

ビジネスの視点による非機能要件の定義フレームワーク

桑原 淳一 柿本 達彦

A Framework for Defining Non-functional Requirements from a Business Standpoint

Junichi Kuwabara and Tatsuhiko Kakimoto

IT システムはビジネス課題を解くための手段であり、解くべき課題であるビジネス要件の定義を行うことは重要である。ビジネス機能要件の抽出手法としてはビジネス・ユースケースなどが活用されているが、ビジネス非機能要件に関しては明確な定義手法や指針が存在しない。本論文は、この課題への解決策として、ビジネス非機能要件を構造化し、ビジネス非機能要件とシステム非機能要件のトレーサビリティを確保するフレームワークを提案する。このフレームワークを実際プロジェクトに適用することにより、その有効性が確認された。

In IT system development, defining business requirements is important, since IT systems are measures to solve business problems. Business functional requirements are obtained by using business use cases and other methods, but there have been no established methods or guidelines for defining business non-functional requirements (BNFR). As a solution to this issue, this paper introduces a new framework for defining BNFR and keeping traceability between BNFR and system non-functional requirements. The effectiveness of this framework has been confirmed in real client engagements.

Key Words & Phrases : 要求工学, ビジネス要件, 非機能要件, システムズ・エンジニアリング
Requirements engineering, Business requirements, Non-functional requirements, Systems engineering

1. はじめに

IT システム開発を成功させる上で、要件を適切に定義することは重要である。Standish Group による 1995 年のレポート [1] では、成功した IT システム開発プロジェクトの主たる成功要因の一つとして「明確に記述された要件」を挙げるとともに、失敗したプロジェクトの原因の筆頭に「不完全な要件」を挙げている。日経コンピュータによる国内の調査 [2] [3] でも、プロジェクトでの品質問題の主要因として「要件定義不十分」が挙げられている。

企業や組織にとって、IT システムの開発・運用は、ビジネス上の目標達成や課題解決のための手段の一部である。従って IT システムの要件を定義するには、もともとのビジネス上の要求を明確にすることが重要な鍵となる。このような考え方に基いて、IBM が携わるシステム開発プロジェクトにおいても、SE&A (Systems Engineering & Architecture) の手法にのっとりつつ、システム要件 (System Requirements, 以下 SR と呼ぶ) を定義する前にまずビジネス要件 (Business Requirements, 以下 BR) の定義を行うケースが増えている。BR 定義には、ビジネス・ゴール達成に必要なビジネス・オペレーションを抽出し、ビジネス・ユースケース (以下 BUC)、ビジネス・プロセスを定義するという手法が用いられる [4]。

BR のうち、機能要件 (Functional Requirements, 以下 FR) はビジネス・プロセス・モデリングなどを通じて定義すべき対象をとらえやすいのに対し、非機能要件 (Non-functional Requirements, 以下 NFR) は、定義項目の明確な指針が見当たらない。NFR についてはソフトウェア品質特性やソフトウェア要求仕様の一部として文献 [5] [6] で標準化されているほか、IBM のグローバル共通の方法論である IBM Unified Method Framework では NFR として定義すべき項目がガイドされている。また 2008 年には日本情報システム・ユーザー協会 (JUAS) による NFR 仕様のガイドライン [7] が策定されている。しかしながら、これらの標準やガイドラインは IT システム視点の NFR を対象としており、ビジネス視点での NFR については明確に規定していない。また NFR に関しては、文献 [8] でアーキテクチャーへの反映と検証の方法が、文献 [9] でインフラ設計での必要項目がそれぞれ提示されているほか、ゴール指向分析で NFR を可視化するフレームワーク [10] なども提唱されているが、ビジネス視点での NFR をどう定義するかは、プロジェクトの現場担当者の判断に委ねられているのが実情である。このような現状の結果として、例えばビジネス機能要件 (Business FR, 以下 BFR) の定義作業とはまったく無関係に現行 IT システムの実績値のみから NFR を定義するなど、ビジネス上の要求を反映しない形で NFR を定義してしまうケースがしばしば見受けられる。このような事態を防止するには、以下の 2 つの課題を解決する必要がある。

提出日:2008年9月9日 再提出日:2008年12月5日

【課題1】ビジネス非機能要件 (Business NFR, 以下 BNFR) としての定義項目を明確化する。

【課題2】定義したBNFRを、どのようにSRに反映させるのかを明らかにする。

本論文は、上記課題を解決するための「BNFRを定義するためのフレームワーク」を新たに提案し、その有効性を論ずることを目的とする。まず2章にてBNFRの重要性について述べ、続く3章において、課題1の解決策としてBNFRフレームワークを提案し、さらに4章で、課題2の解決策としてBNFRとシステム非機能要件 (System NFR, 以下 SNFR) との関係性を述べる。5章ではプロジェクト事例を基にフレームワークの有効性を示す。最後に6章で全体を総括する。

2. BNFR定義の重要性

本章では、BRのとらえ方と、BNFRがBFRに対して持つ意味を整理し、BNFRを定義することの重要性について述べる。

2.1 BRの位置付けおよびBFR・BNFRの関係

要件を定義するには、その議論の対象となるシステムのスコープ (System under Discussion, 以下 SuD [11]) を明確にする必要がある。システムズ・エンジニアリングをリードしている国際的な組織である INCOSE は、システムを「定義された目的を成し遂げるための、相互に作用する要素を組み合わせたもの」と定義しており [12] [13], 必ずしも IT システムに限定していない。システムをこのようにとらえると、BRとSRとは要件を割り当てるスコープのみが異なるものといえる。すなわち、図1に示すように、SR定義の際のSuDがITシステムであるのに対し、BR定義のSuDはビジネス・システムである。ビジネス・システムはその内部で活動する人やITシステムなどを内包し、外部のアクターに対して

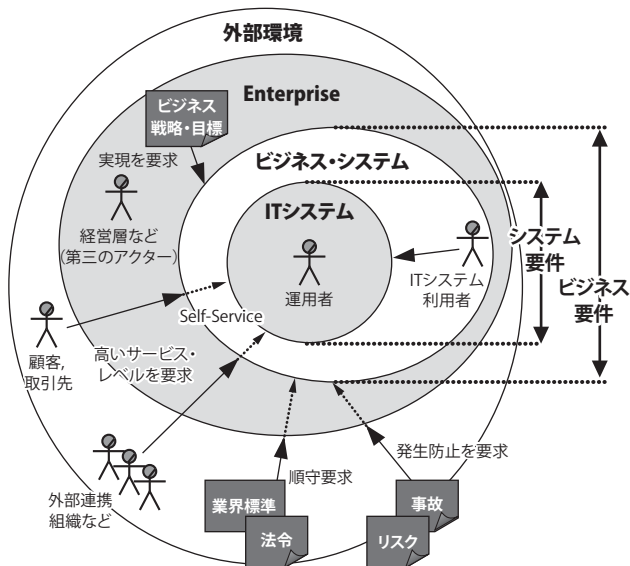


図1. BRおよびSRのスコープ

価値を提供する。BRとはこのビジネス・システムが満たすべき要件をビジネスの用語で表したものであり、そのBRをさらに分析することにより、ITシステムで解決すべき要素として識別された要件がSRである。

BR、すなわちビジネス・システムの要件は、ITシステムの要件と同様、FRとNFRに分類できる。ここでFRとは、ビジネス・ゴール達成やビジネス・リスク回避に必要な手段のことであり、NFRとは、与えられた制約条件の中でFRが満足すべき品質を指す [14]。品質属性には、ビジネス・プロセスの実行頻度や、ビジネス・プロセス完了までの時間などがある。FRとその品質属性要件を定義することは、バランスト・スコアカードにおいてCSF (Critical Success Factor: 主要成功要因) とKPI (Key Performance Indicator: 重要業績評価指標) とを定義することに相当する [15]。

以上を踏まえると、BNFRとして定義すべき対象は、BFRの品質属性およびビジネス制約であると考えられる。従って、BFRとBNFRの間には以下の関係が存在する (図2)。

- (1) BFRは、BNFRのうち「ビジネス制約」からの制約を受ける。
- (2) (1)の状況下でBFRが満たすべき品質要件が、BNFRのうち「品質属性」として定義される。
- (3) BNFRでの要求品質が実現不可能な場合は、品質属性の見直しやBFR・BNFR間での調整が必要となる。

2.2 BNFRを定義すべき理由

実際のプロジェクトにおいては、BRを定義するフェーズではビジネス・プロセス・モデリングとBFR定義にのみ関心が集中し、BNFR定義がおろそかになる傾向が見受けられる。しかし、筆者は以下の理由により、BNFRをBFRと合わせて定義することが重要であると考えられる。

(1) BR実現によるビジネスへの貢献の観点

• BNFRを定義することで、機能だけでなく品質・制約面を含めたビジネス上の要求を施策 (ITシステム開発など) に反映させることができる。BNFRを定義しない場合、施策を実施しても、ビジネス・ゴールの達成に寄与しない可能性がある。

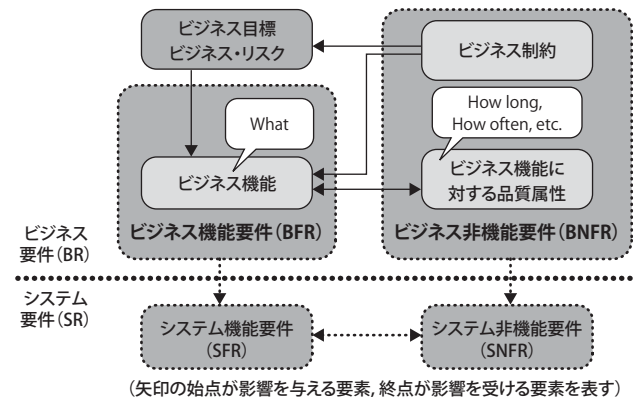


図2. BFRとBNFRとの関係

- BNFR は施策のビジネスへの貢献度を評価する指標となる。施策実施後に、この指標値をフィードバックすることで、次なるビジネス改善策の立案が可能になる。

(2) BFRとしての品質・妥当性の観点

- BNFR を定義することにより、ビジネス上求められる品質属性を踏まえた BFR を定義できる。例えば品質属性として所要時間要件が厳しいケースにおいて、その要件を考慮に入れた上でビジネス・プロセスをモデリングし、BFR に反映することができる。
- BNFR を定義することにより、ビジネス制約を考慮した上で BFR を定義できる。すなわち、定義された BFR が法令などに抵触せず、制約環境下で実現可能なものであることを担保できる。

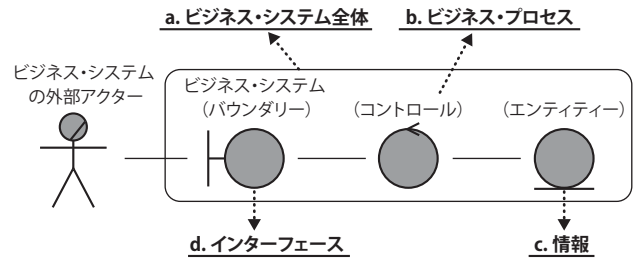


図4. BNFRをとらえる4つの視点

からのアナロジーに基づいている。さらにそのような個別要素に属さない品質属性や制約を a の視点で把握する(図 4)。このようにビジネス・システムを構造的にとらえることにより、フレームワークとしての網羅性を確保している。

3. BNFRフレームワーク(課題1の解決策)

本章では、BNFR を定義するためのフレームワークを提示し、フレームワークに含まれる各項目と、その定義単位の考え方を述べる。

3.1 フレームワークの提案

図 3 に、筆者の提案する BNFR のフレームワークを示す。このフレームワークが、【課題 1】への解決策に相当する。以下、BNFR 項目をとらえる視点と各定義項目について解説する。

(1) BNFRをとらえる4つの視点

本フレームワークでは、ビジネス・システムの NFR を以下の a ~ d の 4 つの視点でとらえている。

- ビジネス・システム全体
 - ビジネス・プロセス
 - ビジネス・プロセスを支える情報
 - ビジネス・アクターとのインターフェース
- このうち b ~ d については IT システムの「分析モデル」

(2) 定義項目

以下、フレームワークに含まれる 13 個の BNFR 定義項目について、それぞれ解説する。

[a. ビジネス・システム全体の品質属性・制約]

1. ビジネス基礎数値

ビジネス・システムが持つ基礎的な数値属性で、ビジネス・システムの KPI に直結するものや、処理量やアクター数(後述)の定義の際の基礎情報となるもの。ビジネスの拡張目標を踏まえた将来の伸び率や目標数値も合わせて定義する。定義例としては、総顧客数、提携先数、社員数、受注総数、決済件数など。

2. ビジネス制約

ビジネス・システムがビジネスを遂行するにあたって受ける制約を指す。具体的には、法令・業界ガイドライン・社内規定、あるいはビジネス環境(ロケーションや組織なども含む)などが例として考えられる。ビジネス制約は、ほかの BNFR 定義項目を含む BR 全体に影響を及ぼす。

3. ビジネス柔軟性

変化が予想されるビジネス要素と、ビジネス・システムとしての対応要件を定義する。変化が予想されるビジネス要素の例としては、取扱商品、対象顧客、チャンネル、提携先、法令、ビジネス・ルールなどが考えられる。

[b. ビジネス・プロセスに関する品質属性・制約]

4. 処理時間

ビジネス・プロセスの全体またはその一部分の所要時間や処理期限。例えば、顧客による購入申し込みから顧客に商品が届くまでの時間などを指す。

5. 処理量

ビジネス・プロセスが実行される回数(頻度)と、一回のプロセスで処理される件数。例えば一日当たりの商品購入申し込み件数が前者、一回の購入申し込みに含まれる商品の件数が後者に相当する。平均処理量だけでなく、ピーク日や時間帯とその処理量も定義する。

6. 可用性

ビジネス・プロセスの可用性(Availability)として、以下の3項目を定義する。

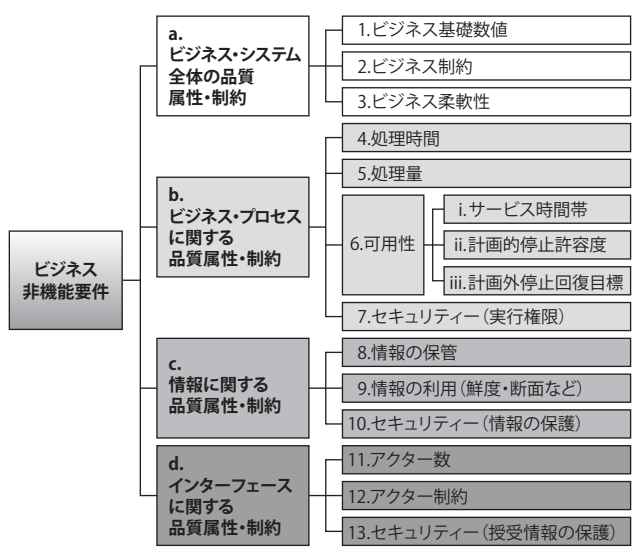


図3. BNFR定義フレームワーク

- i. サービス時間帯：BUC に要求される業務サービスの時間帯。平日、週末などで異なる場合はそれらすべてを定義する。
- ii. 計画的停止の許容度：ビジネス・プロセスの計画的な停止が許容される頻度と、その停止時間。このような計画的停止が必要になる理由としては、例えばそのオペレーションを支える設備の保守点検などがある。
- iii. 計画外停止時の回復目標：設備の故障や停電・災害など、計画外の要因によってビジネス・プロセス停止が生じた際の回復時間目標（RTO: Recovery Time Objective）と回復時点目標（RPO: Recovery Point Objective）を定義する。

7. セキュリティー(実行権限)

ビジネス・プロセスに含まれるアクティビティーの実行を許可されるアクターやロール。

[c. 情報に関する品質属性・制約]

8. 情報の保管

ビジネス・プロセスが利用する情報に対して、特殊な保管要件がある場合の保管方法と保管期間。主に法令や自主ルールなどに基づく。

9. 情報の利用(鮮度・断面など)

ビジネス・プロセスが情報を利用する際、その情報に求める品質属性。例えば情報の鮮度(1時間前の情報で十分か、最新情報が必須かなど)や有効期限(一定期間経過した情報は無効扱いとするなど)、特定時点断面での情報の要否(毎月末時点の状態など)、過去の処理に対する遡及処理の要否、情報の精度などの視点で、情報の利用要件を定義する。

10. セキュリティー(情報の保護)

個人情報や機密情報へのアクセスが許されるアクター、ロールの特定など、ビジネス・システムで扱う情報の保護要件。

[d. インターフェースに関する品質属性・制約]

11. アクター数

ビジネス・システムの外部アクターと、ビジネス・システムの構成要素である内部アクターの数を、そのロケーションとともに定義する。

12. アクター制約

対象とする外部／内部アクターの持つ制約。言語(日本語、英語、中国語など)、アクセシビリティの考慮、IT操作の熟達度など、主にユーザビリティに関連する項目が考えられる。

13. セキュリティー(授受情報の保護)

アクターとの間でビジネス・システムが授受する情報の保護要件。例えば

はがきの個人情報をシールで隠す、口座番号を記載しない、重要物を書留郵便で送付する、などが挙げられる。

3.2 BNFRの定義単位

表1は、3.1で提案したBNFRフレームワークの各定義項目を、どの単位で定義すべきかを示したものである。単位として挙げているもののうち、「ビジネス・エリア」はBUCを特定の観点(例えば事業部門)でグループ化した単位を表し、「アクティビティー」はBUCを実現するビジネス・プロセスの個々のステップを指す。

例えば、ビジネス基礎数値はビジネス・システム全体に対して定義するのに対し、処理時間はBUC単位で定義するべきである。処理時間に関して、もしBUCの要件とは別に、特定のアクティビティーでの固有要件があれば、それについても定義する。また情報の利用要件は、ビジネス・エンティティーとアクティビティーの組み合わせを単位として定義する。

なお、BNFR定義を進める際の注意事項として、以下の点を挙げておく。

- 各BNFR項目を定義する対象を目的に応じて適切に選択する：表1で定義単位として掲げた領域の全インスタンス(例えばすべてのBUC)に対して要件を定義することが常に必要とは限らない。BUC数やアクティビティー数が多い場合は、そのプロジェクトでBR定義を行う目的と、定義

表1. BNFRを定義する単位

	要件の定義領域の単位						
	ビジネス・システム全体	ビジネス・エリア	ビジネス・ユースケース	アクティビティー	ビジネス・エンティティー	アクター	
a. ビジネス・システム全体の品質属性・制約	1. ビジネス基礎数値	○					
	2. ビジネス制約	○					
	3. ビジネス柔軟性	○					
b. ビジネス・プロセスに関する品質属性・制約	4. 処理時間			○	※		
	5. 処理量			○	※		
	6. 可用性	i. サービス時間帯		○	※		
		ii. 計画的停止許容度		○	※		
		iii. 計画外停止回復目標		○	※		
7. セキュリティー(実行権限)				○			
c. 情報に関する品質属性・制約	8. 情報の保管					○	
	9. 情報の利用(鮮度・断面など)				●	●	
	10. セキュリティー(情報の保護)					○	
d. インターフェースに関する品質属性・制約	11. アクター数					○	
	12. アクター制約					○	
	13. セキュリティー(授受情報の保護)					● ●	

[凡例]

○ = 該当する領域の単位で要件を定義

● = ほかの定義領域との組み合わせの単位で要件を定義

※ = 該当する領域の単位で固有要件がある場合に定義

した BNFR がその目的に寄与する度合いを勘案し、パレートの法則などに基づいて、対象とするインスタンスを取捨選択することを検討する。

- 要件をオペレーション・モードごとに定義する：文献 [4] でも述べられているように、BR 定義の過程においては、ビジネス・システムのライフ・サイクルにおいて想定されるさまざまな状況に対応付けられたオペレーション・モード (“Modes of Operation” [16]) のおのおのに対してビジネス・シナリオを定義するが、オペレーション・モードによって求められる BNFR が異なる場合は、そのモードごとに別々に定義する必要がある。

4. SRへのBNFRの反映(課題2の解決策)

本章では、提案したフレームワークの利用による【課題2】の解決策について述べる。

この解決策が、表2のマトリックスである。これは、BNFRの定義項目と主要なSNFRの定義項目との関係、すなわちBNFRがどのSNFRに影響するかを示したものである。例えばBNFR項目の「処理量」は、SNFRのスループット(ト

ランザクション・レートなど)やキャパシティー(データ容量など)に影響を与える。逆にSNFRとしてデータ容量を定義する際は、BNFRの「ビジネス基礎数値」「処理量」「情報の保管」「情報の利用」が根拠となる。

BNFRからSNFRを導く方法には、BNFRとして定義した値を基にSNFRに変換するケースと、BNFRを参考にしながら別途SNFRとしての定義を行うケースがある。例えばSNFRとして「トランザクション・レート」を定義する場合、①BNFRの「処理量」として定義した「ビジネス・プロセスの実行回数」、②そのプロセスが含むシステム・ユースケース(以下SUC)の数、③SUC実現に必要なシステム・インタラクション回数の3要素の乗算により導出できる。一方、「オンライン応答時間」については、BNFRの「処理時間」「アクター制約」を考慮の上で、人間工学的視点での分析を行い、SNFRとして定義する。

いずれの場合も、BNFRフレームワークに含まれる各項目は、SNFRを定義する際の根拠となるものであり、また両者の関係に従ってBNFRからSNFRを導くことにより、BNFRをITシステム開発に反映させることができる。

表2. BNFRがSNFRに与える影響

	パフォーマンス/キャパシティー							可用性				データ整合性	ユーザビリティ	セキュリティ	変更容易性	拡張性		
	応答時間		スループット			キャパシティー		サービス時間帯	計画停止可能時間	サービス回復時間目標(RTO)	データ回復時点目標(RPO)							
	オンライン応答時間	バッチ処理時間	トランザクション・レート	バッチ・スループット	バッチ処理総数	データ容量	入力データ量										同時利用ユーザー数	
a. ビジネス・システム全体の品質属性・制約	1. ビジネス基礎数値																○	
	2. ビジネス制約																○	
	3. ビジネス柔軟性															○	○	
b. ビジネス・プロセスに関する品質属性・制約	4. 処理時間	○	○															
	5. 処理量			○	○	○	○	○										
	6. 可用性	i. サービス時間帯								○								
		ii. 計画的停止許容度									○							
		iii. 計画外停止回復目標										○	○					
7. セキュリティー (実行権限)															○			
c. 情報に関する品質属性・制約	8. 情報の保管						○											
	9. 情報の利用 (鮮度・断面など)						○					○						
	10. セキュリティー (情報の保護)															○		
d. インターフェースに関する品質属性・制約	11. アクター数								○									
	12. アクター制約	○													○			
	13. セキュリティー (授受情報の保護)															○		

縦軸 = BNFR の定義項目
横軸 = 主要な SNFR 定義項目

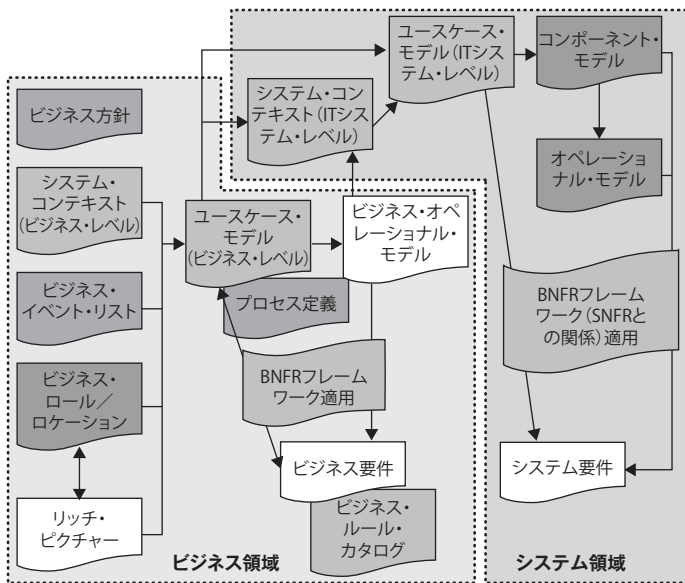


図5. 成果物関連図

5. BNFRフレームワーク適用事例

本章では本論文で提案したフレームワークを適用した事例を紹介する。

当事例のプロジェクト概要は以下の通り：

- (ア) 業界：金融
- (イ) ビジネス目的：新たな法令に対応する業務を支援するITシステムを構築する。
- (ウ) 前提事項：サードベンダーのパッケージ適用が経営レベルで決定していた。
- (エ) プロジェクト局面：要件定義

事例で筆者が作成した成果物関連図を図5に示す。IBM Unified Method Frameworkのアプリケーション開発手法とSE&Aのプロセス・作成物を考慮して、プロジェクト計画を策定した。

5.1 フレームワークを活用したBNFR定義

当プロジェクトでは、ビジネス側（エンドユーザー）およびIT側を代表するお客様担当者との打ち合わせを通じて作成物の確認やBNFR定義を行った。BNFRの定義項目によってとらえるべき視点が異なるため、それを意識した要求抽出を心掛けた。

ビジネス目標を確認しステークホルダーを特定する過程で、BNFRフレームワークの「視点a」の項目を議題に挙げることで、ビジネス・システム全体の要件の抽出が容易になった。また、BUCの作成にあたって記述フォームを定義したが、この中で「視点b」の項目を明記し、BUC定義作業においてエンドユーザーにBNFRを意識してもらうことで、ビジネス・プロセスに関するBNFRを抽出した。「視点c」に関しては、BUC定義と並行して使用する情報（ビジネス・エンティティ）を定義し、その属性として規定した。「視点d」に関しては、ステークホルダー定義およびBUC定義の過程で、

BUCのアクター属性として定義した。

事前定義されたフレームワークを適用することで、ステークホルダーやBUC定義の過程でBNFRを意識して作業を行い、効率的かつ網羅的なBNFR定義を実施できた。また、ビジネス視点での議論の過程でNFRを議題として提示し、ITの専門家ではないエンドユーザーに半ば強制的に要求品質を意識してもらうことで要件確定を推進する効果を得るとともに、ビジネス側とIT側との意思疎通が容易になることを確認した。BNFR定義は、ビジネス側とIT側の双方のステークホルダーの協業が必須である。

5.2 BNFRを参照したSNFR定義

フレームワークを適用して定義されたBNFRを参照してSNFR定義を行った。結果として、大半のSNFRをBNFRからの変換により定義できた。一部のSNFR（応答時間など）は、ビジネス要件とは関連付けずに定義したが、これは当事例の対象が社内業務であり、コールセンターのように応答時間がビジネスのサービス・レベルに直接関係しないことが理由であると分析している。

フレームワーク適用によるBNFRからSNFRへの変換により、作業効率の向上だけでなく、BNFRをエンドユーザーと合意した上でSNFRを定義するというプロセスによりビジネス担当者とIT担当者双方との品質指標の合意が可能となり、結果として説得力のあるNFR定義ができた。

5.3 考察と今後の課題

当事例ではビジネス・サービスの視点から、フレームワークを用いてBNFRを定義することの有効性を確認できたが、BNFR/SNFR定義後のIT運用担当者との調整において新規要件が提示され、それに対するソリューションを別途検討する必要が生じた。業務およびITの運用に関するNFRの取り扱いが今後の課題として挙げられる。

6. おわりに

本論文では、BNFR定義を行うためのフレームワークを提案し、その有効性について論述した。本論文の提案・主張を以下に整理する。

- BNFR定義が重要であること
- BNFRフレームワークによる、BNFR定義項目の明確化（【課題1】の解決策の提示）
- BNFRフレームワークの定義項目と、SNFRとの関連性の明確化（【課題2】の解決策の提示）
- プロジェクトへの適用事例に基づいた、BNFRフレームワークの有効性

本論文で提案したフレームワークにより、ビジネス上の要求事項をBFRだけでなくBNFRに反映させることが可能になるとともに、そのBNFRに基づいてITシステム要件定義が可能になるため、ITシステムによるビジネス上の課題解決に貢献するものと考えられる。

なお、今後の課題としては、以下の2点が挙げられる。

- 今後の活動において、本論文のフレームワークを適用していく過程で新たな知見が得られた場合は、それらを反映させてフレームワークの洗練を図る。
- BNFR フレームワークをビジネス・サービス以外の視点（運用視点など）に拡張する。
これらの課題を解決することにより、ビジネスに貢献できるITシステム開発がさらに容易になるものと考えている。

謝辞

本論文のフレームワークを検討するにあたり、日本アイ・ビー・エムの大嶽隆児氏、山本久好氏から多くの貴重な助言、示唆をいただきました。謹んでここに深謝いたします。

参考文献

[1] The Standish Group: CHAOS report, The Standish Group International, Inc. (1995).

[2] 日経BP社: “2003年情報化実態調査,” 日経コンピュータ, 2003年11月17日号, pp.50-62 (2003).

[3] 日経BP社: “第2回 プロジェクト実態調査,” 日経コンピュータ, 2008年12月1日号, pp.44-49 (2008).

[4] Judy Barkal, 大嶽隆児, 長島礼: “「システムズ・エンジニアリング&アーキテクチャ」の手引き 第2回 ビジネス駆動型アプローチの実践 (その①),” ITアーキテクト, Vol.17, pp.66-72 (2008).

[5] ISO/IEC: Software engineering – Product quality – Part 1: Quality model, ISO/IEC 9126-1:2001.

[6] IEEE-SA Standards Board: IEEE Recommended Practice for Software Requirements Specifications, IEEE Std 830 – 1998, IEEE, ISBN0-7381-0332-2 (1998).

[7] 経済産業省 情報処理振興課, NTTデータ経営研究所, 日本情報システム・ユーザー協会: 非機能要求仕様定義ガイドライン, 日本情報システム・ユーザー協会, ISBN978-4-903477-13-8 (2008).

[8] 山本久好, 榎原彰: “オペレーショナル・モデリングにおける非機能要件の効果的検証方法,” ProVISION, No.41, pp.85-92 (2004).

[9] 長谷川正巳, 石田英理, 小川久範: “変化に即応できるインフラ設計手法,” ProVISION, No.50, pp.68-75 (2006).

[10] Lawrence Chung, Brian A. Nixon, Eric Yu, John Mylopoulos: Non-Functional Requirements in Software Engineering, Kluwer Academic Publishers, ISBN0-7923-8666-3 (1999).

[11] Alistair Cockburn: ユースケース実践ガイド, 翔泳社, ISBN4-7981-0127-3 (2001).

[12] INCOSE – A Consensus of the INCOSE Fellows, <http://www.incose.org/practice/fellowsconsensus.aspx> (2008.09).

[13] 最新システムエンジニアリング情報館 – システムエンジニアリング概要, <http://www.se.rdy.jp/definition.html> (2008.09).

[14] 長井浩, 西原裕善, 大津留史郎, 吉田幸彦, 大嶽隆児: 匠 目指せ, ITアーキテクト講座 第7回 ITアーキテクチャとシステムズ・エンジニアリング (3), <http://www-304.ibm.com/jct03004c/easyaccess/jpgsind/contenttemplate/!/x/mlid=133120> (2008.05).

[15] 経済産業省 ソフトウェア開発力強化推進タスクフォース 要求工学・設計開発技術研究部会 非機能要求とアーキテクチャWG: 非機能要求記述ガイド, 情報処理推進機構 ソフトウェア・エンジニアリング・センター, <http://sec.ipa.go.jp/reports/20080717.html> (2008.07).

[16] IEEE-SA Standards Board: IEEE Guide for Information Technology – System Definition – Concept of Operations (ConOps) Document, IEEE Std 1362 – 1998, IEEE, ISBN0-7381-1407-3 (1998).



日本アイ・ビー・エム
システムズ・エンジニアリング株式会社
アーキテクチャー技術
インテグレーション・アーキテクチャー
ICP アドバイザリーITアーキテクト

桑原 淳一 Junichi Kuwabara

【プロフィール】

1994年日本IBM入社。UNIX[®]やネットワーク管理のスペシャリストとして活動後、ITアーキテクトとして基盤アーキテクチャーやセキュリティー・アーキテクチャーの策定とその実装を担当。現在は、さまざまなお客様のプロジェクトにて、ITアーキテクチャー設計とシステムズ・エンジニアリングの活動に従事している。



日本アイ・ビー・エム株式会社
GBS事業,
エンタープライズ・インテグレーション
マネージャー,
アドバイザリーITアーキテクト-Certified

柿本 達彦 Tatsuhiko Kakimoto

【プロフィール】

1993年日本IBM入社。大規模Webシステム構築、大規模パッケージITソリューションなどのプロジェクトにおいてリードITアーキテクトを担当。現場のシステムズ・エンジニアとして、ITソリューション構築プロジェクトにおけるテクニカル・マネージメントをはじめとする、システムズ・エンジニアリングの活動に従事している。