

車載ソフトウェア開発への ZIPC TERAS 適用

三菱電機コントロールソフトウェア株式会社
三田事業所 技術第1部 技術第1課長

森下 修

1. まえがき

近年、カーナビゲーションをはじめとする車載機器は、機器同士の連携だけではなく無線通信を介した車外ネットワーク環境との連携によって付加価値を創出し、快適性や利便性を高めていこうとしている。

一方、車載ソフトウェアは流用開発や開発段階での仕様変更が多く、上流フェーズから下流フェーズへ仕様を詳細化していく過程で、要求仕様からテスト仕様までの一貫性確保を証明することの重要性が、高品質なソフトウェア開発を行う上でますます高まっている。そこで、トレーサビリティ支援ツールを導入し、効率的かつ正確に仕様の一貫性を確保することで、高品質なソフトウェア開発の実現を目指した。

2. ZIPC TERAS 導入の背景

従来は手作業でトレーサビリティマトリクスを作成し、ドキュメント間の一貫性を確認していた。しかし仕様変更が発生するたびにドキュメント間の一貫性を再確認する作業は、確認漏れによる品質悪化を発生させないためにも慎重に行う必要がある。これは工程やコスト面で問題となっていた。そのためツールを用いた自動化の検討を始め、トレーサビリティ支援ツールの情報収集を行った。その結果、客先から様々な形式のドキュメントで提示される最上位仕様からの一貫性確保を考えた場合、対応形式としてPDFが標準サポートされている、タグ情報の埋め込みを自動で行うことができるキャッツ株式会社のZIPC TERAS^{注1}の採用に至った。

3. ZIPC TERAS 適用範囲

今回、C言語で約8Klineの機種をZIPC TERAS適用対象とした（以下、対象プロジェク

トと記す）。また図1にV字モデルにおける設計詳細化から試験の流れの中でZIPC TERAS適用範囲を示す。手作業での一貫性確認粒度と同じく、各フェーズの成果物間の「章・節・項」単位でトレースを確認することを目標としたが、新たな取り組みとして、開発着手段階で客先との仕様整合がとれていなかった機能は、文章レベルでトレースを確認することにした。これは暫定仕様で開発をスタートしておき、開発途中で仕様確定した場合に、暫定仕様からの変更点を確実にトレースする効果を狙ったからである。

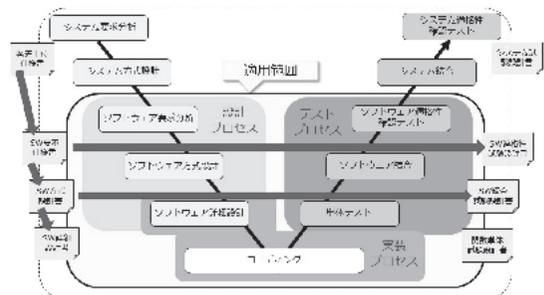


図1 開発プロセス上のZIPC TERAS適用範囲

また対象プロジェクトでは、ソフトウェア詳細設計書にソースコードと1対1レベルのフローチャートを記述している。このためソースコードはトレース対象から除外した。さらにソースコードからツールを用いて関数単体テスト用のテストパターンを自動生成しているため、関数単体テスト設計書もトレース対象から除外した。

4. 当社の導入事例

図2に対象プロジェクトでのZIPC TERAS運用フローを示す。

まず、プロジェクト開始時点でZIPC TERASに対して必要な初期設定を行う。

次にトレース対象のドキュメントを登録し、ト

レーサビリティ作業を行う。

開発途中に発生した仕様変更やテストで検出した不具合の対応に応じてドキュメントを修正し、トレーサビリティの網羅率や影響範囲を確認する。

4.1 初期設定

図3にプロジェクトで初めてZIPC TERASを使用する場合の初期設定手順を示す。

まず、プロジェクト共通ルールの設定が必要である（表1）。

次にプロジェクト毎のアカウントやWorkspace、リポジトリドメインを作成する。

また、ZIPC TERASをドキュメント体系に合致したフォルダ構成で運用するために、デフォルトのフォルダ構成からプロジェクトのドキュメント体系に合った形へフォルダ構成を変更する（図4）。これを「TRAモデルを設定する」という。

表1 プロジェクト共通設定

内容	説明
アカウント名	プロジェクト毎に一意なアカウント名を設定する。
TRAモデル設定	プロジェクト毎にTRAモデルを設定する。
ルールファイル	プロジェクト毎にTRAモデルに適用するルールファイルを設定する。
ドキュメントのTRA要素化の方法	どのようなTRA要素化するか（手動、自動）、どのレベルでTRA要素化するか（ファイルレベル、章・節・項レベル）を設定する。
検証結果の出力先	ZIPC TERAS上で実行するテストの結果をどのファイルで出力するかを設定する。
ZIPC TERASのインストール	ZIPC TERASのインストール先を設定する。
ZIPC TERASのバージョン	ZIPC TERASのバージョンを設定する。
ZIPC TERASのパス	ZIPC TERASのパスを設定する。

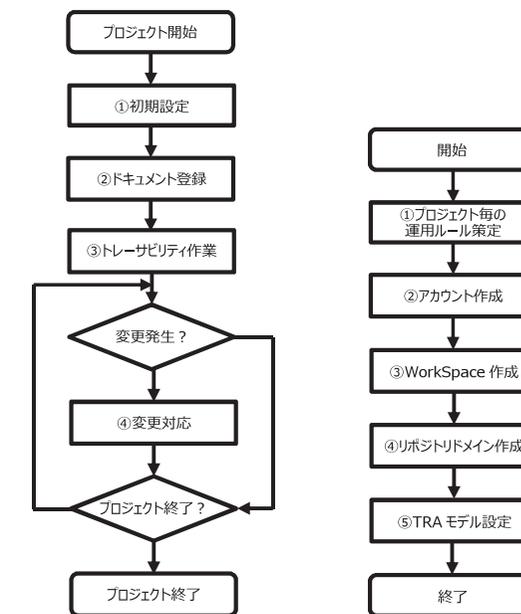


図2 ZIPC TERAS 運用全体フロー

図3 初期設定手順

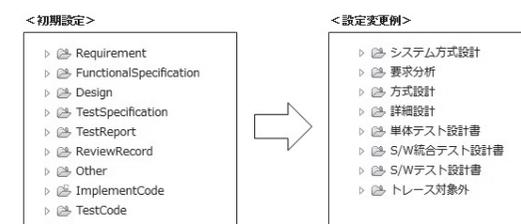


図4 TRAモデル

4.2 ドキュメント登録

図5にZIPC TERAS上にドキュメントを登録する手順を示す。ZIPC TERASではトレース対象として識別するタグ情報の単位を「TRA要素」といい、TRA要素を埋め込むことを「TRA要素化する」という。

Microsoft Office Word 2010^{注2}の見出しマップ機能を用いてドキュメントを作成していれば、ツールが自動で章・節・項をTRA要素化する。

TRA要素化は以下の3通りの方法から選択可能で、複数を組み合わせても良い。対象プロジェクトでは章・節・項単位に加え、文章単位でのTRA要素化も必要だったため(1)と(2)を採用した。

(1)ZIPC TERASのデフォルトのルールファイルを使用し、自動でTRA要素化する。

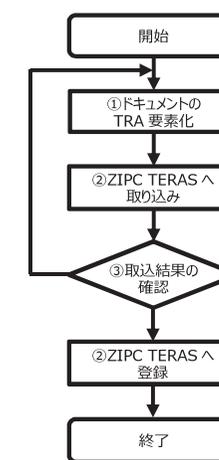


図5 ドキュメント登録のフロー

(2)ドキュメントに合う形でルールファイルを作成し、自動でTRA要素化する。

(3)ルールファイルは使用せず、手動でTRA要素化する箇所を選択しTRA要素化する。

デフォルトのルールファイルの内容を表2に示す。

TRA要素化したドキュメントはTRAエクスプローラ上で、登録したいTRAモデルのフォルダへのドラッグ&ドロップ操作で取り込み可能で

ある。取り込んだドキュメントが意図したとおりにTRA要素化されているかをTRAモデル取り込みウィンドウ上で確認し(図6の枠部分)、問題がなければドキュメント登録は完了である。

表2 デフォルトルール一覧

内容	説明
Word	見出しごとにTRA要素化される。
Excel	シートごとにTRA要素化される。
PDF(しおり付き)	しよりの項目ごとにTRA要素化される。

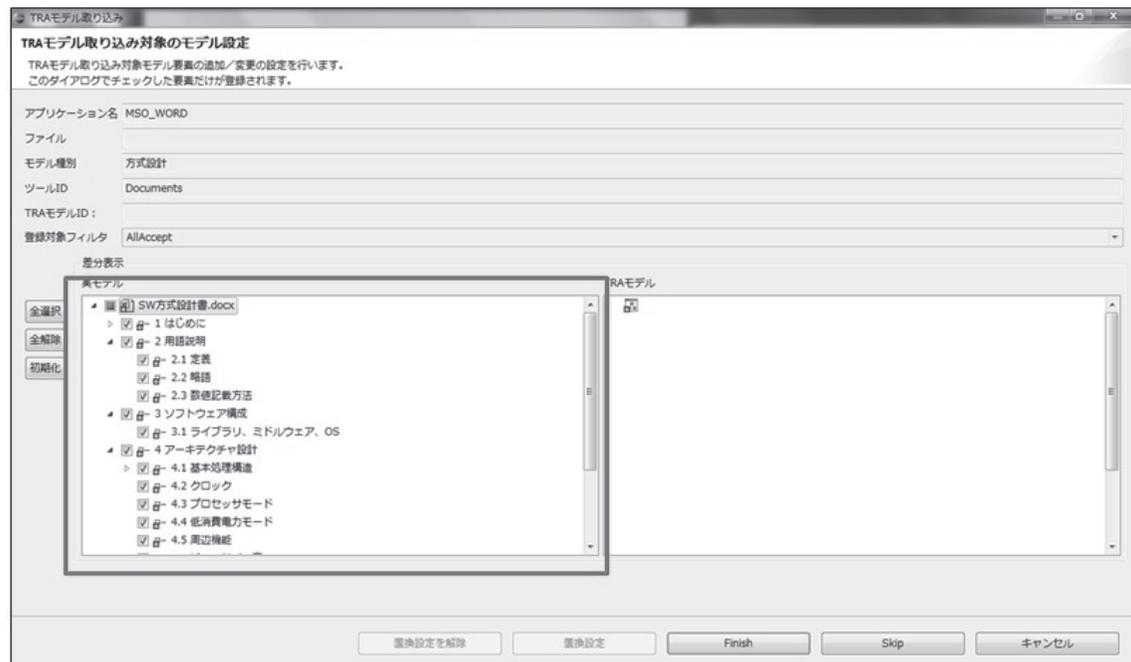


図6 TRAモデル取り込みウィンドウ

4.3 トレーサビリティ作業

図7にトレーサビリティ作業手順を示す。

Microsoft Office Word 2010の見出しマップから自動的にTRA要素化されているものの、各要素のリンク付けは手作業で実施する必要がある。

これを「TRA要素のリンク付け」という。

ここで重要なのは、TRA要素間にリンク漏れがないかを丁寧に確認しながら作業することである。

ZIPC TERASではTRAエクスプローラ上で対象ドキュメントを右クリックし「カバレッジ確認」を選択するとカバレッジ確認ビューが表示され、上流/下流に対してのリンク漏れを検出することができる(図8)。

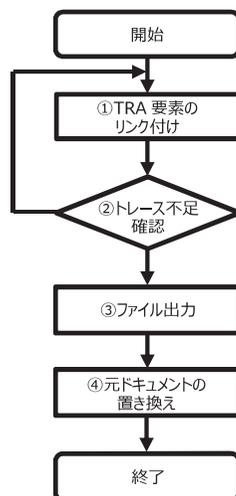


図7 トレーサビリティ作業手順

カバレッジ確認ビューでは、同一 TRA モデルの文書に対して1つでもリンク付けが行われていれば「リンク付けできている」と判断される。例えば下位文書の複数項目とリンクを結ばなければならない箇所に対してリンク付けの漏れがあったとしても認識はできない(図9)。



図8 カバレッジ確認ビュー



図9 リンク漏れの判定

最終的な歯止めとして現状はレビューワがレビュー段階で細部まで確認を行っているが、ドキュメントの章構成を見直すことで回避可能か否かは、今後の検討課題と考える。

またドキュメントの章・節・項には「はじめに」「文書の目的」など、設計情報が記載されない箇所も存在する。これらは上位文書からのリンク付け対象外だが、ZIPC TERAS上ではカバレッジが100%とならずリンク付けNGと判断されてしまう。これを回避するためには、リンクエディタ上でトレース対象外の要素を「例外」と設定することでカバレッジ対象外とすることができる。

リンク付けを行った結果は、ファイル出力することでレビュー用のエビデンスとして使用できる。出力形式は「標準」「トレーサビリティマトリクス」「対応表」の3通りあるが、対象プロジェクトでは過去に手作業で作成していたトレーサビリティマトリクスと構成が似ているという理由から「対応表」を選択した。しかし例外設定によってトレース対象外とした章・節・項は、対応表上では空欄で表示されてしまい、レビューワには「トレース漏れ」と判断されかねない。そこで対象プロジェクトでは図10のように、トレース対象外だけをまとめた「トレース対象外.doc」というドキュメントを追加することで、レビューワにトレース対象外であることが明確に伝わるようにした。

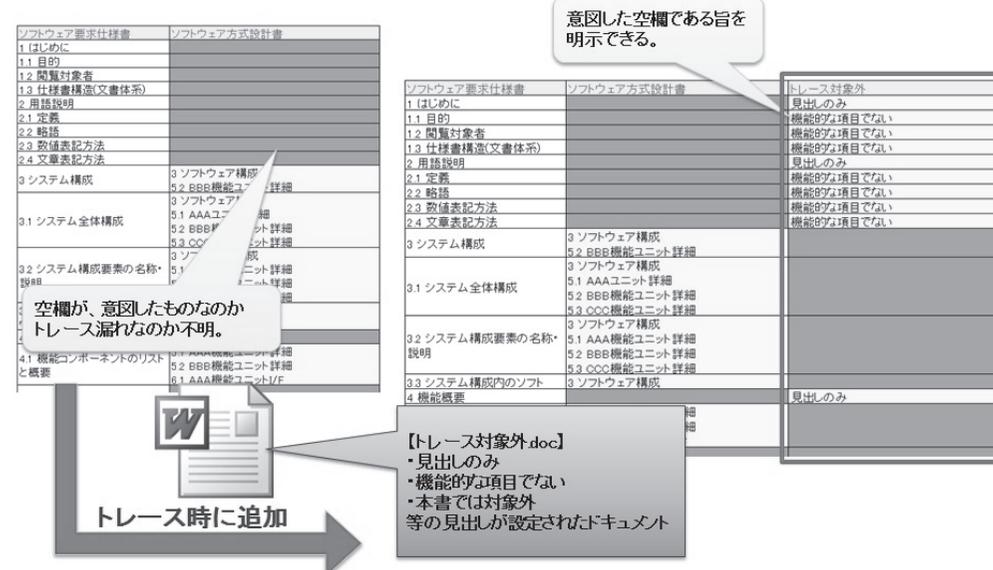


図10 リンク対象外ドキュメント

