

## ソフトウェア品質展開の枠組み設計法†

大 森 晃††

品質展開をどう適用するかの基本的枠組みが、非ソフトウェア分野だけでなく、ソフトウェア分野でも設計・提案されている。枠組み設計の結果を提示し議論することは重要である。しかし一方、品質展開の基本的枠組みをいかに設計するかを議論することも極めて重要である。ある特定クラスのソフトウェアに関する品質展開の枠組みを設計するに際して、設計の助けとなるガイドラインないしは方法論があれば、状況に応じて柔軟に枠組み設計を行うことができる。本論文は、ビジネス・ソフトウェアに焦点をあてて、品質展開の基本的枠組みをどのように設計するかを議論し、設計のガイドラインを提案する。設計において重要な点は、品質展開の基本的特徴を認識すること、設計の基本的立場を明確にすること、および概念設計と方法論設計を行うことである。概念設計と方法論設計については、設計フェーズを与え、設計の方法論を提案する。

### 1. はじめに

製品品質に関する顧客満足に接近するために、機械工業<sup>1)</sup>、素材産業<sup>2)</sup>、建設業<sup>3)</sup>、サービス産業<sup>4)</sup>のような非ソフトウェア分野において品質展開が幅広く実施されてきた。また、製品品質や品質管理に関して様々な効果も報告されている。

「品質展開」という用語にかわり「品質機能展開」という用語がしばしば用いられているが、これまで品質機能展開と称して行われてきたことは、通常ここで議論する品質展開、あるいは少なくともその一部の実施である。大雑把に言えば、品質機能展開は全社的品質保証活動の全体的枠組みの確立とその実施を意味する。そうした枠組みは、例えば「品質保証体系図」<sup>5)</sup>や「品質保証活動一覧表」<sup>6)</sup>などによって整備される。品質機能展開ではこれら体系図や一覧表において品質展開に関連する活動やツール（品質表など）も記載されることから、品質機能展開は品質展開を包含し品質展開よりも幅広い活動を伴う<sup>7)</sup>。

品質展開はテストやレビューのようなアプローチとは基本的に異なる。それらは生産物における誤りや欠陥を除去するものであり、顧客要求品質を満足するような新しい製品を開発するには自ずと限界がある。通常、品質展開ではある特定製品に対する顧客要求品質の言明を出発点とし、顧客要求品質を満たすために品質情報を軸とする様々なマトリックス図（対応表）を作成する。

品質展開は、マトリックス図を用いて情報間の相互参照も行うことから、ソフトウェア工学で利用される

クロス・レファレンス<sup>8),9)</sup>に似ているように見える。しかし、それは上述のように品質情報を軸とするマトリックス図の作成を伴う。ある特定製品の品質が何であるかを明確にするために品質展開を行うのであり、品質情報を軸とする様々なマトリックス図を作成するのである。

ソフトウェア分野においても徐々に品質展開の適用・検討が試みられている<sup>10)-15)</sup>。適用例の多くは顧客要求品質とソフトウェア品質要素との対応関係を示す「品質表」<sup>16)</sup>の開発に焦点をあてている。なかには一種の「QC 工程表」<sup>17)</sup>の開発を試みているもの、機能・モジュール間のマトリックス図を作成しているものもある。

品質展開をどう適用するかの基本的枠組みは、非ソフトウェア分野ではとりわけ機械工業において設計・提案されている<sup>18),19)</sup>。その枠組みは、非ソフトウェア分野のみならずソフトウェア分野にいたるまで、様々な製品分野においてひとつの手本としてこれまで利用されてきた。一方ソフトウェア分野においては、業務システムにおける業務機能の遂行を支援するソフトウェアに焦点をあてた基本的枠組みが設計・提案されている<sup>20)-22)</sup>。ほかに、対象とするソフトウェアのクラスは定かでないが、基本的枠組みの提案がある<sup>23),24)</sup>。それらは、あくまでも品質展開の基本的枠組みのインスタンス（一例）を与えるものであり、設計の結果を示すものである。

既に存在する基本的枠組みのインスタンスは、利用面から言えば、品質展開を適用する上で参考になるであろう。しかし、それに固執するのは必ずしも得策ではない。固執するとすれば、適用対象とするソフトウェアや適用状況などによっては、品質展開を効果的に適用できなくなる危険がある。また、そのインスタ

† How to Design a Framework of Quality Deployment for Software by AKIRA OHMORI (Fujitsu Laboratories, IIAS).

†† (株)富士通研究所国際情報社会科学研究所



第4はマトリックス系統図法の適用である。この図法では、共通タイプの情報を有するマトリックス図が連鎖を形成するように、諸種のマトリックス図を系統的に連結配置する。図1にマトリックス系統図を図解してある。図1から分かるように、それはA-Bマトリックス図、A-Cマトリックス図、D-Bマトリックス図のような諸種のマトリックス図から成る。アルファベットは情報のタイプあるいは情報そのものを意味する。例えば、A-Bマトリックス図は共通タイプの情報Aを有するA-Cマトリックス図とA-Fマトリックス図に連結され、それらはA系統のマトリックス図連鎖を形成している。また、A-Bマトリックス図はD-Bマトリックス図に連結され、それらはB系統のマトリックス図連鎖を形成している。

第5は品質による管理である。品質展開では、顧客要求品質や製品品質要素などの品質の狙いを、マトリックス図を利用して、製品機能や製品構成要素などの設計対象に結びつける。これにより、技術者に品質意識を与え、技術者の思考および行動に影響を与え、設計段階における品質の作り込みを動機づける。このように、設計過程を品質（狙い）によって管理する。

第6は要因による管理である。品質展開では、品質の狙いを人・機械・材料・方法などの製造工程要因に結びつけ、製造工程における中間生産物や最終生産物の品質をそれらの要因を管理することによって確保しようとする。

### 3. 枠組み設計の基本的立場

品質展開の枠組みを設計する上で、品質展開の基本的特徴を認識しておくことは重要であるが、それだけでは不十分である。設計問題にしばしばつきまとうことであるが、設計の基本的方針なり基本的立場なりを明確にしておく必要がある。以下に、幾つかの基本的立場について述べる。

基本的立場1：前章で述べたような基本的特徴のすべてを枠組み設計に取り入れるのは望ましいことである。しかし、ソフトウェア開発工程は多くの心的あるいは知的作業を伴い、管理あるいはチェックすべき工程要因を列挙するのは容易ではない。このことは、ソフトウェア分野においては要因による管理が容易でないことを意味する。機械工業における品質展開ですら、製品の企画・設計工程における工程要因は扱ってはいない。したがって、枠組み設計においては要因に

よる管理は考慮しないものとする。

基本的立場2：ビジネス・ソフトウェアは業務システムのサブシステムと見なせる。また、それは業務システムの質を向上させるための手段とも見なせる。そのため、ソフトウェアの価値は業務システムの価値に照らして判断されるべきものである。よって、ソフトウェアはその周辺システムである業務システムとともに考えられるべきである。

基本的立場3：開発されようとするソフトウェアのイメージが何らないとするなら、ソフトウェアへの要求品質を収集することは非常に困難である。ソフトウェアの基本機能、つまり業務機能の遂行を支援することに密接に関連し取り除くと最終的なソフトウェアが顧客にとって価値をほとんど持たなくなるような機能を、明確にすることはソフトウェアのイメージを得る上で助けとなる。したがって、ソフトウェアの基本機能が要求品質に先んじて明らかにされるべきである。

### 4. 設計フェーズ

品質展開の枠組み設計に関する問題は、大雑把に言って2つの側面を持つ。ひとつは、どんなマトリックス系統図を作成するかである。もうひとつは、マトリックス系統図ないしはそこに含まれる諸種のマトリックス図をどのように利用するかである。前者は概念設計であり、後者は方法論設計である。

枠組み設計の基本的立場を明確にした後、それを考慮しながら、概念設計と方法論設計は基本的には以下に述べるようなフェーズを実行することによって行うことができる。ただし、以下のフェーズは何をすべきかを提示しているに過ぎず、必ずしも逐次的な手順を与えるものではない。通常、それらは相互に影響しあって実行される。

なお、4.1節と4.2節は概念設計に対応し、4.3節は概念設計と方法論設計の中間に位置し、4.4節と4.5節は方法論設計に対応する。

#### 4.1 マトリックス図の吟味と選択

品質展開の枠組みを構成する基本単位はマトリックス図であるから、まずマトリックス図を吟味し選択する必要がある。マトリックス図は2つのタイプの情報およびそれらの間の対応関係を持つ。したがって、ここでは基本的に情報のタイプと対応関係のタイプを決めることになる。

通常、顧客要求品質とか製品品質要素のような品質

情報のタイプ	業務システム 要求品質(BSQR)	業務システム 品質要素(BSQE)	業務システム 機能(BST)	ソフトウェア 要求品質(SQR)	ソフトウェア 品質要素(SQE)	ソフトウェア基本 機能(SBF)	ソフトウェア 付加機能(SAF)	ソフトウェア サブシステム(SSS)	データ ファイル(DTF)
業務システム 要求品質(BSQR)		寄与							
業務システム 品質要素(BSQE)				寄与	寄与	寄与	寄与		
業務システム 機能(BST)						支援			
ソフトウェア 要求品質(SQR)					寄与				
ソフトウェア 品質要素(SQE)									
ソフトウェア基本 機能(SBF)						重要性			
ソフトウェア 付加機能(SAF)						重要性			
ソフトウェア サブシステム(SSS)						重要性			
データ ファイル(DTF)						重要性		更新/ 参照	

英字略称

BSQR : Business System Quality Requirements  
 BST : Business System Tasks  
 SQE : Software Quality Elements  
 SAF : Software Additional Functions  
 DTF : Data Files

BSQE : Business System Quality Elements  
 SQR : Software Quality Requirements  
 SBF : Software Basic Functions  
 SSS : Software Subsystems

図 2 マトリックス・マトリックス図  
 Fig. 2 Matrix-matrix diagram.

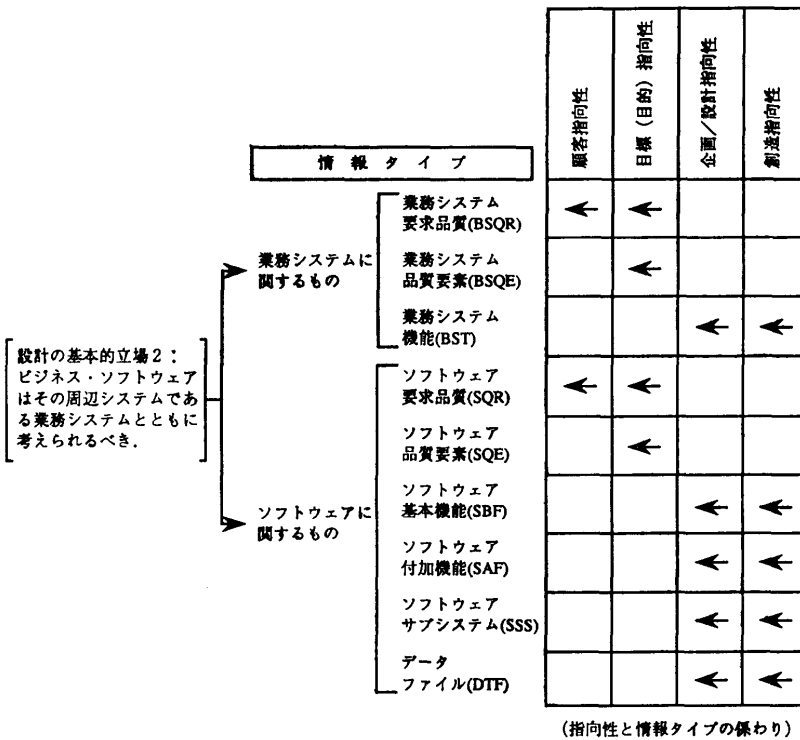


図 3 情報タイプの列挙  
 Fig. 3 Listing information types.

情報を軸とするマトリックス図に焦点をあてる。もちろん、ソフトウェアの機能や構成要素のような設計対象を含むマトリックス図も扱うことは言うまでもない。

マトリックス図を吟味・選択する際、マトリックス・マトリックス図が助けとなる。マトリックス・マトリックス図では、縦軸と横軸にマトリックス図を構成する情報のタイプを列挙する。それらが交差する部分はマトリックス図の候補となる。図2はマトリックス・マトリックス図の一例を示している。

4.1.1 情報タイプの列挙

図2における情報のタイプは、品質展開の基本的特徴である顧客指向性、目標(目的)指向性、企画/設計指向性、創造指向性および設計の基本的立場2を主に考慮して列挙されている。これらの指向性および基本的立場2と列挙された情報タイプとの主たる関連を図3に示す。

設計の基本的立場2として、ビジネス・ソフトウェアはその周辺システムである業務システムとともに考えられるべきであるとした。そのため、図3にはソフトウェアに関係するものだけでなく業務システムに関係するものが含まれている。

図3において、「～要求品質」は顧客指向性に、「～要求品質」と「～品質要素」は目標(目的)指向性に係わるものとして列挙されている。また、「～機能」など業務システムないしはソフトウェアにおける設計対象は、企画/設計指向性に係わるとともに、創造指向性にも係わるものとして列挙されている。ここで、図中の「ソフトウェア付加機能」というのは、ソフトウェア基本機能を強化し、最終的な

ソフトウェアにより高い価値を付与する機能である。

なお、図3には業務システムに関係し顧客指向性や目標(目的)指向性に係わるものとして「業務システム要求品質」(BSQR)とか「業務システム品質要素」(BSQE)が含まれている。このことは、枠組み設計に顧客指向性をより高度に反映させることになる。なぜなら、顧客にとっての目的は業務システムにあり、それに関する要求品質なり品質要素を考えることは、単に開発者側製品であるソフトウェアのみに焦点をあてるよりは、顧客の価値観に近くなるからである。

#### 4.1.2 マトリックス図の選択

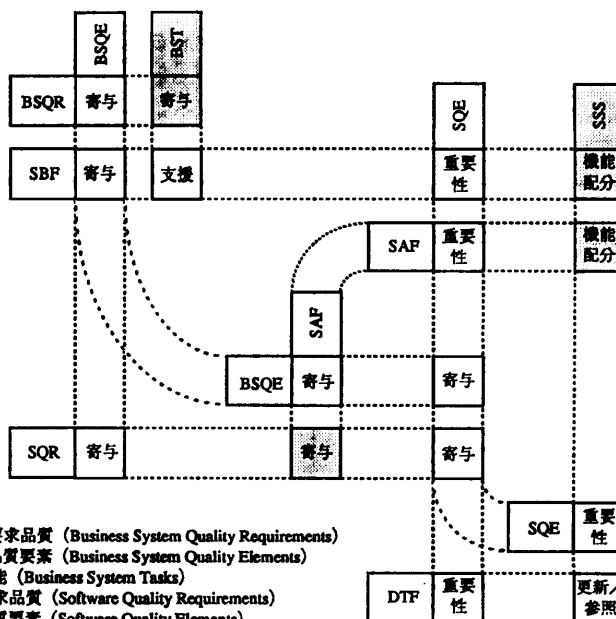
マトリックス・マトリックス

図を利用して必要と考えられるマトリックス図を吟味・選択するのであるが、選択したマトリックス図は四角形で明示する。また、対応関係のタイプは四角形内に表示する。図2はこれらの一例も示している。

ここでは、品質展開の基本的特徴である顧客指向性、目標(目的)指向性、創造指向性および品質による管理を主に考慮してマトリックス図を選択してある。顧客指向性、目標(目的)指向性および創造指向性に係わるものについては、上述の情報タイプの列挙に関する記述から分かるであろう。品質による管理に係わるものは、SBF-SQE マトリックス図のような、ソフトウェアの設計対象と品質要素を伴うマトリックス図である。SSS-DTF マトリックス図はクロス・レフェレンスの必要性を考えて選択してある。

対応関係のタイプは、マトリックス図を構成する2つのタイプの情報間に目的-手段の関係があれば「寄与」、被支援-支援の関係があれば「支援」というように情報間の関係に着目して決めてある。また、品質による管理に係わるマトリックス図については、品質の狙いを重要性の観点から設計対象に結びつけるのが自然であり、対応関係のタイプとしては「重要性」を指定している。

図2において、「寄与」は横軸のものの達成ないし



英字略称

BSQR:業務システム要求品質 (Business System Quality Requirements)  
 BSQE:業務システム品質要素 (Business System Quality Elements)  
 BST:業務システム機能 (Business System Tasks)  
 SQR:ソフトウェア要求品質 (Software Quality Requirements)  
 SQE:ソフトウェア品質要素 (Software Quality Elements)  
 SBF:ソフトウェア基本機能 (Software Basic Functions)  
 SAF:ソフトウェア付加機能 (Software Additional Functions)  
 SSS:ソフトウェアサブシステム (Software Subsystems)  
 DTF:データファイル (Data Files)

図4 マトリックス系統図の作成

Fig. 4 Developing a matrix sequence diagram.

は改善が縦軸のものの達成ないしは改善に寄与する関係を意味する。「支援」は横軸のものが縦軸のものを支援する関係を意味する。「重要性」は縦軸のものの実現において横軸のものが重要かどうかの関係を意味する。「更新/参照」は横軸のものが縦軸のものを更新したり参照したりする関係を意味する。

#### 4.2 マトリックス系統図の作成

マトリックス系統図法を適用して、選択したマトリックス図を系統的に連結配置する。一例を図4に示してある。英字略称は図2のものに対応している。

図4では、例えば、BSQE 系統のマトリックス図が縦方向と横方向の両方にある。これは同方向の系統を形成する情報、例えば縦方向の BSQE と SAF、を連結配置するための処置である。

一般に、必要なマトリックス図のすべてが最初から選択されるわけではない。マトリックス系統図を作成することは他の有用なマトリックス図を見いだすきっかけにもなる。図4における網目状のマトリックス図はそうしたマトリックス図の一例を示している。それらは先に選択したマトリックス図に追加される。ここで、対応関係のタイプ「機能配分」はソフトウェアのどの機能をどのサブシステムで受け持つかの関係を意味する。

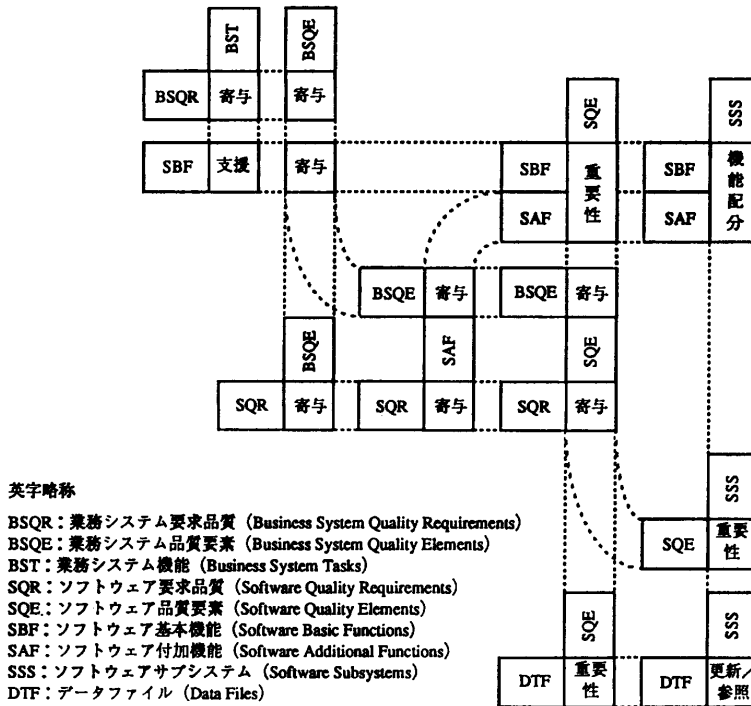


図5 マトリックス系統図の洗練  
Fig. 5 Refining the matrix sequence diagram.

4.3 マトリックス系統図の洗練

幾つかのマトリックス図を適切な場所に移動させながら、作成したマトリックス系統図を洗練していく。これは諸種のマトリックス図をどうい手順で作成していくかを設定することにも関連した作業である。そのため、マトリックス系統図の洗練はマトリックス作成手順をある程度意識して行うのがよい。このように、このフェーズは概念設計と方法論設計の中間に位置する。しかし、このフェーズはマトリックス図の空間的な配置と空間的な関係を洗練するのであり、一連のマトリックス図をどうい手順で作成するかを設定するものではない。

洗練されたマトリックス系統図の一例を図5に示す。英字略称は図2のものに対応している。図4と比べると、図5では縦方向のBSQE系統のマトリックス図配置がBST系統のマトリックス図配置と置き換えられている。また、図4におけるSBF-SQEマトリックス図およびSAF-SQEマトリックス図が、図5ではひとつのマトリックス図にまとめられている。SSS-SBFマトリックス図とSSS-SAFマトリックス図も同様である。

4.4 マトリックス図作成手順の設定

マトリックス図の作成は、極めて創造的な作業を伴い、それを構成する2つのタイプの情報を言明すること、およびそれら情報間の対応関係を規定することから成る。通常、情報の言明は情報を階層的に表現した「展開表」の形をとり、親和図法や系統図法の適用により行われる。一連のマトリックス図に対する作成手順の設定は、マトリックス系統図における情報の言明と対応関係の規定に関する部分に、順序を示す連番を

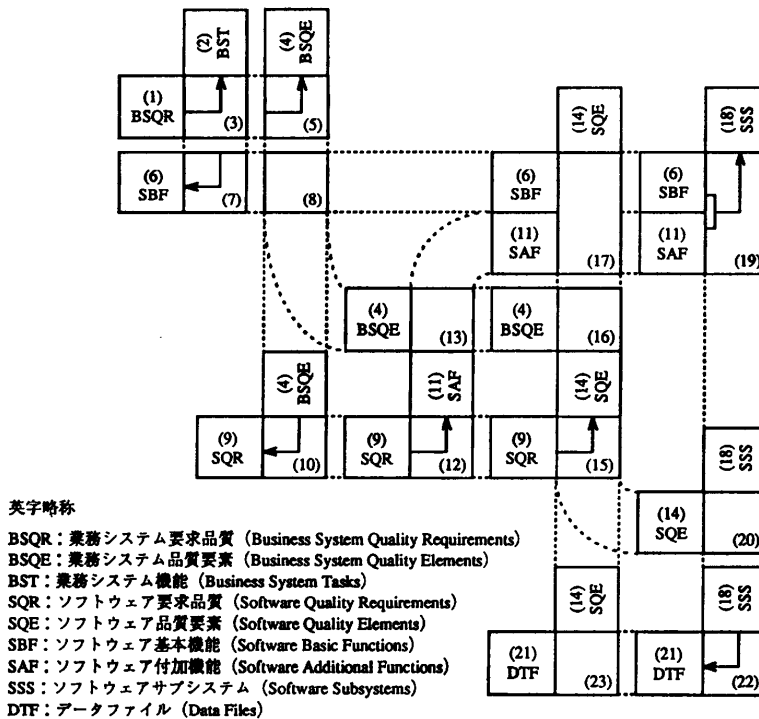


図6 マトリックス図作成手順の設定と重要度変換  
Fig. 6 Establishing a procedural order for developing relationship matrices, and priority propagation.

割り当てることにより行う。

図6に作成手順の一例を示してある。英字略称は図2のものに対応している。同じ番号を持つ部分は同じ情報を含む。設計の基本的立場<sup>3</sup>として、ソフトウェアの基本機能(SBF)が要求品質(SQR)に先んじて明らかにされるべきであるとした。図6ではそうした手順になっている。なお、図6では矢印が明記されているが、これは重要度変換の流れを示すものであり次節で述べる。

手順を設定する際に参考にしうる点が幾つかある。情報の言明に関しては参照関係である。つまり、ある情報の言明を主にどの情報を参照して行うかである。参照関係を明らかにし、被参照情報を先に言明していくようにすればよい。

対応関係の規定に関しては、どの時点でそれが可能かである。基本的には、可能な時点で関係を随時規定していくようにすればよい。ただし、ここでも参照関係が参考になりうる。例えば、図6におけるBST-SBFマトリックス図とBSQE-SBFマトリックス図は、SBFの言明の後、そこにおける対応関係を基本的には同時に規定することが可能である。このような場合、参照関係をもつ情報(被参照-参照の関係をもつ情報)の間の対応関係を先に規定するのが、流れからすれば自然である。

その他、開発における技術的活動との調和とか、マトリックス図の利用法などが手順設定に係わってくる。上述のことは、あくまで手順設定において参考にしうる点を述べたものであり、手順設定の完全な基準を与えるものではない。

#### 4.5 マトリックス図利用法の検討と整備

マトリックス図作成手順の設定は、マトリックス系統図を全体としてどういう手順(流れ)で利用するかを定めている。また、4.1節での議論から分かるように、各マトリックス図が品質展開の基本的特徴、例えば顧客指向性、目標指向性、企画/設計指向性、創造指向性および品質による管理、とどう係わるかは明らかである。さらに、各マトリックス図における対応関係のタイプも明らかになっている。これらのことは、各マトリックス図がなんのために必要で、それをどのように利用するかについて、ある程度の示唆を与える。

しかし、ここまでのフェーズは、一連のマトリックス図の具体的な利用法を提示しているわけではない。一連のマトリックス図の様々な利用法をいかに検討・

整備していくかの一般論を述べることは非常に困難である。しかしながら、以下のことには言及できる。つまり、一連のマトリックス図に関して「なんのために」と「どのように」を注意深く考え、顧客満足により接近するための助けとなる利用法を検討・整備していくことが重要である。その際、すでに提案されている利用法は参考になるであろう。

以下では、品質展開の基本的特徴でありこれまで設計フェーズに絡めて言及しなかった重点指向性に係わるものとして、品質展開の主要な一側面でもある重要度変換のための利用法を述べるにとどめる。図6におけるBSQR(業務システム要求品質)に対して、重要度が与えられるものとする。これをどのように与えるかもひとつの問題ではあるが、ここで扱う問題はその重要度をどのマトリックス図を利用して、どのように他のタイプの情報の重要度に変換していくかである。

利用するマトリックス図は、結論から言えば、図6において矢印が明記されているものである。変換は矢印の起点から終点に向けて、対応関係を利用して行う。

重要度変換に用いるマトリックス図の組みとしてはこれ以外にもいろいろ考えられる。どんな組みを使えばよいかを決めるための確固たる基準はないが、ここでは情報の言明に関する参照関係をもとにして、利用するマトリックス図を決めてある。つまり、タイプBの情報の言明を主にタイプAの情報を参照して行うとすれば、A情報に対する重要度を変換元(矢印の起点)とし、B情報に対するものを変換先(矢印の終点)としている。変換は、双方の間の対応関係を利用して、例えば次のように行う。

$$B(j) = T(\sum_i A(i) \times W(i, j)).$$

ここで、 $A(i)$ は変換元の第*i*項目の重要度、 $B(j)$ は変換先の第*j*項目の重要度、 $W(i, j)$ は第*i*項目と第*j*項目の対応関係の程度を数値化したもの、 $T(\cdot)$ は単調増加関数である。

なお、図6に示されている基本的枠組みはすでに提案されている枠組み<sup>21)</sup>に非常に類似したものとなっている。そこでは「バイニング・ポイント」や「設計ポイント」の分析への利用法や「適合性分析」への利用法が提案されており、参考にされたい。

また、そこでは具体例とともに枠組みの適用指針が与えられている。それは、ここで一例として設計された枠組みの適用法を考える際に参考となろう。

## 5. おわりに

品質展開は、品質に焦点をあてて顧客満足により接近しようとするアプローチである。この意味で、品質展開は品質重視かつ顧客指向のソフトウェア・プロセスを産みだすための有力なアプローチと言える。

そうしたソフトウェア・プロセスに向かうには、いずれにせよ品質展開の基本的枠組みを設計し、それをソフトウェア開発活動に取り入れる必要がある。本論文は、ビジネス・ソフトウェアに焦点をあてて枠組み設計のガイドラインを提案した。枠組み設計において重要な点は以下のとおりである。

- 1) 品質展開の基本的特徴を認識すること。
- 2) 設計の基本方針なり基本的立場を明確にすること。
- 3) 品質情報を軸とするものに焦点をあてて必要なマトリックス図を選択すること。
- 4) マトリックス系統図を作成・洗練すること。
- 5) 一連のマトリックス図を作成していく手順を設定すること。
- 6) 一連のマトリックス図の利用を整備すること。

提案したガイドライン、とりわけ設計フェーズを実行するための方法は設計方法論の部類に属するもので、そこには一般論として明確に論じきれない点、不十分な点がまだある。また、それはビジネス・ソフトウェアに限定したものとなっている。しかしながら、本論文で提案したガイドラインは、状況に応じて柔軟に枠組み設計をする上で参考になるはずであり、他のソフトウェア・クラスに対しても示唆を与えるものと考えられる。

本文では言及しなかったが、設計した基本的枠組みの実適用に先立ち、その実行可能性を小さな例題を対象にした仮適用により確認することが必要であろう。また、こうした仮適用の前に、設計した枠組みが妥当かどうかを吟味しておくことも必要であろう。

枠組みの設計は、基本的には、対象とするソフトウェア・クラスや適用状況、設計の基本的方針なり基本的立場などによって誘導される。なぜならば、それらは枠組み設計に対し要求として作用するからである。したがって、設計した枠組みがそれらに関して適当であると認められるかどうかによって、その枠組みの妥当性を吟味していくことになる。例えば、本論文で一例として設計した枠組みと第3章で与えた設計の基本的立場を考えるなら、その枠組みが顧客指向性、目標

(目的)指向性、重点指向性などを達成しうるものになっているかどうかを吟味する。

設計した枠組みは当然ながら一連の設計フェーズを実行した結果であるから、妥当性の吟味はしばしば設計の各フェーズにまで立ち入る。各フェーズの実行において上述したような枠組み設計に対する要求がどこまで考慮されているか、も枠組みの妥当性を吟味する際のポイントになる。

なお、枠組みの設計者と適用者が分離している場合には、設計した枠組みの妥当性の判断が双方の間で異なることがあろう。したがって、例えば企業内において品質管理部門のみにより枠組みを設計しそれを開発部門が適用するというような場合には、開発部門を巻き込んだ吟味が必要となる。妥当性の吟味や仮適用において不都合があれば、枠組みにフィードバックをかけて改良することになる。

## 参考文献

- 1) 高橋征之：小松製作所小山工場における品質機能展開，品質管理学会「品質」誌，Vol. 13, No. 3, pp. 98-106 (1983).
- 2) 山添 宏，市川 宏：炭化けい素繊維（ニカロン）の開発におけるQC手法の活用，品質管理学会「品質」誌，Vol. 16, No. 3, pp. 51-54 (1986).
- 3) 中島 隆，安富重文，西本和彦：“新国技館”の要求品質の実現，品質管理学会「品質」誌，Vol. 15, No. 4, pp. 55-60 (1985).
- 4) 赤尾洋二ほか：航空運輸業における旅客サービスの要求品質展開，品質管理学会「品質」誌，Vol. 14, No. 3, pp. 68-74 (1984).
- 5) Furukawa, O., Ikeshoji, H. and Ishizuchi, H.: A Theoretical Study on Configurations of Quality Control System, *IEEE Trans. Engineering Management*, Vol. EM-29, No. 4, pp. 128-134 (1982).
- 6) Furukawa, O. and Ishizuchi, H.: Refinement of Quality Control System, *Int. J. General Syst.*, Vol. 9, pp. 161-170 (1983).
- 7) 岩橋聡夫：品質機能展開の実施例，品質機能展開（水野，赤尾（編）），pp. 199-224, 日科技連 (1979).
- 8) Pressman, R. S.: *Software Engineering: A Practitioner's Approach*, p. 134, McGraw-Hill (1982).
- 9) Shooman, M. L.: *Software Engineering*, p. 100, McGraw-Hill (1983).
- 10) 相沢 覚ほか：汎用波形解析プログラム(HAKEI)の開発，品質管理11月臨時増刊号，Vol. 33, pp. 360-366 (1982).
- 11) 田上裕二：ソフトウェアへの品質展開の適用，第5回ソフトウェア生産における品質管理シンポ



- ジウム発表報文集, 日科技連, pp. 71-78 (1985).
- 12) 関川八郎: メカトロ製品における要求品質の実現, 品質管理学会「品質」誌, Vol. 16, No. 1, pp. 73-78 (1986).
- 13) 緒方正典, 片山禎昭, 竹内 誠: ソフトウェア開発現場における品質機能展開, 第8回ソフトウェア生産における品質管理シンポジウム発表報文集, pp. 53-60, 日科技連 (1988).
- 14) 情報処理振興事業協会技術センター (編): 品質機能展開による高品質ソフトウェアの開発手法, コンピュータ・エージ社 (1989).
- 15) 吉澤 正: ソフトウェア品質機能展開の発展をめざして, 品質管理学会「品質」誌, Vol. 20, No. 2, pp. 80-85 (1990).
- 16) 高柳 昭: 品質表の概念と生いたち, 品質機能展開 (水野, 赤尾 (編)), pp. 40-52, 日科技連 (1979).
- 17) 石原勝吉: QC 工程表の作り方, 使い方, 品質機能展開 (水野, 赤尾 (編)), pp. 142-160, 日科技連 (1979).
- 18) 赤尾洋二: 品質展開システムの手順, 品質機能展開 (水野, 赤尾 (編)), pp. 175-199, 日科技連 (1979).
- 19) 赤尾洋二ほか: コスト・信頼性・技術を含めた品質展開 (その1) (その2), 品質管理学会「品質」誌, Vol. 13, No. 3, pp. 61-77 (1983).
- 20) 大森 晃: ソフトウェア生産における品質展開, 第8回ソフトウェア生産における品質管理シンポジウム発表報文集, pp. 45-52, 日科技連 (1988).
- 21) 大森 晃: ソフトウェア品質管理への品質展開アプローチ, 情報処理学会論文誌, Vol. 31, No. 10, pp. 1474-1485 (1990).
- 22) Ohmori, A.: A Framework Study of Quality Deployment for Software, *9th Int. Conf. on EDP System and Software Quality Assurance* (1990).
- 23) 小野道照, 大藤 正, 松井忠彦, 赤尾洋二: 品質展開によるソフトウェア開発方針の指針, 品質管理学会第35回研究発表会研究発表要旨集, pp. 73-76 (1989).
- 24) Zultner, R. E.: Software Quality [Function] Deployment—Applying QFD to Software, *13th Rocky Mountain Quality Conference* (1989).
- 25) 水野 滋 (監修), QC手法開発部会 (編): 管理者スタッフの新 QC 七つ道具, 日科技連 (1986).  
(平成4年4月15日受付)  
(平成4年9月10日採録)



大森 晃 (正会員)

1954年生. 1977年広島大学工学部経営工学科卒業. 1985年同大学院工学研究科博士課程後期修了 (システム工学専攻). この間 1982年9月より1年間ケースウェスタンリザーブ大学客員研究員 (米国オハイオ州クリーブランド). 工学博士. 1985年4月富士通 (株) 国際情報社会科学研究所に入社. 1990年12月より (株) 富士通研究所国際情報社会科学研究所. 品質目標追求のシステム方法論, 意思決定過程論, ソフトウェアの品質管理や構成管理などの研究に従事. 日本品質管理学会会員.