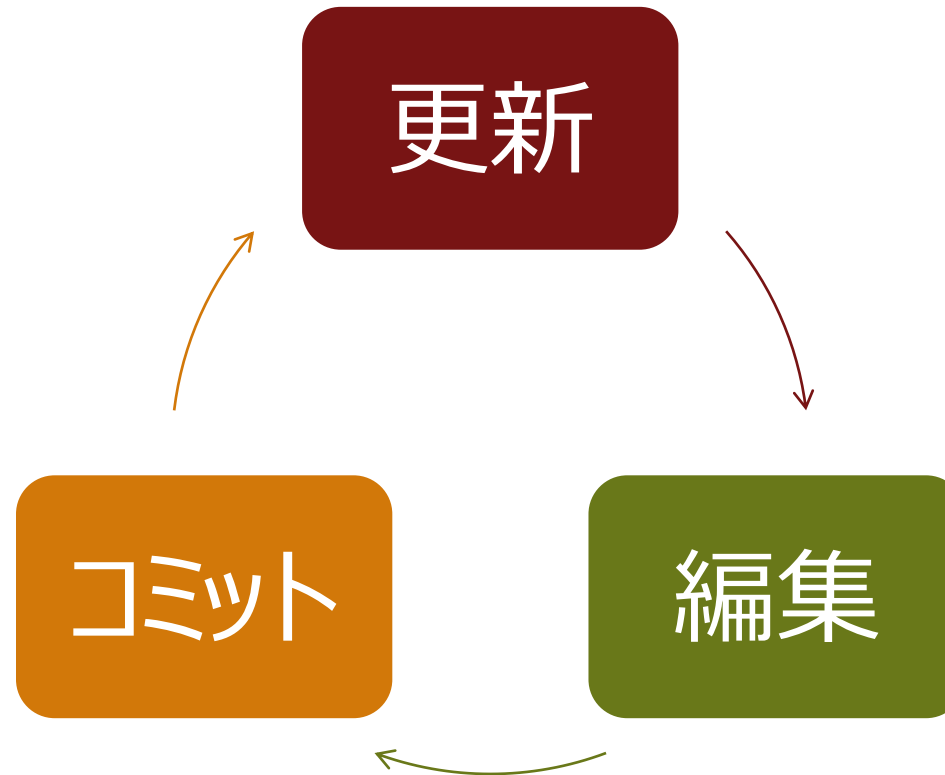
モデルのバージョン: 1.145
 次の Simulink バージョンに保存: R2015a
 最終編集者:

Preview section:

作業中: 青田
 適用 キャンセル

シナリオ④：プロジェクトの共有のポイント

- レポジトリのパスで共有
- 最新版を取得：「更新」ボタン



サンプルシナリオ

1. プロジェクトの開始

2. ファイルの作成・編集

3. モデルの分割と担当の割り振り

4. プロジェクトの共有

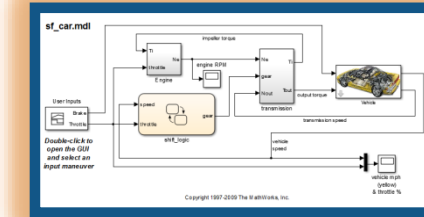
5. 並行開発

シナリオ⑤-1： 並行開発

- 担当
 - 赤井TL
 - 青田さん
- タスク
 - 赤井TLが要求リンクを設定
 - 青田さんが要求書を確認
 - 青田さんがモデルを編集



Requirements



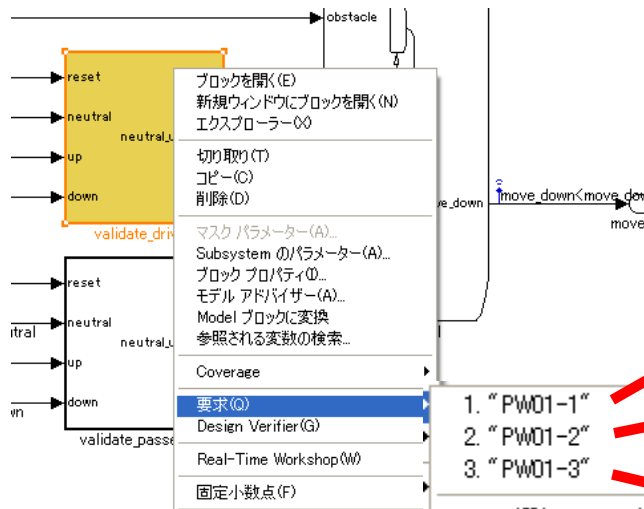
モデルと仕様書のトレーサビリティ設定とは？

Simulinkモデルと仕様書(要求仕様や検証仕様)の対応関係を紐づける設定です。

- モデルの構成要素(ブロック、サブシステム、Stateflow オブジェクトなど)と、ドキュメントファイルとのリンク設定をします。
- ドキュメントファイルは Word や Excel, Doors が利用できます。

モデルからドキュメントへ

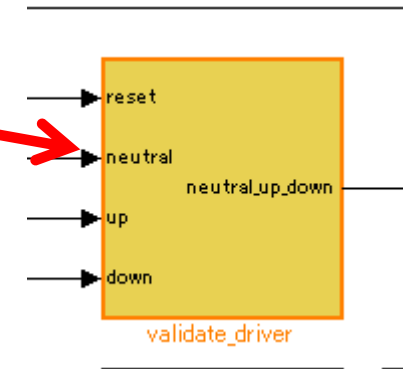
ドキュメントからモデルへ



要求名を選択すると、ドキュメント内の該当箇所がハイライトされる

PW01 スイッチ 操作	
PW01-1	運転席のダウンスイッチ押下時に障害物を検知した場合は動作
PW01-1-1	窓の下端到達を検知した場合
PW01-1-2	ダウンスイッチはアップスイッチ
PW01-1-3	ダウンスイッチが1秒以上継続アップ命令がない限りダウン命令
PW01-1-4	
PW01-2	運転席のアップスイッチ押下時に障害物を検知した場合は動作
PW01-2-1	窓の上端到達を検知した場合
PW01-2-2	アップスイッチはニュートラルより
PW01-2-3	アップスイッチが1秒以上継続ダウン命令がない限りアップ命令
PW01-2-4	
PW01-3	

ActiveXのボタン(Simulinkのアイコン)を押下すると、該当のモデル箇所がハイライトされる



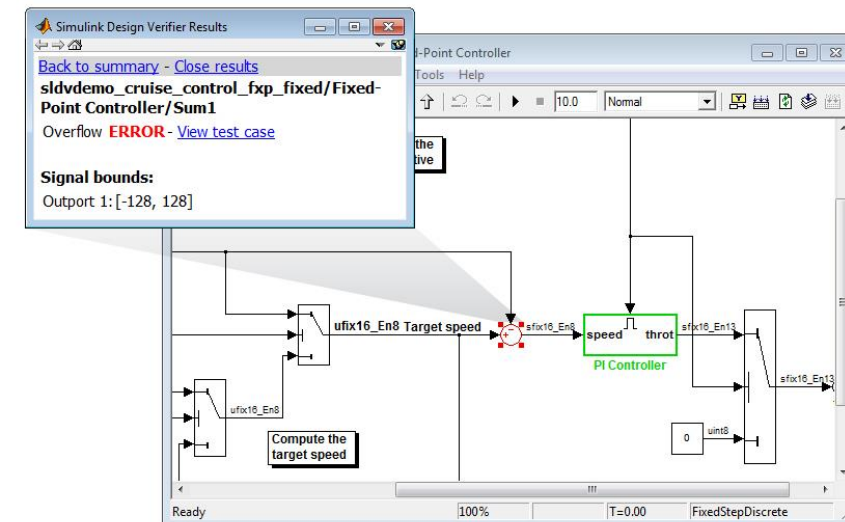
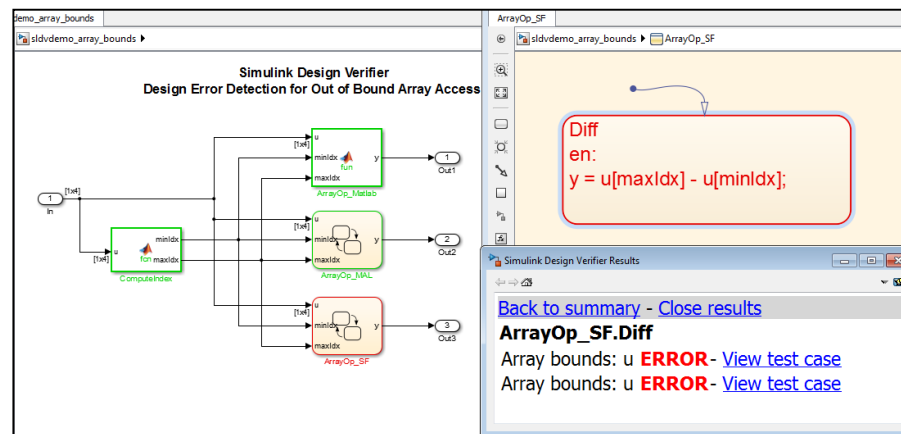
シナリオ⑤-2： 並行開発

- 担当
 - 赤井TL
 - 緑川さん

- タスク
 - 緑川さんがモデルを編集
 - 緑川さんがモデルをチェック
 - 設計エラー検出
 - スタイルガイドチェック
 - 緑川さんがモデルをレビュー依頼
 - 赤井TLがモデルをチェック
 - シミュレーション
 - 緑川さんが赤井TLのモデルを変更（コンフリクト）

設計エラー検出とは？

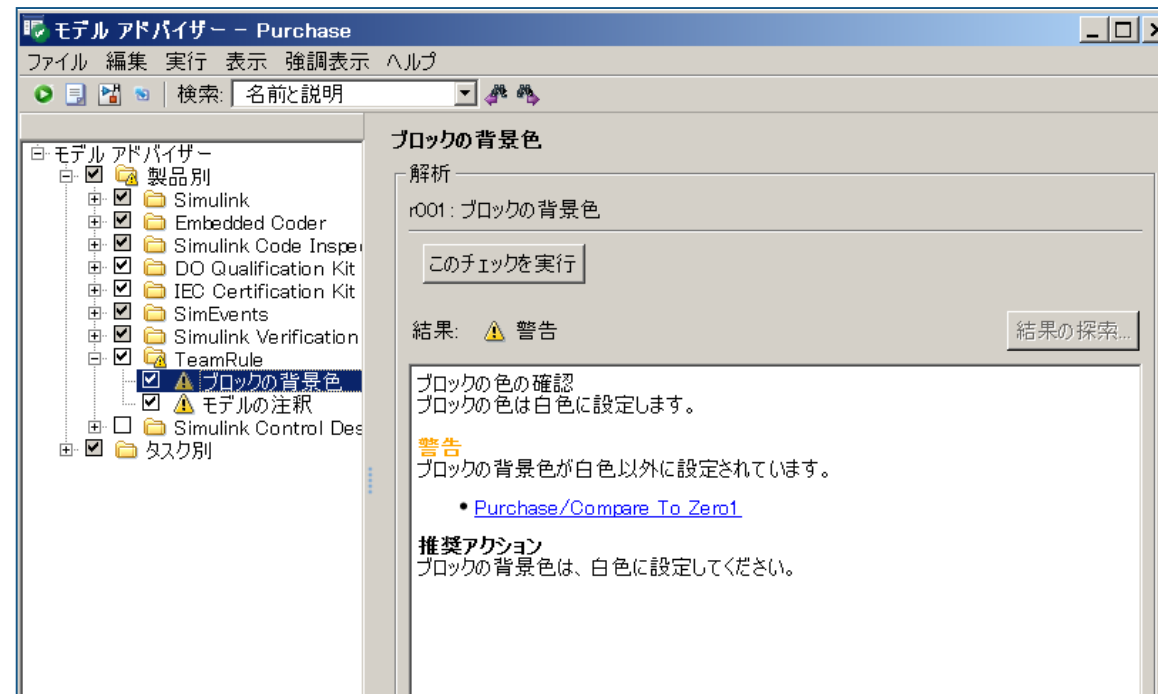
- 静的解析によって設計の矛盾を検出が可能
 - デッドロジック
 - 整数のオーバーフロー
 - ゼロ除算
 - 指定された最小/最大中間地をチェック
 - 範囲外配列アクセス
- 違反パターンがある場合はその入力ケースを作成



※設計エラー検出は、Simulink Design Verifierの機能

スタイルガイドチェック : モデルアドバイザー

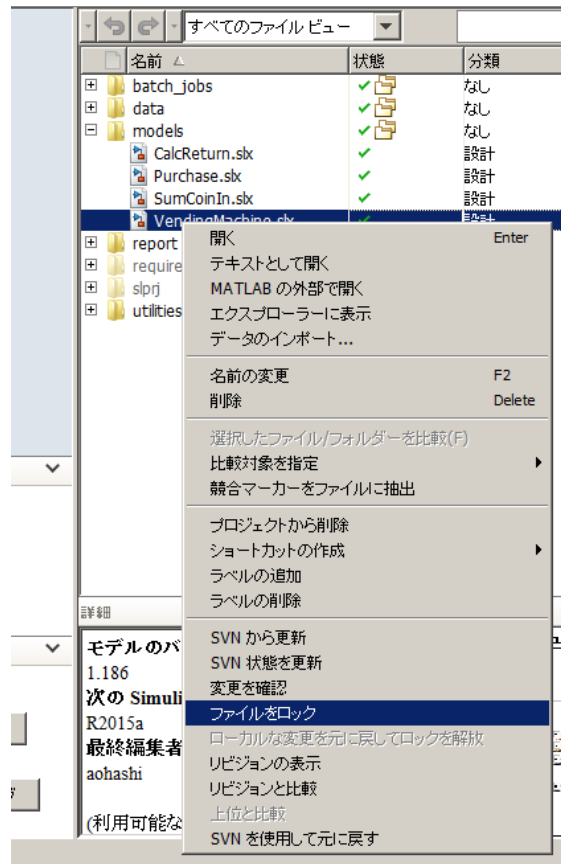
- Simulinkに付属の静的チェッカーであり、モデルスタイルガイドに準拠しているかをチェックが可能
- MAABスタイルガイドのチェックやカスタムルールの作成が可能



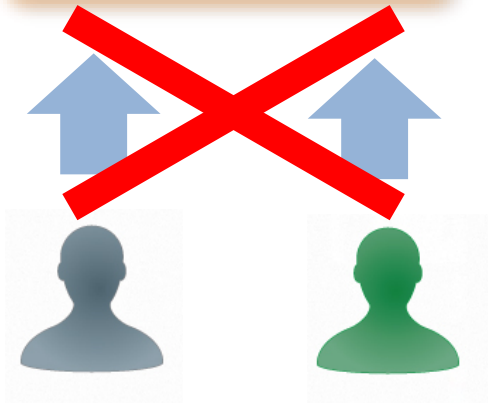
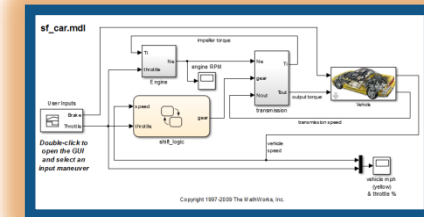
※カスタムルールの作成には、Simulink Verification and Validationの機能が必要です

コンフリクトへの対応

- ダイアログで警告してコミットを中断
 - コミットしようとしたファイルがすでに更新されていた場合
 - コミットしようとしたファイルがチェックアウト (ロック) されていた場合



ロック



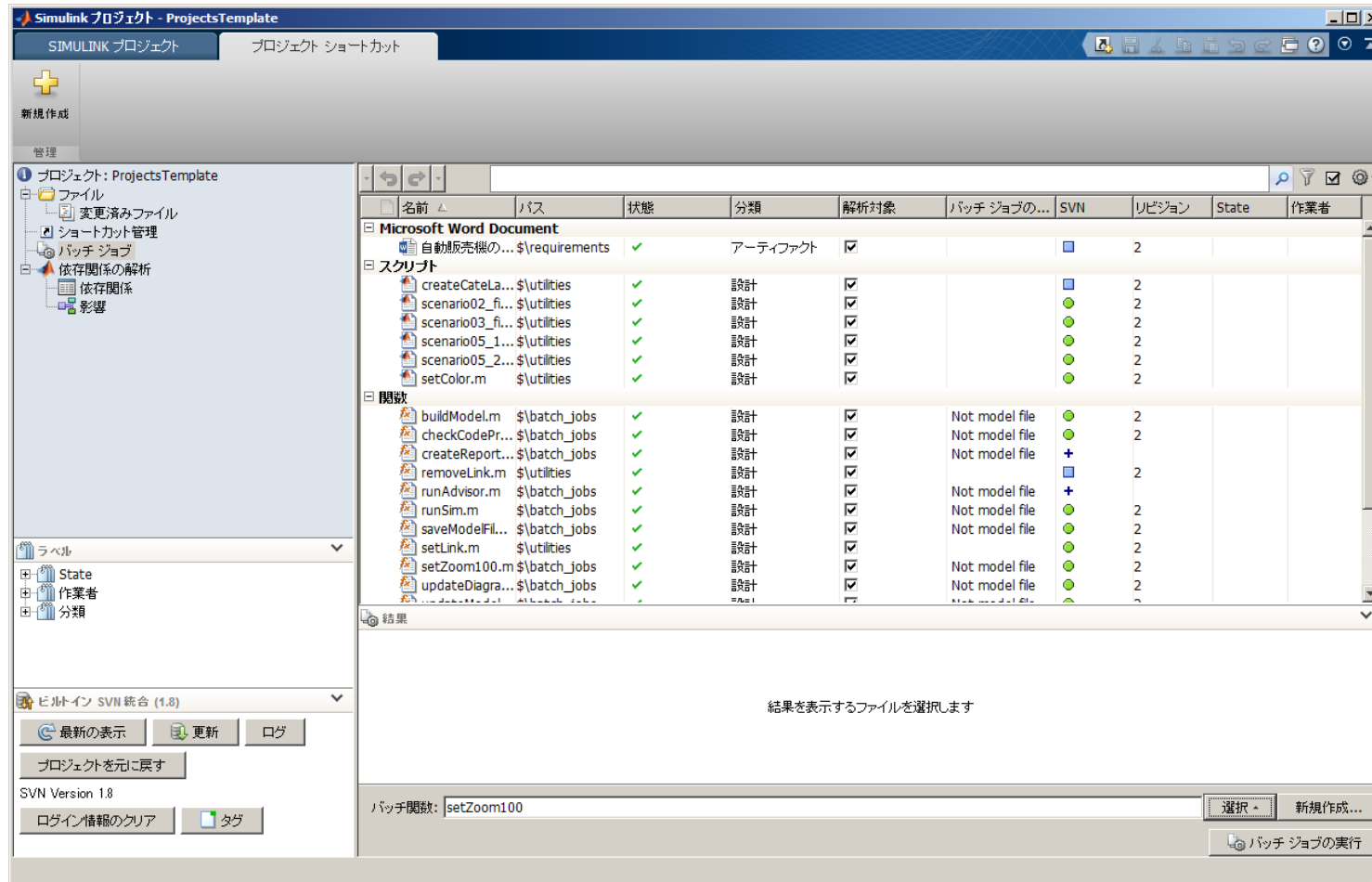
シナリオ⑤-3：並行開発

- 担当
 - 赤井TL

- タスク
 - バッチ処理
 - タグ作成
 - レポート生成

バッチ処理

プロジェクトとして登録されているファイルを対象にバッチ処理を行えます



バッチジョブの実行を押下すると、プロジェクト内のファイルを横断的に処理を行います

タグ作成

プロジェクトの1つの区切りをタグとしてレポジトリ内に登録することができます。

- タグからプロジェクトの状態を復元することができます。
- 復元には、SVN のコマンドから行う必要があります。
 - Simulink プロジェクトでは、タグを登録する機能のみにになります。

The screenshot displays the Simulink Projects environment. On the left, the 'Simulink プロジェクト - ProjectsTemplate' window shows a list of files and folders. A red box highlights the 'タグ' (Tag) button in the bottom left corner. A dialog box titled 'タグを指定' (Specify Tag) is open, showing the text 'phase' in the input field. On the right, a file explorer view shows the repository structure, including folders like 'branches', 'tags', 'phase 1', and 'trunk'. The 'tags' folder is expanded, showing the 'phase 1' tag.

名前	状態	分類	SVN	リビジョン	作業者
batch_jobs		なし		2	
data		なし		2	
models		なし		3	
CalcReturn.sx		設計	+		
Purchase.sx		設計	+		
SumCoi			+		
Vending					
report					
CalcRet					
CalcRet					
CalcRet					
Purchas					
Purchas					
Purchas					
SumCoi					
SumCoi					
VendingMachine.uocx					
VendingMachine.emf					
VendingMachine.html					
requirements		なし		2	
白黒印刷用紙の設計仕様書 docx		アーカイブ		2	

レポート生成

プロジェクト内のモデルごとにレポートを生成します

ガイドラインチェック
結果レポート

モデル アドバイザー レポート - AutoDoorLockDiag_correct.slx
モデル バージョン: 1.67

Simulink バージョン: 8.4
システム: AutoDoorLockDiag_correct
参照モデルとして扱う: off

実行の概要
パス 36 失敗 0 警告 0 実行なし 17 総数 53

概要

モデルの階層構造/複雑度

	DI	CI	MCDC
1. AutoDoorLockDiag_harness	34 100%	98%	96%
2. ... Test Unit (copied fr...			
3. ... AutoDLockPerm			
4. ... Timer			
5. ... DoorLockDiag			
6. ... DLockCondi			
7. ...	20 100%	100%	100%
8. ...	10 100%	100%	100%

カバレッジ測定結果
レポート

有効と証明されたオブジェクト

#	タイプ	モデル項目	説明
9	オーバーフロー	Controller.Sum	オーバーフロー
24	オーバーフロー	Controller/PI_Controller/Discrete-Time Integrator	オーバーフロー
28	オーバーフロー	Controller/PI_Controller/Kp	オーバーフロー
31	オーバーフロー	Controller/PI_Controller/Kp1	オーバーフロー

設計エラー検出
結果レポート

#	タイプ	モデル項目	説明
17	オーバーフロー	Controller.Sum1	オーバーフロー

Fast Accel

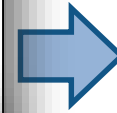
Test Result Information

Result Type: Test Case Result
Parent: Signal Builder Baseline examples
Start Time: 2015-Jan-07 09:05:18
End Time: 2015-Jan-07 09:05:19
Outcome: Failed
Cause Of Failure: Criteria evaluation resulted in failure.

Test Case Information

Name: Fast Accel
Type: Baseline Test

入力に対する出力と期待値
との比較結果レポート



レポート エクスプローラー - レポート - simulink-summary.rpt

Report Generator

- レポート - simulink-summary.rpt
 - if (isempty(find_system('blockdiagramtype','mo...))
 - 実行 - No models are open, open f14...
 - タイトル ページ - Simulink Summary Report
 - モデル ループ - 現在のモデル
 - モデルをシミュレーション - モデルの開始/終了
 - 章 - <Title from first child>
 - Model 名前
 - Model アンカー
 - システムのスナップショット
 - Model プロパティテーブル - %(Name)> Si...
 - Model プロパティテーブル - %(Name)> Su...
 - 第 1 節 - Systems
 - System 概要テーブル
 - 第 1 節 - Blocks
 - Block 概要テーブル
 - ブロックタイプのカウンタ
 - 第 1 節 - Date and Functions
 - 関数/変数表 - 現在のモデルのレポ...
 - 第 1 節 - Signals
 - Signal 概要テーブル
 - 第 1 節 - Scopes
 - スコープのスナップショット - 現在のモ...

名前

- Fixed-Point Designer
 - 固定小数点ブロックのループ
 - 固定小数点プロパティテーブル
 - 固定小数点ログ オプション
 - 固定小数点概要テーブル
- Handle Graphics
 - Figure スナップショット
 - Figure ループ
 - Graphics オブジェクト ループ
 - Handle Graphics アンカーのリンク
 - Handle Graphics パラメーター
 - Handle Graphics プロパティ...
 - Handle Graphics 名
 - Handle Graphics 概要テーブル
 - 座標軸スナップショット
 - 座標軸ループ
- MATLAB
 - MATLAB プロパティテーブル
 - MATLAB 式の評価
 - MATLAB/ツールボックスのバジ...
 - 変数の挿入
 - 変数テーブル
- Polyspace
 - Acronym Definitions
 - Code Metrics Details
 - Code Metrics Summary
 - Code Verification Summary
 - Code and Verification Informat...
 - Coding Rules Details
 - Coding Rules Summary

レポート オプション

レポート ファイルの場所
O:\Work\SLD\Work\simulink-summary.html

ディレクトリ: 現在の作業ディレクトリ
ファイル名: セットアップ ファイルと同じ mode

☐ レポートが既に存在する場合、上書きを防ぐ

レポート形式とスタイルシート
ファイル形式: web (HTML)

生成オプション
☒ 生成後にレポートを表示
☐ 生成前に自動的に保存
☐ コンパイル情報をレポートに含めるためにモデルイ
生成後にこの文字列を実行:

レポートの説明
Summary Report
This setup file creates a high-level overview r...
generate a report for the all open models (or...
The report is created by default in your preser...
have write access in this directory.

元に戻す(B) ヘルプ(H)



シナリオ⑤： 並行開発のポイント

- ラベルを使ってプロセスを確認
- バッチ処理による自動化
 - モデルアドバイザ、レポート生成、など
- ある時点のプロジェクトの保存するために、タグを作成
- コンフリクトの回避
 - チェックアウト(編集権限)
 - コミット時に警告



まとめ

チーム開発をサポートするため、Simulink プロジェクトを活用することで、プロジェクトの並行開発を手助けします。

- スクリプトを共通化して、開発の土台を共通化できます。
- プロジェクトの関連ファイルを管理することができます。
- 構成管理ツールとの連携により、ファイル履歴管理ができます。