

設計・開発における 品質保証・ミス防止活動のあり方

中央大学理工学部 中條武志

質問1 商品は一つ一つ異なり、全く同じ商品を何度も設計・開発することはありません。同じものを何度も繰り返し作る生産とは本質的に異なると思います。生産を効率化するためのマネジメント手法は、商品の設計・開発にそのまま使えるのでしょうか。

全く同じ商品を二度設計・開発することはありません。その意味では同じ物を繰り返し作る生産のマネジメントと設計・開発のマネジメントは本質的に異なります。他方、仕事の質（Quality）を向上させるためには、何らかの繰り返し・類似性があることが必要です。このような視点から、一つの組織が開発していた複数の商品およびそれらの開発過程を比較して見ると、そこには様々な相違点があると同時に、多くの共通点があることが分かります。

第一は、設計そのものの共通性です。同じ系列の商品であれば、新商品だからといってすべてを設計し直している場合は少ないと思います。既存の設計を流用している部分もかなりあるのが普通です。例えば、原子力発電プラントの設計においては、過去の設計を踏襲するのが原則となっており、既存の設計の成熟度をどれだけ上げるかが設計の質の向上の重要なポイントになっています。

第二は、設計・開発を行う際に使用している技術の共通性です。例えば、建物の設計では、建物そのものは異なっても、強度計算の技術は同じものを使います。従来の設計・開発を通して得た技術を新商品の開発で再利用することにより設計・開発の質の向上をはかることができます。

第三は、設計・開発を進める際の組織としてのしくみ、すなわち設計・開発プロセスの共通性です。担当している設計者、開発者は異なっても、どのようなルールで担当者を決めるか、開発計画をどのような様式を用いて文書化するか、できあがった設計の評価・見直しを誰がどのように行うか等については一定のやり方で進めることが可能です。一定の枠を設けることで、必要な活動が抜け落ちたり、設計・開発の質から見て重要でない活動に不要な時間をかけたりすることを防止できます。

設計・開発マネジメントの難しさは、上で述べたような共通性が、個々の商品に固有な問題の中に埋没してしまっているところにあります。設計・開発のねらいあくまでも各々の商品に固有の部分にありますが、これだけに目をうばわれているは仕事の質の向上は望めません。他方、共通性を強調する余り、設計・開発が挑戦的でない、創造的でないものになっては意味がありません。両者をうまくバランスさせること、共通部分と固有部分を明確に区別し、各々を適切に管理することが必要です。

質問2 商品の設計・開発において、品質を保証する上で注意すべきポイントは何か。

(1) 標準化・再利用の促進

設計・開発する中身は商品ごとに異なりますが、そこで使用する技術は大きく変わらないのが普通です。商品に関する様々な技術およびその活用方法を「設計標準」として整理し、これらに基づいて設計・開発が行われるようにすることが重要です。また、新商品といえども過去の設計をそのまま流用する部分も少なくありません。このような技術・設計の再利用を促進するとともに、新しい技術を活用した部分、新しい設計を採用した部分をそうでない部分とに区分し、前者について周知を集めた集中的な検討を行い、後者については既存のものを確実に踏襲することによって設計・開発の質を飛躍的に向上させることができます。さらに、設計・開発は多くの人たちの共同作業で行われますが、これが組織的に整然と実施されるためには、その手続きが定められていなければなりません。品質目標をどのようにして決定するのか、設計業務の進捗、設計内容の管理は誰がどのようにするか、設計が終わった段階でその設計審査をどうするかなど、一連の仕事の進め方について定めておくことが必要です。有効な品質保証のためには、設計・開発の仕組みに関する標準化・再利用が不可欠です。

(2) 階層化とインターフェイスマネジメント

よほど単純な商品でない限り、一度にすべての部分の設計・開発を詳細なレベルで行おうとするのは人間の能力から見ても無理があります。扱える程度の詳細さで全体を検討する（基本設計）、その中の一部を取り出してより詳細な検討を行う（詳細設計）という「階層的アプローチ」が不可欠です。この場合、余り階層を細かく分けすぎると、階層と階層の区切り目が曖昧となり、各々の階層において検討すべき事項が不明確となります。逆に、階層を荒く分けすぎると、一つの階層において検討すべき事項の数が多くなり、誤りを犯しやすくなります。意味の明確な適切な数の階層を設定することが、質の高い設計・開発を行う第一のポイントです。なお、階層化のアプローチが適切に機能するためには、単に階層を考えるだけでは不十分であり、各々の階層における設計・開発の内容を表現するための手段が不可欠です。設計・開発には多くの設計者・技術者が関係しますが、これらの人々の間のインターフェイスの管理の不適切さが設計・開発におけるトラブルの大部分を占めています。他の設計者、技術者に影響を与える情報、その中でも不具合の発生につながりやすい情報は何かを明確にした上で、これらを相手の使いやすい形で提供するためのツールの開発・活用も、設計・開発の質を向上させるポイントとなります。

(3) デザインレビューの活用

設計・開発に起因するトラブルを調べていくと、その多くは目標をかかげて挑戦している部分ではなく、思いもかけないところでの見逃しや思い込みによるものが多いことが分かります。問題があることが分かれば設計者、技術者にとってその解決策を見つけることに注力するのはそれほど難しくありませんが、問題自体を見逃していることが多いのが実態です。トラブルの危険性がないかどうか探す行為はある意味では後ろ向きの活動であり、目標の達成をめざした前向きの活動をいっしょうけんめいやれば自然とカバーできる類のものではありません。トラブル予測という特別の活動を別に計画することが必要です。デザインレビュー（DR）のねらいの一つはまさにここにあると言えます。ただし、トラブル予測の場を作ったからといってすぐにうまくいくわけではありません。FMEAやPDPCなど、可

能性のあるトラブルを系統的に列挙し、検討するためのツールが必要です。

(4) 変更管理の徹底

商品の設計・開発においては、どんなに緻密に行っても、開発が進むにつれて様々な問題が顕在化してくるのが普通です。これは新しい商品、技術に挑戦している以上避けられないことと言えます。しかし、これらの問題を放置し、各々の部門・担当者の改善努力に任せていたのでは、問題の解決がおくれ、他社との競争において不利な立場にたたされてしまいます。また、設計者や技術者のエネルギーを次の新商品へ向けることが難しくなります。顕在化してくる問題に対処するための、人員の確保・組織の設置を含め、そのための準備をしておくことが第一に大切です。次に、どんなに特別の体制を取っていても、発生したすべての問題について必要な処置を同時に取ることは困難です。目的と優先順位を明確にした活動を行うことが必要です。最後に、問題を解決するために必要な変更を正しく管理することも不可欠です。

質問3 設計・開発プロセスの標準化・再利用を進める上での注意点を教えてください。

設計・開発プロセスの標準化を進める場合、どのような節目(ステップとステップの区切り目)を設けるか、各々の節目においてステップ間で受け渡される情報をどのように可視化するかが一つのポイントとなります。設計・開発は生産と異なり試行錯誤の部分が少なくありません。その意味では、手順を細かく規定するのは現実的ではありません。反面、設計・開発の活動を一まとめに捉えていたのでは、どこまで進んでいるのか曖昧となります。また、先行・後続の順序関係が不明確となり、先に行った仕事が後に行った仕事の関係で無駄になる場合もでてきます。結局、最後の方になって多くの問題が一度に顕在化し、開発の効率を著しく阻害することになります。いくつかの節目を設けて、開発の進捗を確認するとともに、その時点において決定すべきことを決定する、検出可能な問題を組織的に明らかにし、必要な処置を早期に計画・実施することが大切です。

各々の節目が有効に機能するためには、そこまでに行われた設計・開発の結果が目で見える形になっていること、すなわち「可視化」されていること、「確認すべき項目・基準」が明確になっていることが必要です。設計・開発プロセスを表すのに表1のような一覧表が用いられる理由はここにあります。各々のステップの内容を詳細に規定するのではなく、そのアウトプットとその確認方法を明確にすることで、各々のステップのやり方に自由度をもたせた上で、開発の進捗および問題の早期解決を確実にすることが可能となります。

表1 設計・開発プロセス

ステップ	インプット	活動内容	担当	アウトプット	確認方法	関連規定

設計・開発プロセスの標準化・再利用を考える場合のもう一つのポイントは、各々の商品に固有の部分と複数の商品間に共通する部分をどのように区別し、管理するかです。各々の商品の開発は他の商品とは異なる固有のねらいをもった活動です。

その意味では、当該の商品に関する個別の目標をいかに明確にし、これらの目標の達成に特化した活動を効果的・効率的に展開・実施するかが重要となります。プロジェクト的な組織体制によって設計・開発を進める企業が多いのは当然のことと言えるでしょう。しかし、プロジェクトの成功が、個別目標の設定・展開に関わる部分だけでなく、それ以外の、言い替えればその設計・開発においては余り関心が払われていない部分での問題によって脅かされる場合が多いのも事実です。従来の商品と余り変わらない部分については、過去の経験を最大限に活用し、同じ誤りを何度も繰り返し起こさないようにすることが必要です。このためには、個々プロジェクトの結果を、設計・開発プロセスに関する標準に照らして解析し、得られた知見をもとに、個別の商品ごとに行われるプロジェクトにおいて共通的な取り組みが行われるようにすることが大切です。

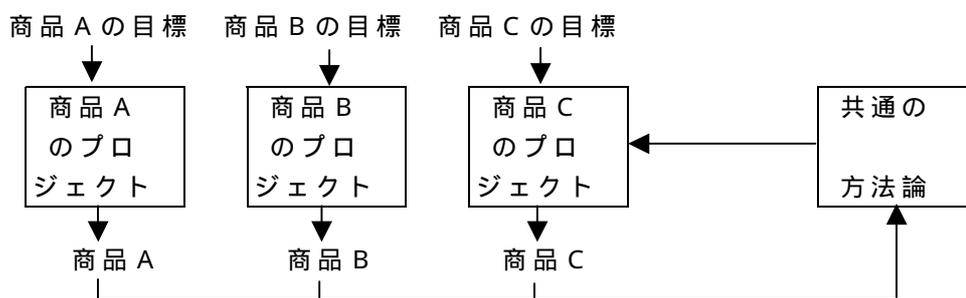


図 1 個別目標の展開と方法論の共通化

質問 4 個々のプロジェクトの結果を設計・開発プロセスの視点から見直す方法、注意点を教えてください。

第一に、開発した商品の販売後の評価結果を踏まえた上でプロジェクトごとの反省を行うことが前提となります。多くの企業ではプロジェクトの終了時に反省を行っていますが、具体的なデータに基づく議論でなければ、抽象論、あるべき論に終始し、改革を生み出す真の原動力となりません。いろいろな企業を見てみると、市場に製品が導入され、ある程度の評価が見える時期には、プロジェクトに関わった大部分の設計者、技術者は次のプロジェクトに移っており、十分な反省が行われないうまま、同じ誤りを繰り返していることが少なくありません。発生した問題を解析し、設計・開発のプロセス・進め方を見直し・改善する活動は、当該商品の設計やその生産・提供プロセスの設計のまずさを手直しする活動とは本質的に異なります。後者はプロジェクトに参画していない他の設計者、技術者にも可能ですが、後者は当該のプロジェクトにおいてそれらがどう行われたのかを実際に知っている人の協力が得られなければできません。プロジェクトに参画した中心メンバーの協力を得て、市場で発生した様々な問題を議論できる場を設定することが必要です。

発生した一つの問題を分析する視点は様々です。設計・開発プロセスを見直すためには、その問題がどのような技術的なメカニズムによって生じたのかという議論だけでは不十分です。質問 1 の解説で論じたように、複数の商品の開発に共通する要素は 3 つありますが、各々の視点からの適切な分析が行われてはじめて、他の商品の開発に利用できる意味のある対策が検討できます。例えば、「設計そのもの共通性」を利用することを考えるなら、発生した不具合が新たに設計し直した部分が既存の設計を流用した部分か、新規に設計する部分をどのような基準・考え方で選

んだのか、既存設計の部分の正しさをどのように確認したのかなどを明らかにし、その悪さを明らかにしなければなりません。また、「技術の共通性」を利用することを考えるのなら、当該の技術及びその利用の仕方を規定した標準類、その作成方法、使用の仕方などとの関連を付ける必要があります。技術標準に当該の問題についての記述が抜けていたというだけの解析では不十分です。技術標準が従来経験した問題を単にリスト化したものとなっている場合、これに抜けていた一項目を追加するのは、もぐらたたきの繰り返しに過ぎません。経験した問題をリスト化するという技術の蓄積・利用の仕方そのものを見直さない限り同種の問題が次から次へと発生することになります。最後の「設計・開発プロセスの共通性」を利用する場合には、問題をシステムのどの部分に起因するものかという視点から解析し、当該の部分のしくみ、およびそれを規定している標準類を改訂する必要があります。表2にT型マトリックスを使った解析例を示しますが、このような立場からの解析の好例と言えるでしょう。

質問5 設計における人のミスが多く困っています。人のミスに起因するトラブルを分析する場合、注意すべき点はなんですか。

人のミスの発生に影響を与える要素は、大きく

- (1) 示票の書式、設備、作業の順序などの作業方法に関するもの
- (2) 作業に従事する人の注意力、知識、技能、モラルなどの

の2つに分けられます。ここで、(2)の要因を改善することはミスを防止するための重要な要素ですが、疲労や慣れにともなう注意力の低下は避けられずこれだけでは十分な効果を期待できません。標準的な人であれば経験が無くともどのような体調や精神状態で作業してもミスしないようミスしても大丈夫なように作業方法を変更すべきです。

作業ミスの防止活動を組織的に推進する場合、第一に、作業ミスは作業員個人の問題でなく、プロセスの設計と密接に関係する管理上の問題であることを管理者を含めた職場の全員に徹底させる必要があります。このためには、作業ミスに起因する品質トラブルや事故の原因調査を行うための様式を作業管理の悪さを解析できるものに変更することが重要です。表3は「作業管理の仕組みの悪さ」の点から作業ミスを分類するためのものです。この表に基づいて発生している作業ミスを分類し、最も多く発生しているミスのタイプを求めることによって、作業管理の仕組みの悪さを明らかにすることができます。また、表3の解析の結果、タイプ のミスが多いことが分かった場合には、作業方法をどのようなところがミスを引き起こしやすいものになっているのかという点から解析することが必要となります。このためには表4に示すようなFMEAを用いた作業の分析を行うことが有効です。

特に、設計・開発におけるミスを分析する場合について言えば、一つのミスの事例から得られる次の3つの情報を区別しておくことが重要です。

- (1) Fault: 設計内容を規定している図面やプログラムにおける間違い。
- (2) Error: 設計者やプログラマーがちょっとした忘れ、勘違い、見間違いなどから犯す、作業における標準又は目標からの逸脱。
- (3) Process Flaw: エラーの発生やその検出に影響する開発方法・開発手順の問題点。

表5はソフトウェア開発における人のミスについてこのような視点より分析を行

表2 T型マトリックスを用いたプロセスの評価

	販売サービス	本生産	工場試作	生産準備	製品化計画	開発試作	詳細設計	基本設計	研究試作	製品構想	開発提案	作り込んだ工程	発見した工程	開発提案	製品構想	研究試作	基本設計	詳細設計	開発試作	製品化計画	生産準備	工場試作	本生産	販売サービス	発見すべき不良の件数		発見すべき不良の構成比率		見逃し件数	見逃し率
																									個別	累計	個別	累計		
											8	開発提案		7									1		8	8			1	12
										5	1	製品構想			4	1							1		6	14			2	33
									3	5		研究試作				3					4			1	8	22			5	62
								10				基本設計					2						5	3	10	32			8	80
							10	5	3	3		詳細設計						11	5		2		1	2	21	53			10	48
									2			開発試作								2					2	55			0	0
						3				5		製品化計画									7		1		8	63			1	12
						1					1	生産準備										1		1	2	65			1	50
				1		3						工場試作										2	1	1	4	69			2	50
		2	2	3		1	2	5	2		2	本生産											12	5	17	86			5	29
	2											販売サービス												5	5	91			1	25
作り込んだ不良の件数	個別	3	2	2	3	1	5	15	20	10	18	12		7	4	4	2	11	7	11	3	3	21	18	個別	発見した不良の件数				
	累計	91	88	86	84	81	80	75	60	40	30	12		7	11	15	17	28	35	46	49	52	73	91	累計					
作り込んだ不良の構成比率	個別																								個別	発見した不良の構成比率				
	累計																								累計					

表3 作業管理の仕組みの評価のための作業ミスの分類とその対策

ミス発生時の作業状況			ミスの発生傾向	ミスに対する対策		
				個別の対策	水平展開による対策	仕組みに対する対策
) 標準作業が確立してなかった (どのような作業が正しい作業か決まらなかった)	a) 標準作業を決めてなかった	必要がないと判断した 作れなかった	<ul style="list-style-type: none"> 同一作業者が行う作業全てに系統的に発生する 作業者の交替時又は新人作業者の時に発生する 量産立ち上がり時又は小量生産品に多い 1作業当たりの発生率が高い (数~0.1%程度) 	・ミスが起こった作業について標準作業を決める	・他の作業について標準作業が決まっているか見直す	・標準作業の設定に関する判断基準、手順、担当などを明確にする
	b) 正しく行なえば問題が発生しない標準作業にしてなかった	問題に気づいてなかった 十分な対策を取ってなかった		・ミスが起こった作業について生産技術の面から標準作業を再検討し、改訂する	・他の標準作業について内容を見直す	・標準作業の設定における技術上の注意点を明確にする(技術標準の作成)
	c) 適切な標準化を行ってなかった	使える形に保守してなかった 理解・実施しやすいものにしていなかった		・標準作業を最新のものにする ・手順を具体化する ・写真、図、見本等を活用する	・他の標準書について管理状態を見直す	・標準書の作成、管理、改訂の仕組みを改善
) 標準作業は確立していたがミスが起こった時、作業者はそれに従って作業してなかった	a) 正しい標準作業の方法を理解させてなかった	教育してなかった 作業者が正しく覚えているか確認してなかった	<ul style="list-style-type: none"> 同一作業者が連続して行った作業の中、1回又は数回発生する 疲労時や作業負荷の大きい時に多い 1作業当たりの発生率が低い (0.1~0.01%程度) 	・ミスを起こした作業者に対して標準作業の教育を行なう	・他の作業者について標準作業通り作業しているか、技能的に無理がないか見直し、必要な処置を取る	<ul style="list-style-type: none"> 標準作業の教育・訓練の仕組みを改善する 技能の資格制度を制定又は改善する 職場配置の決め方を見直す
	b) 標準作業通り作業するのに必要な技能を考慮してなかった	技能訓練を行ってなかった 作業者の技能に合った作業を与えていなかった		・ミスを起こした作業者に対して標準作業の実施に必要な技能の訓練を行なう ・作業者の職場配置を替える		
	c) 標準作業通り作業する重要性を理解させてなかった	標準作業通り作業するように指示・指導してなかった 標準作業通り行わない場合の危険を周知してなかった 標準作業通り作業するのに必要な時間を与えてなかった		<ul style="list-style-type: none"> ミスを起こした作業者に標準作業通り作業するよう指導、指示する ミスを起こした作業者に標準作業通り作業しない場合の危険について分からせる 標準作業通り作業する時間を与える 		
) 標準作業が確立しており、ミスが起こった時も、作業者はそれに従って作業していた	a) 作業に対する外部からの影響を考慮していなかった	作業計画について十分検討していなかった 作業環境についての配慮が欠けていた		・ミスが起こった作業について作業計画及び人員計画を変更する ・ミスが起こった作業の作業環境及び余裕率を改善する	・他の作業について作業計画、環境について同様の問題がないか見直し、改善する	<ul style="list-style-type: none"> 作業計画の立て方を改善する 作業環境の評価・管理方法を見直す
	b) 標準作業に対してミスの観点から改善を行ってなかった	ミスの危険性に気づいてなかった 作業方法の改善が不適切又は不十分であった		・ミスが起こった作業について当該のミスに対する作業方法の改善を行なう	・他の標準作業について同様のミスによる危険がないか見直し、対策する	・作業ミスの危険性評価を含めた作業設計の方法を改善する

表4 作業FMEAの例

No	作業要素	エラーモード	影響	発生原因	発生可能性		影響の致命度	波及防止度		F P 化 必 要 性
					実績	可能性		異常度検	影響緩和	
1	部品Aを取る	抜け	A欠品		0	0		2		0
2	部品箱から ネジを取る	抜け	A欠品	類似作業の繰り返し	0	0		2		0
		選び間違い	Aの緩み	種類不明確、標準書との対応が複雑	0	1		4		4
3	良否を確認する	抜け	外観不良	付随的作業、動作を伴わない	0	1		3		3
4	部品箱から スプリングを取る	抜け	Aの緩み	類似作業の繰り返し、付随的作業	2	1		3		9
		選び間違い	Aの緩み	種類不明確、標準書との対応が複雑	0	1		4		4
5	良否を確認する	抜け	外観不良	付随的作業、動作を伴わない	0	1		3		3
6	ネジにスプリングを入れる	抜け	Aの緩み	実施結果が外観で不明、付随的作業	2	1		3		9
7	ネジで部品Aを 本体に仮組みする	抜け	A欠品		0	0		2		0
		位置の間違い	Aの作動不良		0	0		2		0
8	電気ドライバーのトルク設定	抜け	Aの緩み	やったりやらなかったりする、付随的作業	0	1		3		3
9	電気ドライバーで本締めする	抜け	Aの緩み	実施結果が外観で不明、類似作業の繰り返し	2	1		4		12

実際の
ミスの有無

ミスの起こる
可能性

有 : 2 大 : 2
無 : 0 中 : 1
 小 : 0

「実績」と「可能性」の合計を
「発生の可能性」とする

ミスを見える
可能性

大 : 0
中 : 1
小 : 2

自工程と後工程の評価点の
合計を「発見の可能性」とする

フルプルーフ化
(FP化)の必要性

「発生の可能性」×
「異常検出能力」

(注)「影響緩和度」と
「影響の致命度」は評価しない

った例です。一件一件のトラブルについてこのような分析を行った後、Fault、Error、Process Flawをそれぞれの類似性に基づいて分類し、FaultとError、ErrorとProcess Flawの相互関係について二元表を用いて解析することにより、エラー発生メカニズムを構造的に把握することができます。

表5 エラー分析の例（ソフトウェア開発の例）

情報の種類	観測された情報
Fault	・ファイルからデータを誤った順序で読み込む
Error	・ファイルにおけるデータ格納順序を誤解した
Process Flaw	・ファイルにおけるデータの格納順序が読み込んだデータを入れる共通変数の順序と一致していなかった ・ファイル構造に関する定義をコーディング中に簡単に参照できるようになっていなかった

質問6 人のミスを、作業方法を変えることによって防止する場合、どのような点に着眼して進めればよいのでしょうか。

作業方法を変えてミスの発生を防止する工夫には様々なものがありますが、それらは大きく二つのタイプに分類されます。一つは、「エラーが起こらないようにする」というヒューマンエラーの発生過程に着目したものであり、もう一つは、「エラーが品質トラブルや事故につながらないようにする」という影響の波及過程に着目したものです。

ヒューマンエラーの発生を防ぐ最も効果的な方法は、「作業を行わなくても済むようにすること」です。次に、これが不可能な場合の方法としては、「作業を人間に任せないようにすること」や「作業を人間にとって容易なものにすること」が考えられます。他方、ヒューマンエラーの波及を防ぐ一つの方法としては、「エラーによる異常を検出し処置すること」が考えられます。またもう一つの方法としては、「エラーの影響を緩和するような作業や緩衝物を組み込んでおくこと」が考えられます。以上の5つがフルプルーフ、すなわち作業の方法を改善することによってヒューマンエラーによる不具合の防止を行なう場合の基本となる考え方であり、それぞれ『排除』、『代替化』、『容易化』、『異常検出』、『影響緩和』と呼ばれています。これらの原理を頭にいれておくことは、様々な人のミスの防止策を考える上で非常に役立ちます。また、ミスが発生する危険性がないかという観点から作業方法を見直す場合にも有効です。

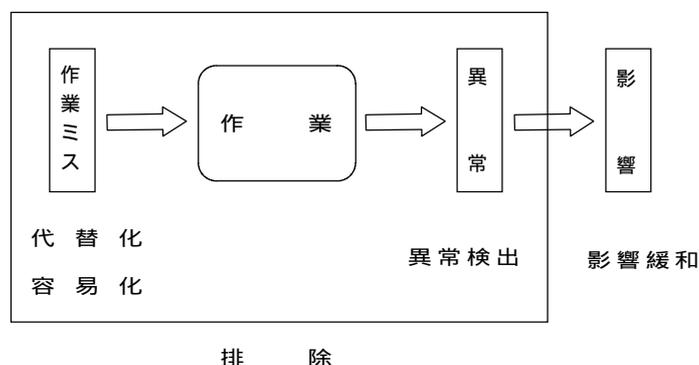


図2 フールプルーフ化の原理

設計・開発工程では多数の設計者や技術者の間の情報伝達エラーが問題となる場合が少なくありませんが、このような人のミスを防ぐためには、各々が担当する設計間のインターフェイスを文書化しておくことが有効です。表6は、発生防止の原理である『容易化』に含まれる3つの考え方

(原理1) 適合化：作業の対象を人間の能力に合ったものにする

(原理2) 特別化・個別化：作業者に要求される機能の変化や相違を混同しないように、作業の内容や対象の差を鮮明にする

(原理3) 共通化・集中化：作業者に要求される機能上の変化や相違を少なくするために、作業の内容や対象を時間的・空間的に共通にする、あるいは、関連する作業の内容や対象を時間的・空間的に集中する

をもとに、設計における文書化を考える場合の3つの問題

(問題1) どのような種類の情報を含めるか

(問題2) どのような階層構造を持つ文書にするか

(問題3) 個々の項目をどのような様式や表記で記すか

にそれぞれ適用することによって、インターフェイス仕様書が情報伝達エラーを防止する上で有すべき性質を整理した結果です。例えば、問題1については記載する情報の量を制限した上で、重要な情報及びそれらに関連の深い情報を明らかにすることが必要なことを、問題2については特別の注意を要するものを別にする、文書の構造を複雑にしない等の配慮を行った上で、文書の構造を作業の構造にできるだけ一致させることが必要であることを示しています。

表6に示されたような性質を満たすインターフェイス文書を見いだす方法は一通りではありませんが、当該インターフェイスの不具合を調査し、エラーの発生に直接の関係する設計情報を列挙する、得られた情報を使用する設計作業を解析し、その構造を明らかにする、作業構造と同じ構造を持つ一連の文書群を考える、情報を混同しないよう図や色を適切に活用する。等により、適切なものを見いだすのがよい。図3にこのようにして得られた文書の一例を示す。

表6 エラー防止の観点からインターフェイス仕様書が満たすべき性質

問題 \ 原理	適合化	特別化 個別化	共通化 集中化
どのような種類の情報を含めるか	一つの文書に含まれる情報の量を制限する	ミスしやすい情報を重点的に含める	関連する情報は一連の文書にまとめる
どのような階層構造を持つ文書にするか	文書の構造をできるだけ単純にする	特別の注意を要するものは別の文書とする	設計作業の構造と文書の構造を一致させる
個々の項目をどのような様式で記すか	文字等の大きさを適正に保つ	情報を混同しないよう図や色を用いて識別する	同種の情報については同じ表記法を用いる

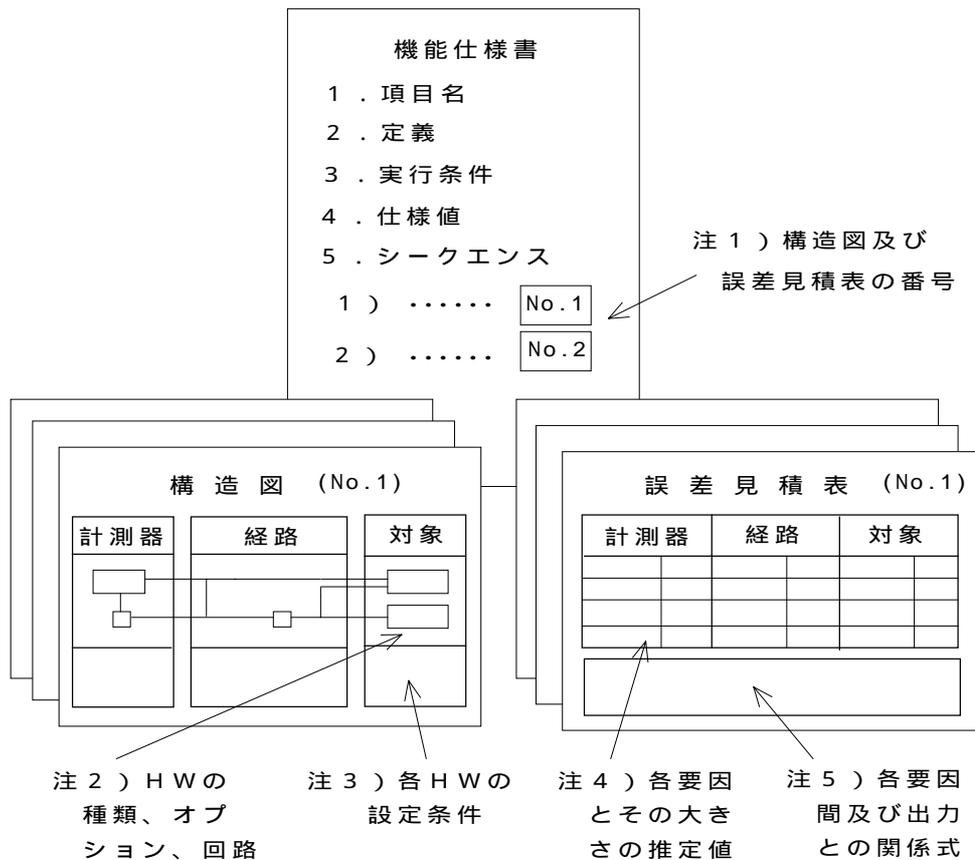


図3 インターフェイス仕様書の例
(計測システムの開発における
HW技術者とSW技術者の間のインターフェイス仕様書)

質問7 デザイン・レビューとは何ですか。

商品の最終的な品質はその設計開発段階の活動で大きく左右されます。その意味では、設計・開発プロセスのできるだけ早い段階で、問題を顕在化させ、必要な処置をとることが必要となります。しかし、商品の高度化、複雑化にともなって、顧客の品質要求も多様化しつつあります。機能・性能、安全性、信頼性、保全性、使いやすさ、デザイン、価格、ライフサイクル・コスト、環境性など総合的な視点からの検討が必要となっています。また、商品およびその生産・提供プロセスに使われる技術も量的に質的に大きく変化しています。このような中で、一人の設計者、技術者がすべての面に目をくばり、適切な決定を行うのは至難のわざと言えます。

デザインレビュー (Design Review) は、商品の設計品質及びそれを具現化するために計画された製造・輸送・据付け・使用・保全などのプロセスについて、客観的に知識を集めて評価し、改善点を提案し、次の段階に進みうる状態にあることを確認する組織的活動の体系です。通常、設計者が図面やその他の設計資料を他の技術者に提示し、自分では気付かなかった設計の問題点や改善方法を指摘してもらい行われます。

デザインレビューは、1名の委員長と複数のメンバーで構成されるチームによ

って行われます。委員長は技術知識が豊富で、設計の経験があり、決断力があることが必要です。担当役員、技術・設計部門の部長・課長、品質保証部門の部長・課長などが勤めている場合が多いようです。メンバーの選定は、委員長に誰がなるかということと合わせてレビューの成否を決する重要な要因となります。デザインレビューの目的の一つは様々視点からの検討を加えることですので、できる限り広い範囲の専門領域から参加者を含めるのがよいでしょう。ただし、なんでもかんでも全ての部門の人間が集まるというのでは、参加者の時間を浪費するだけでなく、レビューのねらいが曖昧になりかねません。実施するレビューのねらい・検討範囲を考慮して選ぶという立場が重要です。

デザインレビューは次の手順で行われるのが普通です。

- (1) 計者は設計図面やその他の資料をレビュー・メンバーに事前に配布する。最低1週間前までに渡すのがよいとされています。
- (2) メンバーは配布された図面や資料を事前に検討しておく。
- (2) デザインレビューの会合においては、まず設計者がその設計内容について説明を行い、メンバーが問題点を指摘する形で進める。次に出された問題点について討議し、改善方法の検討を行う。
- (4) 問題点及びそれに対する対策、要検討事項などを議事録にまとめる。この時、誰がいつまでに処置を取るのか明確にしておく。
- (5) 次回の会合等の適切な時点で、対策や要検討事項が確実に処置されたかどうかフォローする。

1回の会合は2時間程度が限度と言われています。したがって、複雑なシステムになると数回に及ぶ場合もあるので、事前の計画・スケジュール調整が必要となります。また、検討にもれがでないよう、(2) 等の実施に際して、チェックリストを活用する場合も少なくありません。

質問 8 デザインレビューを成功させるポイントを教えてください。

他の部署や他の経験豊富な人から意見を貰ながら、自分の責任であるこの設計内容をよりよいものにするために行うのだという、レビューの役割・位置づけに関する全員の理解が成功のための第一のポイントです。

第二に、討議・検討する範囲を明確にしておくこと、レビューのねらい・レビューすべき範囲を適切な大きさに区分することが重要です。一度にたくさんのごとをレビューするのは不可能です。デザインレビューは大きく

- (1) 構想デザインレビュー
- (2) 中間デザインレビュー
- (3) 最終デザインレビュー

の3つに分けて考えることができますが、このように設計・開発プロセスの各段階に対応してデザインレビューを計画することで、各々のレビューの目的・範囲を明確にすることができます。また、それぞれ段階においても、いくつかの別々のねらい・検討範囲を絞ったデザインレビューを計画するのが有効です。例えば、商品の開発を次の段階に進めてよいかどうかを判断するためのフォーマル・レビューと設計者が気付いていない問題の摘出と解決策の検討を中心とするインフォーマル・レビュー、コンポーネントごとのレビューとコンポーネント間のインターフェイスを含めたシステム全体のレビューなどです。

第三に、デザインレビューが適切に行われるためには、レビューの対象となる設計が他の人に見える形になっている必要があります。何の資料も準備せずに単

に頭のなかにある内容を説明するという形でデザインレビューを行っても殆ど効果がありません。デザインレビューを成功させるためには、設計・開発プロセスの節目において設計内容を可視化する方法を確立すること、設計の可視化とデザインレビューを連動させることが大切です。

設計開発におけるトラブル予測の重要性については既に述べましたが、デザインレビューはこのトラブル予測を周知を集めて行う活動と言うこともできます。したがって、設計者がどのようなトラブル予測を行ったのか、予測しているトラブルに抜け落ちがないか、トラブルの影響評価は正しく行われているかをレビューすることが効果的な実施のためには欠かせません。設計内容だけでなく、FMEA等のトラブル予測の手法を適用し、その結果をレビューの対象とすることが重要です。

最後に、デザインレビューは即効性の方法ではありません。デザインレビューを導入した最初はなかなか効果が見えない場合も少なくありません。長期的な視野に立ってデザインレビューを継続的に実施していくことも成功のポイントです。

質問 9 デザインレビュー用のチェックリストを作成する場合、注意すべき点は何ですか。

設計・開発の現場では様々なデザインレビューのための様々なチェックリストが考案され、活用されています。これらのチェックリストに含まれている項目を見ていくと

(1) 行うべき仕事が行われているかどうかをチェックする項目

(2) 行われた仕事の内容に問題が含まれてないかチェックする項目

の二つがあることが分かります。(1)は、対応する仕事の内容から系統的に抽出できるものであり、これらを抜け落ちのないものにするのはそれほど難しくありません。他方、(2)は、可能性としては無数の問題が存在し得るため、全てを網羅するのは不可能です。この困難さを克服するための解答は一通りではありませんが、(2)のチェック項目を作る場合には少なくとも以下のような点を考慮するのがよいでしょう。

(1) 個別の問題を羅列するのではなく、様々な設計に共通的に適用できる問題としてまとめる。

(2) 問題をまとめる場合、結果に着目した記述、原因に着目した記述など色々考えられるが、結果、原因のいずれに寄りすぎても多様化する傾向にある。原因から結果に至る過程の中で共通性の最も高いものに着目する。

(3) チェックリストにまとめられている各々の問題が現在検討している設計

に当てはまるかどうかを考えるためには、検討の対象を考えやすい大きさに区分した上で検討する必要がある。問題のリストと合わせてこの区分の大きさ・仕方を決めておく。

(4) リストされている問題が発生するかどうかの判断は、どのような状況を想定しているかに大きく依存する。考慮すべき状況についても合わせてキーワード化しておく。

質問 10 変更管理のポイントを教えてください。

設計・開発段階で品質トラブルを引き起こす主要な原因の一つに変更に関連するものがあります。企画の変更、設計審査や試作品のテストで検出された設計の

不具合、設計部門あるいは設計者自らによって見出された設計ミス、コスト・ダウン要請などにより、多くの設計変更が発生します。設計変更は決して望ましいものではなく、これを最少に抑えることが必要ですが、設計作業が人間の営みである以上、変更を一切行わずに済ませるということは、ほとんど期待できません。

開発した商品に欠陥があれば、これを是正するための設計変更は行わざるを得ません。問題は変更によって引き起こされる二次、三次のトラブルです。これらのトラブルの原因は、

- (1) 変更の技術的検討が不十分で、変更内容そのものが悪い（材料選定の誤り、強度不足など）
- (2) 変更が他に及ぼす影響を十分考慮せず、目前の改良だけを考えている（干渉、両立性・互換性不良など）
- (3) 変更の実施が徹底せず、変更が未実施あるいは一部のみで行われる
- (3) 変更指示内容の誤り、変更内容の誤解などのため意図通りの変更が実施されない

などに分類することができます。これらの原因の背景には、変更の難しさに対する認識が不足しており、軽い気持ちで変更を行ってしまうという風習があるかもしれません。変更をトラブルなしに行うためには、変更の難しさを設計者や管理者が十分に認識することが必要です。さらに、これが徹底したとしても変更を行う手続きが適切に定められ、個々の変更がこの方法に従って行われなければ、変更によるトラブルはなくなりません。以下の点を考慮した上で、変更を実施する際の手続きを標準化しておくことが必要です。

(1) 変更の要求・提案

変更要求を行う担当者あるいは部署は、変更を必要とする理由、変更の緊急度、変更時期、変更内容、変更にあたって考慮すべき事項等を取りまとめて、変更を実施する部署、変更の審議を行う組織に提出することが必要です。

(2) 変更内容を検討するための組織

変更の必要性、変更内容の妥当性の検討やチェックを行う組織は、変更の要否を決定する権限を有する人を長として、審査を行う能力・知識を有する関連部署の代表者で構成されたチームで行います。変更内容の規模に応じて適切な検討チームが編成される弾力的な運営が必要ですが、一方、製品の構成(configuration)は開発責任者において常に把握・確認できるようになっていなければなりません。

(3) 変更にあたって検討すべき事項

設計変更の検討にあたっては、総合的な見地（設計の基本理念、製品としてのバランス、開発日程への影響、製品の機能・性能、信頼性、安全性、調達、製造、運転性、保守性、製造コスト、ライフサイクル・コスト、法規など）から行うこと、変更に関係するハードウェア（部品、サブユニット、サブシステムなど）、及び関係する組織のインターフェイスを明確にし、技術検討の落ちや谷間をなくすことが大切です。

設計の変更の検討にあたっては、まずその変更要求の妥当性が検討されなければなりません。これについては以下の検討を行います。

- ・不具合事象の把握、解析は要求・提案者によつて的確に行われているか。
- ・提案されている方法以外の他の処置によって解決はできないか。
- ・その変更によって基本設計との整合性がくずれないか。
- ・その変更によって確実な効果が期待できるか。
- ・変更要求時期は妥当か。

第二に、ある箇所を変更すると、それと併せて他の部位の変更が必要となることが少なくありません。組立製品の部品、サブユニットについて組付部の寸法形状が変更された場合は、当然相手の部品の寸法も変更されなければなりません。また1つの部品が多くの機種に用いられている場合は、この変更はこれらの機種のすべてにおいて実施されなければなりません。部品や材料が低位の基本的なものであれば、それだけ影響は広い範囲に及びます。この関連変更を漏れなく行うためには、該当部品がどこにどれだけ使用されているかを示す表“where use list”を備えていることが必要です。

第三に、欠陥を是正するための設計変更が、それまで問題のなかったところで問題を引き起こすことが少なくありません。材料や構造の変更は一般にこの種の問題を引き起こしやすく、変更にあたってはFMEAなどによる不具合の事前評価、試作機による試験がなされなければなりません。設計・開発の初期段階での設計変更は、設計・開発プロセスがしっかりしていれば、設計審査や試作テストにおいてチェックされるため、設計変更の問題があったとしても、それを未然に防止できる可能性が大きいと言えます。しかし、かなり後になって行われた設計変更は、うっかりすると必要なテストが行われないうまま変更が実施されてしまい、後になって問題を起こすので特に注意が必要です。

(4) 変更実施のための文書

変更を正確に組織的に実施するためには、変更の内容を明確にし、変更において検討すべき事項およびそれによって発生する仕事を確実に伝える文書が必要です。これには、変更登録番号、変更の名称、変更を行うアイテム、変更内容、変更の理由、変更の影響を受ける他のアイテム・作業、開発日程に対する影響、変更実施に伴って発生するコスト・変更実施後の品質確認の方法、すでに本生産に入っている場合には、変更実施号機などが明確に記述されなければなりません。

参考文献

- [1] 久米均 (1999) ; 「設計開発の品質マネジメント」、日科技連。
- [2] 中條武志 (1993) ; "ヒューマンエラー事例に基づく作業管理システムの評価", 「品質」, 23, [3], 105~113.
- [3] 中條武志・久米均 (1985) ; "作業のフルプルーフ化に関する研究 - 製造における予測的フルプルーフ化の方法", 「品質」, 15, [1], 41~50.
- [4] T. Nakajo, I Azuma and M. Tada (1993) ; "A Case History Development of a Foolproofing Interface Documentation System," IEEE Trans. on Software Engineering, Vol.19, No.8, pp.765-773.
- [5] 中條武志・久米均 (1984) ; "作業のフルプルーフ化に関する研究 フールプルーフ化の原理", 「品質」, 14, [2], 128~135.
- [6] 中條武志・久米均 (1985) ; "作業のフルプルーフ化に関する研究 - 製造におけるフルプルーフ化の方法 (1)", 「品質」, 15, [4], 350~359.
- [7] 中條武志・久米均 (1986) ; "作業のフルプルーフ化に関する研究 - 製造におけるフルプルーフ化の方法 (2)", 「品質」, 16, [1], 4~13.
- [8] T. Nakajo and H. Kume (1991) ; "A Case History Analysis of Software Error Cause-Effect Relationships," IEEE Trans. on Software Engineering, Vol.17, No.8, pp.830-838.
- [9] 菅野文友、額田啓三、山田雄愛 (1993) ; 「日本的デザインレビューの実際」、日科技連出版社。