

ステークホルダ分析の反復による最適な要求獲得方法論に関する研究

2003MT004 青山 直樹 2003MT013 長谷部 敬祐

指導教員 青山 幹雄

1. はじめに

現在、様々な情報システムが日常生活やビジネスに至るまで私たちの生活に幅広く関わり、深く依存している。情報システム開発の成功は、いかに迅速に要求を満たすことができるか、ユーザの期待にどの程度合致した要求を実現できるかにかかっている。また、ステークホルダの要求が対立した際、どの要求を優先させるかという問題が発生する。その問題を解決する方法としてステークホルダ分析が注目されている。本研究は、ステークホルダにとって最適なシステムを構築する基となる要求獲得方法を提案する。

2. ステークホルダ分析の問題点と解決案

2.1. ステークホルダ分析の問題点

現状のステークホルダ分析はシステムに対して行われる。よって、要求の対立が発生した場合、システムを対象とした優先度で解決を図る。

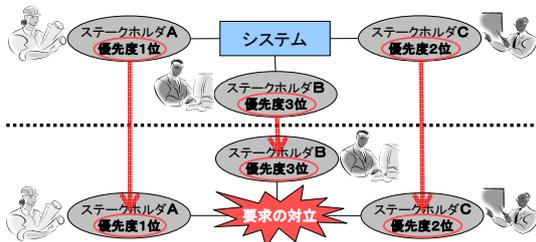


図 1 ステークホルダ分析の要求対立の決定方法

図 1 の上半分が、従来行われているシステムに対するステークホルダ分析を示す。ステークホルダとして A, B, C が存在している。ステークホルダ分析を行った結果、優先度が A は 1 位, B は 3 位, C は 2 位になったとする。開発中にステークホルダから要求を抽出し、その要求に対して対立が発生すると、対立を解決する尺度として下半分のようにステークホルダの優先度が用いられる。しかし、その優先度で要求の対立を解決すると、システムに対して優先される一部のステークホルダにとってのみ使いやすいシステムとなる。したがって、各ステークホルダの要求が反映されているとはいえない。

2.2. 解決案

要求ごとにステークホルダ分析を行うことで問題点を解決する。ステークホルダ分析方法論として、ISAP(Iterative

Stakeholder Analysis Process:反復型ステークホルダ分析プロセス)を提案する。反復型とは、要求の数だけステークホルダ分析を反復することを意味する。要求に対するステークホルダ分析を行うために要求獲得でステークホルダ分析を用いる段階と、要求獲得全体の概要を図 2 に示す。

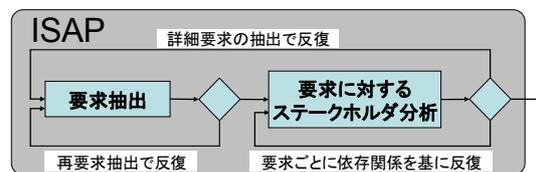


図 2 ISAP の位置づけ

ステークホルダ分析を図 2 のように要求抽出の後に行う。ISAP は要求抽出, 要求に対するステークホルダ分析, ISAP 全体を反復する。反復の定義は 3.3 で述べる。

3. 要求獲得方法論 ISAP

3.1. ISAP のプロセス

提案する ISAP のプロセスを図 3 に示す。

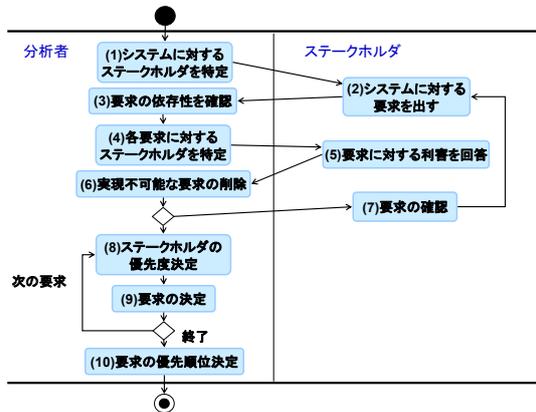


図 3 ISAP のプロセス

3.2. ISAP の詳細なプロセス

(1) システムに対するステークホルダを特定

システムのステークホルダをすべて特定する。何をステークホルダとし、どのレベルまでステークホルダを特定するのかという一般的な基準を定義するのは困難であるため、本研究では、ステークホルダを顧客から獲得した分析可能な範囲とする。

(2) システムに対する要求を抽出

ステークホルダからシステムに対する要求を抽出する。図 2 で要求抽出を反復した場合は再要求抽出を行い、ISAP 全体を反復した場合は要求の詳細内容を抽出する。詳細内容は一度で全ての要求を抽出することはできないこと、粒度の大きな要求を抽出すると、その粒度を小さくして詳細内容を抽出する必要があるため、抽出が不可欠となる。

(3) 要求の依存性を確認

(2) で抽出した要求に対し、要求の依存関係を確認する。依存関係は、次の 2 パターンを確認する。

- A) 依存関係がある要求の中でどれか 1 つに絞る必要のある要求の集合
 - B) 依存関係がある要求の中である 1 つの要求の有無によって他の要求の有無が判断される要求の集合
- これら A), B) の依存関係は、A)→B) の順番で確認する。まず、A) の例を図 4 に示す。

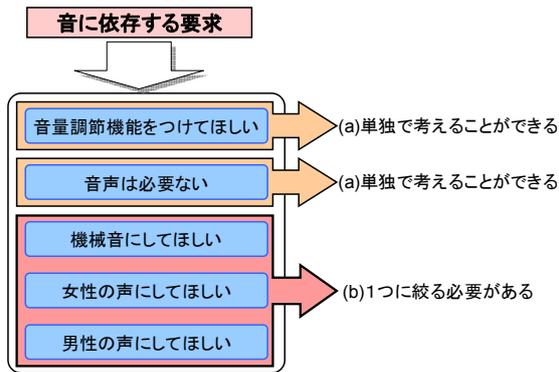


図 4 依存関係の例 A)

分類の粒度を順に小さくしていくことで絞り込み、最終的に図 4 に示すように音に関する要求が 5 つの場合、(a) 単独で考えることができる要求、(b) 1 つに絞る必要のある要求に分類する。依存関係は(a), (b) のような関係を表し、依存性のある要求を削除してプロセスが進んでから要求間の統一性が欠けることを防ぐためにすべき作業である。したがって、要求を削除する前に行う必要がある。本研究では(a) を「YES/NO 要求」、(b) を「HOW 要求」と呼ぶ。HOW 要求に関して、新規のシステムの場合は(b) の 3 つの要求で考えることができるが、既存のシステムは(b) の選択肢に「今まで通り」という要求を加える場合も考えられる。

次に、B) の例を図 5 に示す。図 5 は、A) を B) の定義で階層化したものである。階層化は要求を削除する前に考慮する必要がある。音声の有無が下位 2 つの要求に影響し、音声が必要ないと判断した場合は下位 2 つの要求を自動的に削除する。そのため、要求の対立を解決する必要はなく、ステークホルダ分析の対象外となる。よって、ステークホルダ分析は階層化した最上位の要求から行う必要がある。

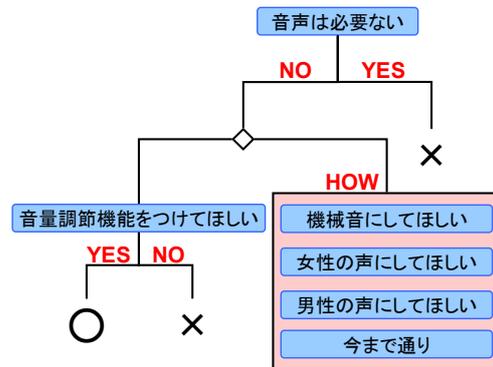


図 5 依存関係の例 B)

(4) 各要求に対するステークホルダを特定

各要求に直接利害があるステークホルダをシステムのステークホルダの中から特定する。

(5) 要求に対する利害を抽出

分析者が要求に対する利害を聞くことで、ステークホルダが回答する。利害を聞く順番は、開発者→顧客(運用者)→その他のステークホルダである。要求が技術的に不可能なことも考えられる。実現できない要求に対して他のステークホルダに意見を聞くことは、時間の無駄となる。顧客は、システムを依頼する立場として導入する目的がある。要求がシステムの目的に沿わないことも考えられる。また、運用上、可能な要求と不可能な要求がある。

(6) 実現不可能な要求の削除

(5) でステークホルダが回答した利害に対して、開発者、顧客(運用者)が技術的、運用的に不可能な要求、またはシステムの目的に沿わない要求をこの段階で削除する。

(7) 要求の確認

(6) で要求が実現不可能であるため削除した理由を考慮し、ステークホルダは要求の実現可能性を把握する。それを確認した上で、新しい要求を抽出できる。このプロセスは(2) から反復する。また、この反復を終了する基準は要求抽出が一定水準に達したこととする。一定水準は、以下の 2 点である。

- (A) 要求が前回に抽出した要求と類似していた場合
- (B) ステークホルダに要求がない場合

(8) ステークホルダの優先度決定

要求ごとに、ステークホルダの優先度を決定する。優先度の決定には、表 1 と以下の方法を用いる。

表 1 ステークホルダの利害と影響マップ

ステークホルダ	要求	利害	要求に与えるインパクト	優先度
ステークホルダA				
ステークホルダB				
ステークホルダC				
ステークホルダD				
⋮				

ISAP は要求の数だけ表を作成する。表 1の要求は(3)の YES/NO 要求か HOW 要求のいずれかとなり、利害は(5)でステークホルダから聞き出した利害となる。その利害に対して、分析者は要求に対するインパクトを Low~High+の 6 段階で評価する。

このプロセスの最後に優先度を決定する。既存のプロセスはインパクトの平均値順に優先順位を決定する。しかし、ISAP は優先順位ではなく平均値をそのまま記入し、ステークホルダの優先度とする。

(9) 要求の決定

(8)で評価したステークホルダの優先度を基に、要求ごとに対立したステークホルダの優先度を合計し、対立を解決することで要求を決定する。

(10) 要求の優先順位決定

(9)で決定した要求を、コスト・期間を考慮して優先順位の高い要求から実現していく。例を図 6に示す。

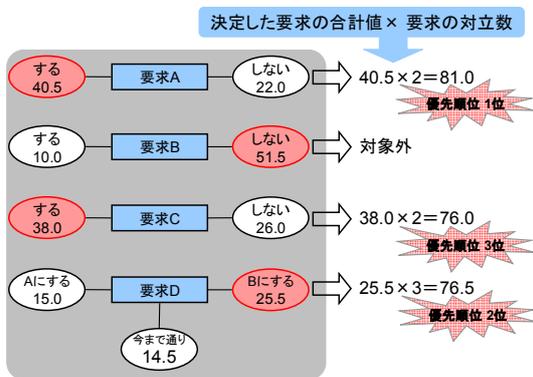


図 6 優先順位の決定方法

優先順位は(9)で述べた対立ごとの優先度の合計値だけを考慮して決めることができない。YES/NO 要求は 2 つの対立した要求の合計値であることに対し、HOW 要求は 3 つ以上の要求の対立を解決した合計値となる。そのため、HOW 要求は 1 つの要求に対する優先度の合計値が低くなる。YES/NO 要求と HOW 要求を同じレベルで考えるため、決定した要求の合計値に要求の対立数を掛けた値で優先順位を決定する。

3.3. 反復の定義

ISAP は、要求ごとに反復し ISAP 全体の要求獲得でも反復する。(3)の B)で述べた依存関係を考え、要求の詳細を聞く必要がある。これらは、ISAP を規則的に反復することでより効率的に要求を抽出できる。

図 7に示すように、反復の 1 回目「音声は必要ない」という要求のステークホルダ分析を優先する。分析結果が YES の場合は後の要求を分析する必要はないが、NO の場合は 2 回目を行う。ここでは、「音量調節機能をつけてほしい」という要求は HOW 要求に依存関係がないため、同時にステークホルダ分析を行える。その後 HOW 要求が

「機械音にしてほしい」という要求に決定した場合、どのような機械音にするのかといった詳細な要求は 3 回目に聞く。

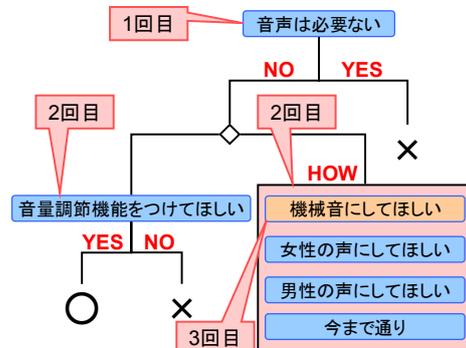


図 7 依存関係による反復の順番

4. 例題による評価

システムを対象とする従来のステークホルダ分析と ISAP を利用した場合の、(1)分析方法の違いによる優先度、(2)要求獲得方法の違いによる分析結果、(3)ステークホルダの満足度の 3 項目について比較し、ISAP の有用性を評価する。

4.1. 実証実験

分析対象を南山大学で運用している証明書自動発行機システムとし、ステークホルダを開発、運用者である南山大学とユーザである学生 20 人として検証した。要求の獲得方法としてはアンケートを用いた。ISAP によって抽出した要求は 19 件となった。

4.2. 分析方法の違いによる優先度の比較

本検証では、従来のシステムを対象としたステークホルダ分析で特定した最も優先度の高いステークホルダは顧客となり、対立した要求に顧客の利害がない場合でも顧客の意見を中心に実現する要求を決定することになる。この問題点に対し、ISAP は要求ごとにステークホルダを特定して優先度を考慮するため、システムに対して優先度の高いステークホルダが要求に対しても優先度が高くなることは限らなかつた。したがって要求ごとに分析することで重要なステークホルダの意見を反映できたといえる。

4.3. 要求獲得方法の違いによる分析結果の比較

決定した同じ要求 8 件の分析結果を比較し、図 8に示す。対立数は要求に対する反対意見の数であり、期待度はステークホルダ間の対立を測る値である。期待度の計算を式(1)に示す。

$$\text{期待度} = \frac{\text{対立数の最大値} - \text{対立数}}{\text{対立数の最大値}} \dots \text{式(1)}$$

システムを対象とした要求の優先順位はシステムに対する優先度で求めたため、要求の対立数に比例した。しかし、ISAP は複数の要求ごとにステークホルダから抽出した利害で優先度を考慮したため、優先順位は対立数に比例しな

かった。したがって、要求の(b)と(d)が大きく変化した。図 8 の右上に示す要求は対立数が少なく、優先度が高いためステークホルダの意見が一致している。これはステークホルダの期待が高いといえる。

また、システムを対象として獲得した要求の相関係数は 0.98, ISAP で獲得した要求の相関係数は 0.75 となった。ISAP はシステムを対象として要求を獲得するより優先度と期待度の相関が低い。

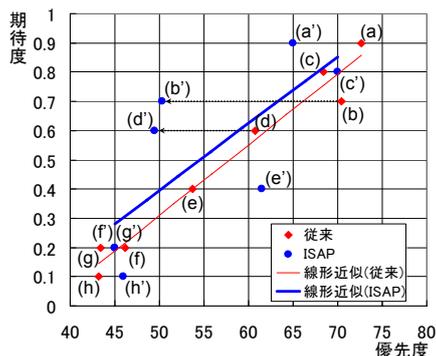


図 8 分析対象の違いによる結果の比較

4.4. ステークホルダの満足度

従来の方法と ISAP で決定した要求の満足度を比較した。満足度は、決定した要求の各ステークホルダの希望実現数と非希望実現数の差で求めた。その結果、20 名のユーザのうち満足度の向上は、5 人が従来の方法、7 人が ISAP、8 人がどちらの方法を用いても変わらなかった。したがって、ISAP を用いることでステークホルダにとってより満足度の高いシステムとなる。

5. 考察

5.1. 優先するステークホルダ

ステークホルダ分析は対象となるシステムの代表者を特定して限られたステークホルダ間で討議する傾向にある。本検証では、システムに対する優先度が上位 5 人のステークホルダの意見を採用すると、要求をどのように実現するかといった結果がすべてのステークホルダの優先度を考慮した結果と近似した。

ISAP は要求に関わるすべてのステークホルダを特定し、意見を反映させることでステークホルダの満足度を向上できると考えたため、要求の優先度を決定する際は抽出した要求ごとに代表者の意見で対立を解決することはできないと考えた。しかし、優先度の上位 5 人の要求を考慮すると、すべてのステークホルダの優先度を考慮した要求と等しくなった。したがって、要求ごとに優先度の上位 5 人までを考慮すればよいといえる。そこで、要求ごとにステークホルダを的確に特定することが重要である。

5.2. 要求ごとにステークホルダ分析を行う有用性

要求抽出を反復したことによって新たな要求を獲得した。

そこで、ステークホルダのシステムと要求の双方に対する優先度を考慮して決定した要求の満足度を比較した。ISAP を用いると、偏ったステークホルダの意見によるシステムではなく、ステークホルダにとって満足度の向上する結果が得られた。したがって、本研究で提案した ISAP は対象をシステムではなく要求とした点において従来の方法論より優れている。しかし、要求の数だけ分析を繰り返すため、従来の方法よりも時間がかかる。これは、要求定義を十分にするためには必要だと考える。

6. 今後の課題

ISAP は各要求の顧客、ユーザ以外のステークホルダの利害も抽出して考慮する必要がある。ISAP で作成した優先順位は、今後の課題として以下の点を考慮する。

(1) 緊急性

情報システムの導入の際、緊急性の高い要求は実現すべき優先順位を高くしなければならない。

(2) 開発コスト

ISAP で決定した要求の優先順位で開発し、優先順位の高い要求のコストが大きい場合、先に開発すると他の要求が実現できなくなる可能性がある。

(3) 開発期間

開発期間の延長によって開発コスト同様、優先度の低い要求が実現できなくなる可能性がある。

7. まとめ

本研究では、要求が対立した際に要求の必要性の確信度とステークホルダの満足度の向上を目的とし、要求を対象としてステークホルダ分析を反復する ISAP を提案した。ISAP は要求ごとにステークホルダの意見を反映させ、ステークホルダの優先度で対立の解決を図ることによって実現すべき優先順位をつけることができる。検証として、要求獲得方法の比較により ISAP の有用性の評価を行った。

参考文献

- [1] 青山幹雄, 要求分析から要求工学へ, ソフトウェアプロセスエンジニアリングシンポジウム(SPES2004)ワークショップ, 2004.
- [2] M. C. Ruhe, Stakeholder Analysis in Health Services Management, <http://www.case.edu/med/epidbio/>.
- [3] H. Sharp et al., Stakeholder Identification in the Requirements Engineering Process, Proc. of the 10th Int'l Workshop on Database and Expert Systems Applications, 1999, pp. 387-391.
- [4] L. W. Smith, Project Clarity through Stakeholder Analysis, Cross Talk, Dec. 2000, pp. 4-9.