

鉄道車両用空調制御装置のソフトウェア開発におけるプロセス改善

長崎事業所 技術第2部 ファームウェアシステム課
峯本 秀樹、太田 雄幸

1. まえがき

鉄道車両用空調装置(図1)は、三菱電機(株)長崎製作所の主要製品である。鉄道車両用空調装置(以下、空調装置)は、空調制御装置と装置に組み込まれた制御ソフトウェア(以下、空調制御ソフトウェア)で制御している。

近年、空調装置は市場が拡大し、快適性の追求により、きめ細かい制御が要求されるようになってきた。

また、鉄道車両には、通勤電車、特急電車、新幹線や豪華列車など様々な種類があり、それらの車両特性に合わせた制御が要求されてきた。

このため、空調制御ソフトウェアは種類が多く、仕様が複雑化している。



図1. 鉄道車両と空調装置の外観

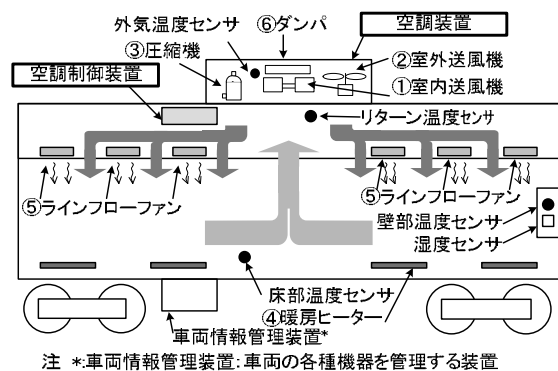
当所は、空調制御装置のソフトウェア開発を担当している。空調制御ソフトウェアの開発では、ベースとなる標準ソフトウェアがあり、種類ごとに仕様変更や追加を行う派生開発を行っている。

本稿では、2章で空調装置についての説明を行い、3章～6章で空調制御ソフトウェア開発プロセスの課題と対策を述べる。

2. 車両空調システムの概要

2.1 空調装置

通勤電車を例に、車両内機器の構成を図2に、冷凍サイクルの概念を図3に示す。



注 *車両情報管理装置:車両の各種機器を管理する装置

図2. 車両内機器の構成

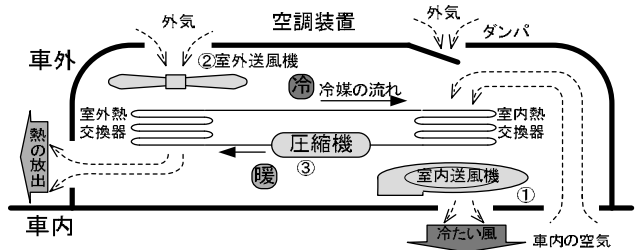


図3. 冷凍サイクルの概念図

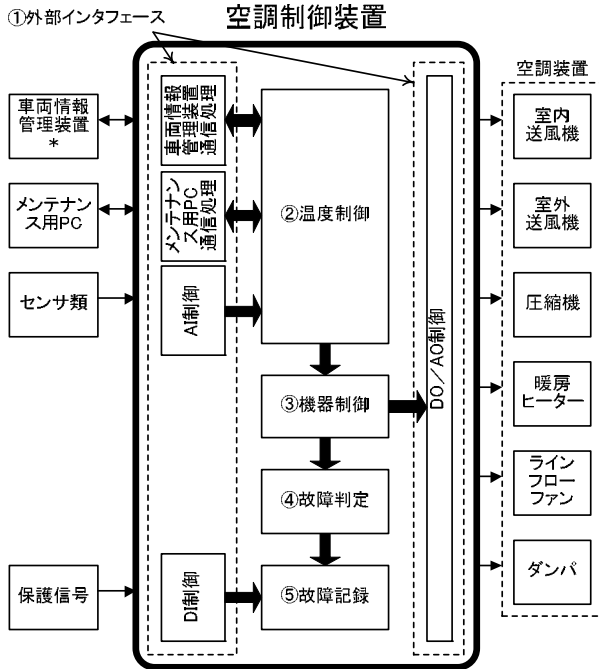
空調装置は、室内送風機(図2①、図3①)、室外送風機(図2②、図3②)、圧縮機(図2③、図3③)により構成される。その他には、車両に設置される暖房ヒーター(図2④)、ラインフローファン(図2⑤)、ダンパ(図2⑥、図3⑥)がある。

暖房ヒーターは、車両の床や座席下に設置される。ラインフローファンは、天井内に設置される線状の扇風機で、空気のかくはんや涼感を与える役目がある。ダンパは、外気の取り込み量を調節する。

空調装置では、冷凍サイクルの原理により、冷房を行っている。圧縮機により圧縮した冷媒ガスを循環し、室外送風機により車内の熱を車外に排出し、室内送風機により車内に冷風を吹き出し、冷気を発生させている。

2.2 空調制御ソフトウェア

空調制御ソフトウェアの構成を、図4に示す。



注 *:空調装置を含む各種車載機器を一括して管理する装置

図4. 空調制御ソフトウェア構成図

空調制御ソフトウェアは、①外部インタフェース、②温度制御、③機器制御、④故障判定、⑤故障記録により構成される。

以下に空調制御の流れを説明する。

- (1)外部インタフェースからの入力情報により冷房か暖房か(空調運転モード)を決定する。
- (2)温度制御で車内温度と目標温度のかい離により冷暖房の強さ(運転能力)を決定する。
- (3)機器制御で空調運転モード及び運転能力に基づき、室内送風機や暖房ヒーターといった空調装置を構成する機器を制御する。
- (4)空調装置の各機器の状態監視を行い、故障と判定した時は、対象機器を停止する。
- (5)故障記録で故障発生時のトラブルシューティングや空調制御状態の把握を目的としたデータの記録を行う。

空調制御装置の外観を図5に示す。空調制御ソフトウェアは、CPU基板に実装され、入出力基板、通信基板の制御を行っている。

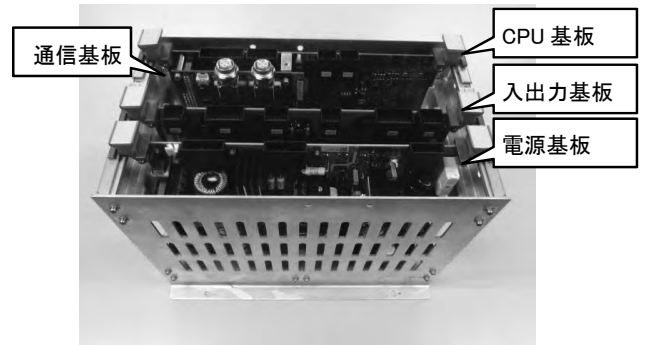


図5. 空調制御装置の外観

3. ソフトウェア開発プロセスにおける課題

空調制御ソフトウェアの開発は、図6で示すように設計部門、製作部門、品管部門に分かれている。当所は製作部門を担当しており、設計部門からシステム仕様書を入力し、それを基にソフトウェアを製作している。

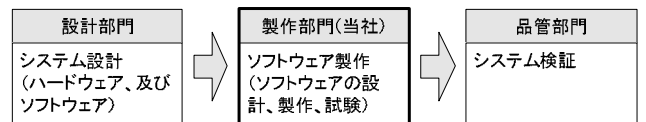


図6. 空調制御装置の開発体制

このソフトウェア製作において、以下の問題が顕在化してきた。

- (1)システム仕様書の記載内容に粒度の差
- (2)変更による影響範囲の漏れ
- (3)開発期間の不足

これらの問題により、手戻りが増加し、その改善が急務となった。

このため、以下の課題を解決する必要があった。

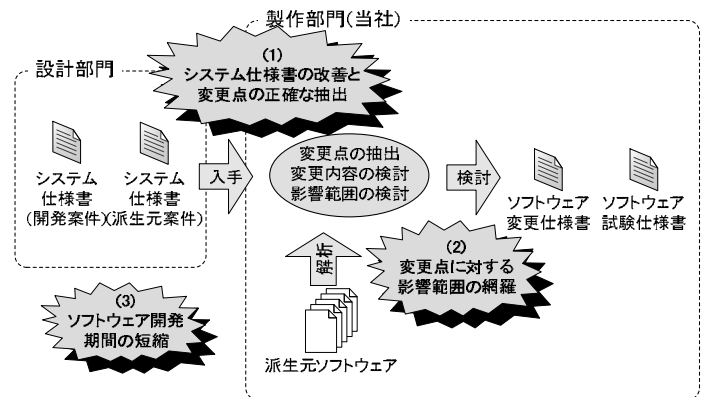


図7. 従来のソフトウェア製作の課題

(1) システム仕様書の改善と変更点の正確な抽出

製作部門は、設計部門から入手したシステム仕様書と、派生元のシステム仕様書を基に、変更点を抽出する。設計部門と製作部門、共に担当者を増員したことで、システム仕様書の記載内容に粒度の差が現れ、仕様の要求意図を読み違える問題が生じ、製作部門で正確に変更点を抽出できず漏れが発生していた。これを防ぐために、システム仕様書の記載内容の改善と変更点の正確な抽出が課題である。

(2) 変更点に対する影響範囲の網羅

仕様が複雑化しており、機能間の関係が複雑になり、1つの変更点に対するソフトウェアの変更箇所は複数箇所に及ぶようになった。また、変更や追加を繰り返したことで、ソフトウェア構造が複雑化し、熟練者以外では変更によって影響を受ける機能を正確に抽出できなくなってきた。このため、変更前に動作していた機能が、変更後は動作しなくなる場合が発生するようになった。これを防ぐために、影響範囲を正確に網羅することが課題である。

(3) ソフトウェア開発期間の短縮

ソフトウェアの製作期間は、過去に比べて短くなってきており、受注規模や案件の増加により、複数の案件が輻輳することがある。このため、ソフトウェア製作において、手戻りによるロスを削減し、短い工程で完了することが課題である。

4. ソフトウェア開発プロセスの改善方針

今回、開発プロセスに派生開発の要求に合ったXDDP (eXtreme Derivative Development Process)を導入し、開発プロセス及びその成果物を見直すことで、3章で挙げた課題の解決を図った。

4.1 XDDPの特徴

XDDPとは、派生開発に特化した開発アプローチであり、短納期及び全体を理解して取り掛かることが困難といった派生開発特有の問題に対して、合理的に対応する方法である。

XDDPでは、変更漏れや変更誤りを防止するため、以下の3つの成果物を作成する。

- (1) 変更要求仕様書
- (2) トレーサビリティマトリクス
- (3) 変更設計書

システム仕様書からXDDPの成果物への展開イメージを図8に示す。また、XDDPの成果物及び記述内容を表1に示す。

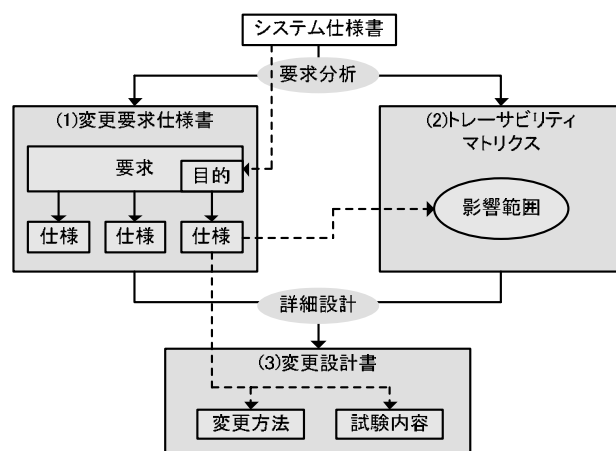


図8. システム仕様書からXDDPの成果物への展開

表1. XDDPの成果物の記述内容

No.	成果物	記述内容
1	変更要求仕様書	<ul style="list-style-type: none"> ・何を変更するか ・なぜ変更するか ・どういう仕様をどのような仕様に変更するか ・何を変更すればよいと考えているか
2	トレーサビリティマトリクス	<ul style="list-style-type: none"> ・どこを変更するか ・他に変更する箇所がないか
3	変更設計書	<ul style="list-style-type: none"> ・具体的にどのように変更するのか ・どのように変更すれば目的を達成できると考えているか ・どうやって確認するか

「変更要求仕様書」(図8(1)、表1 No.1)は、客先要求(「要求」と「何のために変更するのか」「何を期待しているのか」(「目的」)を明確にし、「要求」を実現する視点でソフトウェアをどのように変更するか(「仕様」)を導き出す。また、「要求」と「仕様」を階層関係で表す。これにより、客先要求からソフトウェアへの展開イメージが明確になり、変更漏れや変更誤りを防止する効果がある。

「トレーサビリティマトリクス」(図8(2)、表1 No.2)は、「仕様」に該当する箇所や変更箇所の関連性から影響範囲を抽出する。さらに、「仕様」と後述する「変更設計書」を関連付けることにより、影響箇所(他の変更箇所)の抜け、思い込みによる無関係な箇所の変更、「要求」の解釈誤りを防止する効果がある。

「変更設計書」(図8(3)、表1 No.3)は、具体的な変更方法を記述する。また、変更に伴う試験内容も記述する。

4.2 XDDPの成果物導入方針

XDDPの3つの成果物について、以下の方針で本ソフトウェア開発に適用した。

「変更要求仕様書」と「トレーサビリティマトリクス」を同一様式で表現するとさらに効果的であるため、「変更点管理マトリクス表」として作成し、適用を行った。「変更要求仕様書」部は、「要求」と「仕様」を階層関係で表し、「要求」の目的、または理由を記述する「目的」欄及びその補足説明を記述する「説明」欄がある。「トレーサビリティマトリクス」部は、機能を定義し、「仕様」がどの機能に影響するかを明示する。

変更点管理マトリクス表の適用イメージを図9に示す。

変更要求仕様書			トレーサビリティマトリクス			
要求	仕様	内容	機能1	機能2	機能3	機能4
1	要求	要求内容を記述				
	目的	本要求の目的・理由を記述				
	説明	上記「目的」の補足説明を記述				
	仕様A	変更内容を記述	●			
	仕様B	変更内容を記述	●			
2	要求	要求内容を記述				
	仕様C	変更内容を記述				●

図9. 変更点管理マトリクス表への適用イメージ

また、「変更設計書」については、既に運用していたソフトウェア変更仕様書、及びソフトウェア試験仕様書が表1に示す記述内容に対応していたため、そのまま適用した。

XDDPの3つの成果物と空調制御ソフトウェア開発の様式の対応について、表2に示す。

表2. XDDPの成果物と様式の対応

No.	XDDPの成果物	空調制御ソフトウェア開発での様式
1	変更要求仕様書	変更点管理マトリクス表
2	トレーサビリティマトリクス	変更点管理マトリクス表
3	変更設計書	ソフトウェア変更仕様書 ソフトウェア試験仕様書

5. ソフトウェア開発プロセスの改善

5.1 開発プロセスへのXDDP導入

変更点管理マトリクス表及び設計部門で新たに作成されるようになった「変更点一覧」を適用し、従来の開発プロセスを以下のようにした。改善後の開発プロセスの比較を図10に示す。

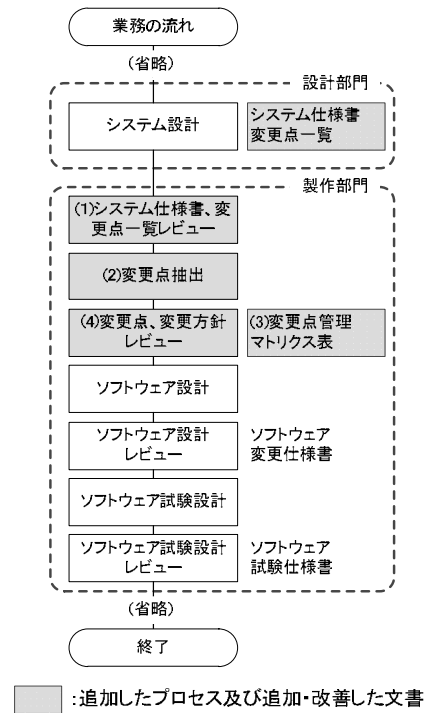


図10. 改善後の開発プロセスの比較

(1) システム仕様書、変更点一覧レビュー

変更点のチェックプロセスを新たに設けた。このプロセスでは、開発案件と派生元案件のシステム仕様書に加え、変更点一覧を入手する。入手した資料のレビューを実施し、ソフトウェア開発に着手する。変更点一覧には、開発案件と派生元案件の差異の他、変更理由及び変更内容が記載されている。

(2) 変更点抽出

このプロセスでは、開発案件と派生元案件のシステム仕様書を参照し、変更点一覧をもとに変更点を抽出する。また、抽出した変更点に対し、ソフトウェアの変更箇所及び変更内容を検討する。さらに、変更に対する影響箇所も確認する。

(3) 変更点管理マトリクス表

(2)の検討結果を変更点管理マトリクス表に反映する。

(4) 変更点、変更方針レビュー

変更点及び変更方針の妥当性をレビューする。

製作部門での設計作業開始後、システム仕様書が変更された場合は、随時、上記(1)～(4)を繰り返す。

また、システム仕様書において、章構成の統一及び図や表現方法の改善など記載方法の改善も行われた。

5.2 変更点管理マトリクス表の改善

開発プロセスの見直しに併せて、図9で示した変更点管理マトリクス表に対して、以下の改善を実施した。変更点管理マトリクス表への展開イメージを図11に示す。

【変更点一覧】

派生元	開発案件	変更前(派生元)の内容	変更後(開発案件)の内容
3	I/F		
1	I/O I/F		
1	AI	AS01:FATS...	AS01:SATS...
2	AO	BS01:FAD11...	BS01:FAD1...

「変更点」 「要求」を「上位要求」と「下位要求」の2階層に分割 「要求」

【変更点管理マトリクス表】

変更点一覧 No.	項目	上位要求	下位要求	仕様番号	内容	変更有無	記載箇所	TM*	
								AI	AO
3	3.1.1. AI	1	要求		アナログ入力今回仕様の通り変更する。 ハードウェアの機器構成が変更となり、センサ数の増減へ対応するため。 ハードウェアの機器構成が変更となるので、センサ数が増減ソフトウェアも変更となる。 以下の通り、変更する。 <変更前> AS01:FATS... <変更後> AS01:SATS...				
			1	要求	アナログ抵抗入力CHを変更する。 車内、外気温度を検出する温度センサの接続変更へ対応するため。 アナログ抵抗入力CHIに応じて... 以下の通り、...				
				仕様A			変更仕様書	3.1.1	4.1.1
							試験仕様書	3.1	4.1

注*:トレーサビリティマトリクス

図11. 変更点管理マトリクス表への展開イメージ

- (1) 変更点一覧に記載された項目を「変更点」とし、変更点管理マトリクス表へ追加した「変更点一覧」欄へ展開する。また、変更点一覧に記載されている「変更前(派生元)の内容」と「変更後(開発案件)の内容」の差分を「要求」として変更点管理マトリクス表へ記載し、「変更点」と紐付ける。さらに、「要求」に対するソフトウェアの変更内容(「仕様」)を記載する。これにより、関連性が明確になり、変更点の抽出漏れをチェックできる。
- (2) 設計部門が提示する変更点は、システム全体を対象としているため、必ずしもソフトウェアに変更が入るとは限らない。「変更有無」欄を追加し、ソフトウェアの変更有(○)無(×)を記載することで、製作部門や品管部門での試験の要否を明示できる。
- (3) トレーサビリティマトリクス欄には、機能を定義し、ソフトウェアの変更で影響がある機能に「●」を記入する。また、ソフトウェア変更仕様書及びソフトウェア試験仕様書内の項番を併記し、対応を明確にすることで、変更漏れや試験漏れをチェックできる。

- (4) 「要求」の範囲が広い場合、「要求」の範囲を狭めることで、より詳細な検討ができる。今回、「上位要求」と「下位要求」の2階層の構成とすることで、「要求」の漏れを無くすようにした。

6. 開発プロセスへのXDDPの導入による効果

開発プロセスにXDDPを取り入れ、変更点一覧及び変更点管理マトリクス表を追加することで、以下に示す効果を確認できた。

- (1) システム仕様書に記載された仕様の意図が明確に理解できるようになり、設計部門及び製作部門による変更点の相互チェックも可能となることで、変更点の抽出漏れが減少した。
- (2) 変更理由を明確に記述し、それを理解した上でソフトウェア製作を行うため、変更内容と影響範囲の誤りが減少した。
- (3) 開発プロセスに「変更点抽出」とそのレビューを追加し、妥当性をレビュー後、後工程へ移行するようにしたこと、問題の早期発見が可能となった。これにより、手戻りするロスが減り、工期短縮が図れた。

7. むすび

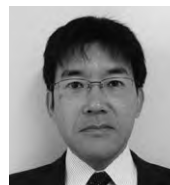
派生開発の要求に合ったXDDPを取り入れ、開発プロセスの見直しを行うことで、派生開発で抱えていた問題を解決することができた。変更点管理マトリクス表、ソフトウェア変更仕様書及びソフトウェア試験仕様書内に試験結果を記入したソフトウェア試験成績書を他部門とのレビューの審議資料としており、引き続き運用していく所存である。

最後に、ソフトウェア開発プロセス改善にあたりご協力いただいた関係者の方々に深く感謝申し上げる。

執筆者紹介



峯本 秀樹 ミネモト ヒデキ
1992年入社。主に空調制御システムのソフトウェア開発に従事。現在、長崎事業所技術第2部ファームウェアシステム課。



太田 雄幸 オオタ タケユキ
1997年入社。主に空調制御システムのソフトウェア開発に従事。現在、長崎事業所技術第2部ファームウェアシステム課。