

# エンジニアリングから学ぶ業務構造化

谷脇 孝一郎：株式会社クロスフィールド

## 1. はじめに

BPR (Business Process Re-engineering) を行なう上で、「業務の可視化」は欠かすことのできない最初のステップです。しかしながら、「とりあえず個々のパターン全てについてフロー図を作成し可視化してみたが、どこから見直していけばよいか見当がつかない」といった悩みを抱えることも少なくないと思います。

そこで本稿では一つの提案として、エンジニアリングにおける「構造的思考」を用いた業務の「構造化」と「見直し」について述べたいと思います。問題の発見と修正・改善を絶え間なく行なうエンジニアリングの世界で培われたノウハウは、業務のPDCA サイクルにも通用する場面は少なくないと考えます。

## 2. エンジニアリングにおける「構造」と構築手法

まずここで言う「構造」とは部品等のつくりそのものを指すものではなく、各要素の機能的な関連性をまとめたものを指します。

製造業では「BOM (Bill of Material)」と呼ばれる、製品を構成する部品、組立を想定した階層構造を表す一覧を作成することが一般的です。階層構造にすることで、上層・中層で凡その機能を把握することができるため、全体管理する上で見るべき情報を最小限に留めることができます。これを業務の可視化に置き換えると、複数の業務プロセスに含まれる要素が一元的に参照できる形に整理されるため、業務フロー図を1本1本見るよりも容易に自社の業務全体を把握することができます。更に業務個別の独自条件を下層に切り出すことで、管理の観点として把握すべき情報の単純化、業務改善の観点としては標準化対象を発見しやすくする効果も期待できます。

ここでは具体的な例を用いて、複数の業務プロセスについてフロー化と階層構造化を行なってみます。

**業務プロセス 1：取引先からの受注に合わせ商品を在庫から探し出荷・納品する**

**業務プロセス 2：取引先からの受注量を基に生産量を増やし工場から出荷、倉庫へ納品する**

これらを単純にフロー化すると下図のようになります。



では次に業務プロセス 1、2 を基に階層構造を定義してみます。

まず階層の最上位を業務領域で定義します。

例えば業務領域を“販売”、“物流”、“製造”と整理した場合、下図のように整理することができます。

階層1	階層2	プロセス1	プロセス2
販売	受注	○	○
	在庫確認	○	○
製造	生産計画		○
	生産		○
物流	工場出荷		○
	倉庫納品		○
	倉庫出荷	○	○
	客先納品	○	○

フロー図と同様の情報を階層2から把握できるだけでなく、2つのプロセス間に重複して存在する業務が集約され、より見やすくなります。

更に、ここで階層2から類似業務の個別条件を下層に切り出すことで、下図のように階層2をよりシンプルにすることができます。

階層1	階層2	階層3	プロセス1	プロセス2	
販売	受注		○	○	
	在庫確認		○	○	
製造	生産計画			○	
	生産			○	
物流	出荷	工場から出荷		○	
		倉庫から出荷	○	○	
	納品	倉庫へ納品			○
		客先へ納品	○		○

階層2が単純化され、下層を見れば個別運用の部分が容易に把握できるようになりました（上記例で言えば物流領域の業務は出荷と納品に大別され、その先のバリエーションは配送元・配送先の違いでしかないことが整理されています）。フロー図では記載された粒度でしか業務内容を確認できませんでしたが、階層構造では目的に応じたレベルで、業務領域のみ知りたければ階層1のみ見る、各業務領域の業務概要を把握する必要がある場合は階層2、より詳細を知りたければ階層3を見るといった使い分けが可能になっています。

一方、階層構造にするとフロー図と比べて業務の流れが定義できないのでは？と思われるかも知れませんが、下図の様に階層構造にアドレスを設けることで業務手順を表現することも可能です。この時、最下層（以下では階層3）は空白にせず、最低限階層2と同じ表現を使う必要があります。

階層1	階層2	階層3	プロセス1	プロセス2	
1. 販売	1-1. 受注	1-1-1. 受注	○	○	
	1-2. 在庫確認	1-2-1. 在庫確認	○	○	
2. 製造	2-1. 生産計画	2-1-1. 生産計画		○	
	2-2. 生産	2-2-1. 生産		○	
3. 物流	3-1. 出荷	3-1-1. 工場から出荷		○	
		3-1-2. 倉庫から出荷	○	○	
	3-2. 納品	3-2-1. 倉庫へ納品			○
		3-2-2. 客先へ納品	○		○

階層構造は通常、上層→下層の順に定義しますが、逆に最下層に残りそうな細かい業

務（一部取引、個人の運用でしか発生しないような業務）を洗い出してから上位の階層に仕分けていく手法もあります。これは、製造において「リバースエンジニアリング」と呼ばれる手法で、全社の業務を漏れなく階層構造に落とし込む際には有効なアプローチです。具体的には、独自の業務やルールが、全体から見るとどこまで有用かを評価した上で、最下層（個別）として細分化すべきか、他の業務と統合すべきか、あるいは上位の階層に組み込むべきかを判断します。

### 3. エンジニアリングから学ぶ構造的思考プロセス

次に、階層構造化された業務を継続的に改善していく上で注意すべきポイントをお伝えしたいと思います。

#### ① 課題発生時は必要条件に立ち返る

階層構造においては、上層と下層の矛盾や下層に合わせた上層の見直しは、原則的には認められません。数学的に言えば、上層は必要条件にあたり、下層は必要十分、最下層は十分条件にあたります。簡単に見えますが、前述の「リバースエンジニアリング」の手法に基づいて、より業務の実態に近い情報や新しく発生した（しそうな）業務を階層構造に落とし込もうとした際に、誤った整理をしないために常に意識しておく必要があります。具体例として、先ほどの例に「出荷実績を報告する」という業務を追加するケースを説明します。

「出荷実績を報告する」を追加すると下図のようになります（説明の都合上、階層4まで細分化しています）。

階層1	階層2	階層3	階層4
3. 物流	3-1. 出荷	3-1-1. 工場から出荷	3-1-1-1. 工場から出荷
		3-1-2. 倉庫から出荷	3-1-2-1. 倉庫から出荷
		3-1-3. 出荷実績を報告	3-1-3-1. 出荷実績を登録する
			3-1-3-2. 出荷実績データの送信
		3-1-3-3. 出荷報告書の送付	

ここで3-1-3-3.「出荷報告書の送付」において、相手先から要求された出荷報告書の作成・送付タイミングに出荷作業の完了時間が間に合わない、という事象が発生したとします。応急策として「予定情報を基に出荷報告書を作成する」という手段をとった場合、下図のように展開してしまうと問題があります。

階層1	階層2	階層3	階層4	階層5
3. 物流	3-1. 出荷	3-1-3. 出荷実績を報告	3-1-3-1. 出荷実績を登録する	
			3-1-3-2. 出荷実績データの送信	
			3-1-3-3. 出荷報告書の送付	3-1-3-3-1. 予定ベースで作成
				3-1-3-3-2. 実績ベースで作成

一見、階層4と階層5で業務が成立しているように見えますが、階層3では「出荷実績」に関する業務を定義しており、その下層である「出荷報告書の送付」は「出荷実績に基づく」という前提があるため、改めて階層3から階層5までを並べると、「実績」と「予定」で矛盾が生じていることが分かります。

このような状態を看過すると階層4までを見て業務プロセスを定義していった際に、時系列を誤ってしまう可能性が出てきます（「出荷実績を登録」→「出荷報告書を作成」と考えていたが、実運用は「出荷報告書を作成」→「出荷実績を登録」だった、

など)。

正しくは階層3から切り分け、以下のように整理すべきです。

階層1	階層2	階層3		階層4		階層5		
3.	物流	3-1.	出荷	3-1-3. 出荷予定を報告	3-1-3-1.	出荷報告書の送付	3-1-3-3-1.	予定ベースで作成
				3-1-4. 出荷実績を報告	3-1-4-1.	出荷実績を登録する		
					3-1-4-2.	出荷実績データの送信		
					3-1-4-3.	出荷報告書の送付	3-1-4-3-1.	実績ベースで作成

## ② 表現と解釈は互いに一意にする

曖昧な表現や独自の表現は書き手個人としては便利ですが、読み手側に解釈を委ね、全く違う内容で受け取られてしまうといったことも起きかねません。例えば「在庫の保管場所を変える」業務を単に「移動する」と表現すると、「ヒトが移動する」なのか「モノを移動する」なのかわからなくなります。また、「在庫を移動する」と「在庫を転送する」が混在すると、この2つの表現が同一の業務を指すものか否か判断しづらくなります。構造の全体が大きくなるほど複数人での管理は避けられませんが、階層構造の肥大化・複雑化を抑制するためにも表現の統一は不可欠です。

## ③ 設計者から作業員まで同一の認識を持つ

整理した階層構造を継続的に活用していく上での最大の敵は、最下層（≒実作業）の情報だけを注視して個別最適化してしまい、上位と下位の階層の整合性が損なわれることです。実態が構造から逸脱しないようにするためにも、階層構造の設計者だけでなく、最下層の実業務を遂行する人も、①②で挙げたような考え方を理解し、行動することが重要であることは言うまでもありません。

## 4. おわりに（業務の見直しへの利用について）

階層構造は各業務プロセスと対比することによって、実施頻度の高い標準的な業務と、特定の業務プロセスでしか発生しない個別最適化された業務を明らかにします。下図のように下層の各要素に対し、自社の全ての業務プロセスをプロットすることにより、一つの行にプロット数が多いほど複数の業務パターンにわたって共通的に発生している業務であると判断できます（色が濃いほど実施頻度が高い業務）。

階層1	階層2	階層3	プロセス1	プロセス2	プロセス3	
1.	販売	1-1. 受注	1-1-1. 受注	○	○	○
		1-2. 在庫確認	1-2-1. 在庫確認	○	○	○
2.	製造	2-1. 生産計画	2-1-1. 生産計画		○	○
		2-2. 生産	2-2-1. 生産		○	○
3.	物流	3-1. 出荷	3-1-1. 工場から出荷		○	○
			3-1-2. 倉庫から出荷	○	○	
	3-2. 納品	3-2-1. 倉庫へ納品		○		
		3-2-2. 客先へ納品	○	○	○	

この分析は階層を増やしてより細かく個別運用の要素を下層に切り出していき、業務プロセスをより細分化（個人や取引単位での実業務レベル）して並べることで、具体的な属人的業務の抽出も可能です。

エンジニアリング（開発）の考え方に馴染みのない方も多いかとは思いますが、本稿を通して興味を抱いていただき、今後の活動にお役立ていただける機会があれば幸いです。